

Akeeba Ticket System for Joomla!™ 4

Nicholas K. Dionysopoulos

Akeeba Ticket System for Joomla!™ 4

Nicholas K. Dionysopoulos

Copyright © 2011-2022 Akeeba Ltd

Abstract

This book covers the use of the Akeeba Ticket System component which allows you to easily create a support ticket system area in your Joomla!™-powered web site.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license can be found on-line at <http://www.gnu.org/licenses/fdl.html>.

Table of Contents

I. User's Guide	1
1. Introduction and installation	5
1. Introduction to Akeeba Ticket System	5
2. Requirements and compatibility	7
3. Migration notes	8
3.1. Migration from Akeeba Ticket System 2.x or 3.x to 5.x	9
3.2. Migrating from Akeeba Ticket System 4.x to 5.x	9
4. Installing Akeeba Ticket System	12
4.1. Installing or manually updating the component	12
4.1.1. Install from URL	13
4.1.2. Upload and install	13
5. Upgrading from Core to Professional	14
6. Automatic updates	14
6.1. Troubleshooting the update	16
6.1.1. Addressing server issues	16
6.1.2. Check the validity of your Download ID	17
6.1.2.1. Check your subscription status	17
6.1.2.2. Multiple Professional edition Akeeba extensions with different Download IDs	18
6.1.2.3. Entering or changing your Download ID after an update is available	18
6.1.3. Updates are showing after installing the latest version	19
6.1.4. Updates not showing despite having an older version	19
6.1.4.1. Check the update site	19
6.1.4.2. Refresh the update cache	19
6.1.5. Miscellaneous troubleshooting and information	20
6.1.5.1. The update fails to download	20
6.1.5.2. Updating with a third party service fails	20
6.1.5.3. Manual update	20
6.1.5.4. Update installation problems	21
7. Entering your Download ID	21
8. Requesting support and reporting bugs	23
9. Uninstallation	23
2. Overview and Quick Setup	25
1. Overview	25
2. Quick Setup	25
3. Deep dive into advanced features	29
1. Permissions	29
2. Configuring Joomla's WYSIWYG editor (TinyMCE)	30
2.1. In-line images	31
3. Attachments	32
4. User custom fields	32
4.1. Avatar (profile image)	33
4.2. Country	34
5. Custom fields	35
5.1. Custom fields in Categories	36
5.2. Custom fields in Tickets	37
5.2.1. Support staff-only fields	40
5.2.2. Client-only fields	42
5.2.3. Conditional fields (or how to ask clients for additional information)	43
5.2.4. Replacing the geolocation field	48
6. Tags	48

6.1. Support staff-only tags	49
7. Nested Categories	49
8. InstantSearch	50
8.1. Smart Search menu items	50
8.2. External search engine	52
9. Guest tickets	52
10. Receiving emails	54
10.1. The plugin options	55
10.2. Integrating with GMail / G Suite	57
10.2.1. Initial Configuration	58
10.2.2. Troubleshooting	77
10.2.3. Frequently asked questions	79
11. Customising the frontend layout	80
11.1. Component view template overrides	81
11.1.1. Custom fields in view template overrides	82
11.1.2. Layout overrides	82
11.2. Module view template overrides	83
11.3. Media (CSS, JavaScript and image) overrides	84
11.4. Language overrides	85
12. Custom module positions in the front-end	85
13. CRON jobs to set the ticket system offline	87
4. The Component	89
1. The Control Panel page	89
2. Component Options	90
2.1. Common	90
2.2. Frontend	91
2.3. Categories View	93
2.4. Tickets view	93
2.5. InstantSearch	95
2.6. Attachments	95
2.7. Security	96
2.8. Automation	97
2.9. Permissions	97
3. Categories	99
4. Tickets	103
4.1. Editing a ticket	106
5. Canned Replies	110
6. Email Templates	111
6.1. Allowed variables in e-mails	112
7. Auto-replies	113
8. The menu item types	115
5. The plugins	116
1. The “ats” plugins	116
1.1. Automatically close old tickets	116
1.2. Automatic replies	116
1.3. Delete Manager Notes on ticket close / unpublish	117
1.4. Fetch email	118
1.5. Remove Obsolete Attachments	118
2. The “actionlog” plugins	118
2.1. Action Log plugin	118
3. The “console” plugins	118
3.1. Akeeba Ticket System	118
4. The “content” plugins	119
4.1. Akeeba Ticket System Custom Fields Show-on Behavior	119

5. The “editors-xtd” plugins	119
5.1. Akeeba Ticket System Canned Replies	119
6. The “finder” plugins	119
6.1. Akeeba Ticket System	119
7. The “user” plugins	120
7.1. Akeeba Ticket System	120
6. The modules	121
1. Administrator modules	121
1.1. Ticket Graphs	121
1.2. Ticket Stats	121
2. Site modules	121
2.1. Tickets info	121
7. Automation and CRON scripts	123
1. CLI script for CRON jobs	123
2. URL for CRON jobs	124
3. CRON commands	125
II. Appendices	127
A. GNU Free Documentation License	129

Part I. User's Guide

Table of Contents

1. Introduction and installation	5
1. Introduction to Akeeba Ticket System	5
2. Requirements and compatibility	7
3. Migration notes	8
3.1. Migration from Akeeba Ticket System 2.x or 3.x to 5.x	9
3.2. Migrating from Akeeba Ticket System 4.x to 5.x	9
4. Installing Akeeba Ticket System	12
4.1. Installing or manually updating the component	12
4.1.1. Install from URL	13
4.1.2. Upload and install	13
5. Upgrading from Core to Professional	14
6. Automatic updates	14
6.1. Troubleshooting the update	16
6.1.1. Addressing server issues	16
6.1.2. Check the validity of your Download ID	17
6.1.2.1. Check your subscription status	17
6.1.2.2. Multiple Professional edition Akeeba extensions with different Download IDs ...	18
6.1.2.3. Entering or changing your Download ID after an update is available	18
6.1.3. Updates are showing after installing the latest version	19
6.1.4. Updates not showing despite having an older version	19
6.1.4.1. Check the update site	19
6.1.4.2. Refresh the update cache	19
6.1.5. Miscellaneous troubleshooting and information	20
6.1.5.1. The update fails to download	20
6.1.5.2. Updating with a third party service fails	20
6.1.5.3. Manual update	20
6.1.5.4. Update installation problems	21
7. Entering your Download ID	21
8. Requesting support and reporting bugs	23
9. Uninstallation	23
2. Overview and Quick Setup	25
1. Overview	25
2. Quick Setup	25
3. Deep dive into advanced features	29
1. Permissions	29
2. Configuring Joomla's WYSIWYG editor (TinyMCE)	30
2.1. In-line images	31
3. Attachments	32
4. User custom fields	32
4.1. Avatar (profile image)	33
4.2. Country	34
5. Custom fields	35
5.1. Custom fields in Categories	36
5.2. Custom fields in Tickets	37
5.2.1. Support staff-only fields	40
5.2.2. Client-only fields	42
5.2.3. Conditional fields (or how to ask clients for additional information)	43
5.2.4. Replacing the geolocation field	48
6. Tags	48
6.1. Support staff-only tags	49
7. Nested Categories	49

8. InstantSearch	50
8.1. Smart Search menu items	50
8.2. External search engine	52
9. Guest tickets	52
10. Receiving emails	54
10.1. The plugin options	55
10.2. Integrating with GMail / G Suite	57
10.2.1. Initial Configuration	58
10.2.2. Troubleshooting	77
10.2.3. Frequently asked questions	79
11. Customising the frontend layout	80
11.1. Component view template overrides	81
11.1.1. Custom fields in view template overrides	82
11.1.2. Layout overrides	82
11.2. Module view template overrides	83
11.3. Media (CSS, JavaScript and image) overrides	84
11.4. Language overrides	85
12. Custom module positions in the front-end	85
13. CRON jobs to set the ticket system offline	87
4. The Component	89
1. The Control Panel page	89
2. Component Options	90
2.1. Common	90
2.2. Frontend	91
2.3. Categories View	93
2.4. Tickets view	93
2.5. InstantSearch	95
2.6. Attachments	95
2.7. Security	96
2.8. Automation	97
2.9. Permissions	97
3. Categories	99
4. Tickets	103
4.1. Editing a ticket	106
5. Canned Replies	110
6. Email Templates	111
6.1. Allowed variables in e-mails	112
7. Auto-replies	113
8. The menu item types	115
5. The plugins	116
1. The “ats” plugins	116
1.1. Automatically close old tickets	116
1.2. Automatic replies	116
1.3. Delete Manager Notes on ticket close / unpublish	117
1.4. Fetch email	118
1.5. Remove Obsolete Attachments	118
2. The “actionlog” plugins	118
2.1. Action Log plugin	118
3. The “console” plugins	118
3.1. Akeeba Ticket System	118
4. The “content” plugins	119
4.1. Akeeba Ticket System Custom Fields Show-on Behavior	119
5. The “editors-xtd” plugins	119
5.1. Akeeba Ticket System Canned Replies	119

6. The “finder” plugins	119
6.1. Akeeba Ticket System	119
7. The “user” plugins	120
7.1. Akeeba Ticket System	120
6. The modules	121
1. Administrator modules	121
1.1. Ticket Graphs	121
1.2. Ticket Stats	121
2. Site modules	121
2.1. Tickets info	121
7. Automation and CRON scripts	123
1. CLI script for CRON jobs	123
2. URL for CRON jobs	124
3. CRON commands	125

Chapter 1. Introduction and installation

1. Introduction to Akeeba Ticket System

Akeeba Ticket System is a Joomla! 4™ help desk component. Users can be given permissions to create support tickets in one or more support categories.

Only the person who filed the ticket and specifically authorised users, called the support staff or “managers”, can reply to them. This is unlike a forum where anyone can reply.

Support tickets can be private or public. Private tickets are only visible to the user filing them and the support staff of this category. Public tickets are visible to everyone — but, still, only the user filing the support ticket and the support staff can reply to it.

Tickets can optionally have up to ten attached files. You can configure the attachments to have the same visibility as the ticket itself or always be treated as private. In the latter case only the user filing the support ticket and the support staff can see the attachments, even if the ticket itself is public.

All of that, including who can file public and private tickets and who can attach files, is controlled with the built-in Joomla permissions system, commonly referred to as the Joomla ACL.

Key features:

- Native Joomla 4 component using the Joomla backend (MVC) and frontend (Bootstrap 5) frameworks with plain vanilla JavaScript.
- Deep integration with the the Joomla Permissions a.k.a. Joomla ACL for fine grained access control.
- Nested categories at an infinite depth, using Joomla's native Category management system.
- Integration with Joomla Fields a.k.a. custom fields in Categories and tickets. Using standard Joomla ACLs you can create custom fields which are only editable and/or visible to just the support staff, the person filing the ticket or a combination thereof. You can even create sets of fields which only become editable when a manager changes the value of another field.
- Integration with Joomla tags for tickets and users. You can use standard Joomla ACLs to make these tags only visible to the support staff.
- Meaningful and predictable SEF URLs with automatic canonical URL redirection without the need for a third party SEF / SEO component, using the standard Joomla 4 component SEF router API.
- Dark Mode support with our optional, free of charge DarkMagic plugin.
- Custom module positions, allowing you to easily customise front-end pages.
- You can turn off replies / new tickets with an optional message e.g. for holidays or planned maintenance periods.
- Automatic HTML sanitation to prevent XSS exploits and other security issues using the third party HTMLPurifier library.
- Private and public tickets.
- Display user avatars using Gravatar or a Joomla field you have created in the user's profile on your site.

- Display the user's country. The country information comes from a Joomla field you have created in the user's profile on your site.
- Attachments support. Optionally make attachments only visible to and downloadable by the ticket owner and the support staff, even on public tickets.
- User profile plugin. Users can set their signature for the ticket system and/or add email addresses their ticket replies should be CC'ed to.
- Manager notes, for keeping private notes visible only to support staff.
- Pre-filled new ticket body, customisable per support category, guiding the user to enter all the necessary information you need to help them.
- Email notifications of new, replied to and edited tickets with customisable, HTML templates.
- Receive new tickets or ticket replies via email.
- Integrates with Smart Search, making *public* tickets searchable. You cannot search private tickets; Joomla's Smart Search does not allow excluding results based on the currently logged in user.
- Automatically display Smart Search results or third party search engine results when the user files a new support ticket, reducing your support load. This can be used to implement an FAQ (Frequently Asked Questions) functionality / knowledge base using standard Joomla articles — no need to reinvent the wheel!
- Optionally allow guest tickets (tickets filed by users who do not yet have a user account on your site).
- Saved replies ("canned replies") your support staff can easily use when replying to tickets.
- Automatic replies based on rules including ticket age, who has replied to it, content etc.
- Automation for common administrative tasks using either a real, CLI-based CRON system or by accessing a specific URL.
- Custom ticket statuses when the default (Open, Pending and Closed) are not enough for your use case. Up to 99 additional ticket statuses are supported.
- Time cards, i.e. how much time each member of the support staff has spent replying to tickets over a specific period of time.

Important

Some features are only available in the for-a-fee Akeeba Ticket System Professional edition. They are not present in the Akeeba Ticket System Core edition.

What Akeeba Ticket System is NOT

The term "ticket system" may mean different things to different people. Before we discuss what Akeeba Ticket System is not, let us explain the context in which Akeeba Ticket System is designed.

A ticket system, as implemented in Akeeba Ticket system, means that you have customers creating tickets in different categories. Each category typically represents a particular skill set, area of expertise or product your company or organisation offers support for. You have teams of support people for each of these categories. Any one of them can answer any ticket in the category. On most tickets only one person will handle each ticket and if they have to escalate they will assign it to someone more apt to the task, implemented by assigning the ticket to the other person.

If a ticket would only be visible to support staff if someone else (let's call them an "admin") assigned it to them then the support staff would not answer any tickets unless that "admin" went first and dispatched tickets. That's quite the opposite concept to a ticket system. If you need something like that stop looking for a support ticket system and start looking for CRM software.

Akeeba Ticket System is not a CRM, nor are we interested in making it possible to be used as a CRM. This design goal has some very important implications on what is and is not possible with Akeeba Ticket System:

- You cannot have different administrative / concierge staff and support staff. If someone can assign tickets to someone else they can also themselves be assigned tickets and reply to them. Conversely, if someone can be assigned and reply to tickets they can also assign tickets to someone else.
- You cannot have predefined escalation paths (workflows). The status of a ticket can be changed to any other status any time by any member of the support staff. Escalation paths (e.g. an Open ticket must first go to first level support before being tagged as "needs review by supervisor") is something that a CRM can handle and requires separate roles for the staff which, as we explained, is something that will not be implemented in ATS.
- The support staff will see all tickets, NOT just the tickets assigned to them. They will see all open tickets with the Latest Open Tickets view, even if they are assigned to other members of the support staff.
- Any member of the support staff can reply to any ticket, even if that ticket is assigned to a different member of the support staff. In our experience, when doing support you will frequently need the combined experience of two or more members of your support staff to handle complex cases.
- You can see the total time spent on each client next to each client's username in one of their tickets. You cannot create a report to display the total time spent per user for multiple users or with specific parameters (e.g. date range or support categories). This information can, however, be pulled very easily from the database by querying the #__ats_tickets table.
- You cannot handle client invoicing through Akeeba Ticket System. This is what a CRM sometimes does (and definitely what an invoicing application is designed to do).
- You cannot have an FAQ / Knowledge Base inside Akeeba Ticket System. It makes no sense since Akeeba Ticket System runs inside Joomla which is in fact a Content Management System. Managing articles, what an FAQ / Knowledge Base is, is really what a CMS is designed to do. Implementing a CMS within a CMS would be a waste of effort if not completely pointless. Use Joomla's excellent content management features instead.

If you need any of the above features the Akeeba Ticket System is NOT for you. You would become frustrated with it and ultimately not be able to use it.

If you are unsure if something you want is possible please try using the free of charge Core version and/or ask us a pre-sales question. We always reply to pre-sales questions honestly. If your use case is not compatible with the existing features and / or design goals of Akeeba Ticket System we will tell you upfront. We prefer to lose a sale than having a frustrated user because we recommended that they use software unsuitable for their use case.

2. Requirements and compatibility

Akeeba DocImport requires the following server-side configuration:

- Joomla!™ and PHP version compatibilities are detailed in our Version Compatibility matrix [<https://www.akeeba.com/compatibility.html>].
- MySQL 5.5 or later. Earlier database server versions will not be supported. Do note that earlier releases of MySQL are obsolete and not supported any more by Oracle (the company who controls the development of MySQL).

As far as the browser is concerned, you can use any modern version (i.e. published within the last year) of Microsoft Edge, Safari, Opera, Firefox or Google Chrome. We no longer support Internet Explorer; our software will display incorrectly or not work at all on this old, buggy and obsolete browser.

In any case, you must make sure that JavaScript is enabled on your browser for the administration interface to work. If you are using AVG Antivirus, please disable its Link Checker feature (and reboot your computer) as it is known to cause problems with several JavaScript-based web applications, including Akeeba Ticket System.

You are very strongly advised to disable or properly configure Internet firewalls, antivirus applications and browser extensions which interfere with the site's loading such as script blockers (such as NoScript) and ad blockers (such as AdblockPlus) *only for your site's domain* to ensure that the administration interface (both in the backend and the frontend of the site, the latter if you are logged in as a ticket system staff member) is displayed correctly. Remember that these applications and browser extensions are designed to protect you against third party sites. As a result they are very aggressive and WILL break your own sites. We can't do anything about it: your computer and your browser are under your control alone.

Kindly note that our software is compatible with Joomla's caching. If you are using an external caching solution (such as LiteSpeed Cache, Varnish etc) or a CDN / third party service (such as CloudFlare) kindly note that we are not responsible if your site displays out of date information or otherwise misbehaves. Our responsibility ends at sending the correct HTTP headers to indicate that a page **must not** be cached and clearing Joomla's cache when information which can participate in a cached page is modified. What happens beyond that is objectively outside the scope of our software and into the configuration scope of the external caching / CDN / third party service you are using.

3. Migration notes

While we try to make migration between major versions of Akeeba Ticket System as smooth as possible this is not always possible. For reasons beyond us, the first two major versions introduced a lot of technical debt that had to be repaid over the next several years in another two major and one minor rewrite of the software, making for a problematic migration experience.

Akeeba Ticket System 1 was using Bootstrap 2 as its CSS framework. It sprang out of a proof of concept hastily put into production when the third party ticket system we were using until that point in time was discontinued without prior notice. This created a lot of technical debt that had to be repaid over the next decade.

Akeeba Ticket System 2 was using XML forms for most of its interface and was compatible with Bootstrap 2 and 3 but didn't allow you to override its output very easily. The reason for that odd choice is that at the time Joomla had decided to use our FOF framework as the core supported Rapid Application Framework. Using XML forms would be the standard way to write interfaces, each template providing their own rendering backend, making the output truly template-agnostic. By the time we had finished converting our software Joomla changed its mind and we were stuck with an inflexible output method which necessitated yet another massive rewrite. This was a massive setback.

The combined effects of ATS 1 and 2 was that the first 5 years of development left us with a component that was nowhere near what we wanted to offer and in need of successive major rewrites to bring it in line with our vision, something only tampered by available time and the uncertainty of when Joomla 4 would be released at that time.

Akeeba Ticket System 3 was written with Blade view template files and our own Akeeba FEF framework. It also moved from using jQuery to using plain vanilla JavaScript in anticipation for changes which were scheduled for Joomla 4. This was the first version that made it practical to have template overrides to customise the entirety of the component's output. At this point we were interested in striking a balance between adding features and repaying the technical debt from the first two versions of the software as we didn't know at which point in time Joomla 4 would be released.

Akeeba Ticket System 4 was developed when the timeline for Joomla 4's release started becoming more clear. Our goal was to provide a version of ATS which would bridge Joomla 3 and 4 before embarking on a full rewrite which would do away with the remainder of the technical debt incurred in the first two major versions. We upgraded the

backend (FOF) and frontend (FEF) frameworks and further reworked our JavaScript to increase performance and remove all traces of inline scripts. Template overrides were mostly but not fully compatible with ATS 3. You mainly had to change FOF3 with FOF4 in them to make them compatible.

With Akeeba Ticket System 5 we stopped using our own frameworks, instead using the core Joomla backend framework (Joomla core MVC) and Joomla's standard frontend framework (Bootstrap 5). This was not possible before; Joomla's core MVC in Joomla 4 is a MAJOR update to what Joomla 1.6 to 3.10 inclusive offered, making it possible to use it for complex extensions like ours. We also further refined our JavaScript to make it even more lightweight and reduce the number and difficulty of media overrides should you choose to use a different frontend framework.

The only downside is that ATS 5 being the final major rewrite to realise our original vision for ATS it needs more migration work on your part than its predecessors. The upside is that barring any massive, unforeseen changes in Joomla itself we do not anticipate another major rewrite — frankly, we are tired of having to do that. We expect that future versions in the 5.x version range and beyond will be iterative improvements, making it easier for all of you to keep up and for us to improve on the component without the distraction a full blown rewrite.

3.1. Migration from Akeeba Ticket System 2.x or 3.x to 5.x

If you have an old Joomla 3 site you may have to do a multi-step migration to Joomla 4 and Akeeba Ticket System 5.

First, make sure that you have installed the latest Akeeba Ticket System version for your Joomla and PHP version per our Compatibility page [<https://www.akeeba.com/compatibility.html>].

Upgrade to the latest Joomla 3 version and PHP version supported by the Akeeba Ticket System version you installed.

Repeat these last two steps until you have the latest Joomla 3.10 version. You should also have the latest Akeeba Ticket System 4 version.

Upgrade your site to Joomla 4.

Follow the migration instructions from Akeeba Ticket System 4.x to 5.x.

3.2. Migrating from Akeeba Ticket System 4.x to 5.x

To ease the migration of your sites we made sure that both Akeeba Ticket System 4.x and 5.0 support Joomla 4.0. Before updating Akeeba Ticket System to version 5 you need to be aware of some important changes in this version and what you need to do to address them.

Removed features

The Credits system has been removed. This feature was rarely used, made our code overly complex and we have reached the point where there is no officially supported way to add credits anyway. If you relied on this feature you can remain on Akeeba Ticket System 4 noting that its support will end on August 2022 and have no plans on making it compatible with Joomla 4.1 or later if any changes are necessary.

The Buckets feature has been removed. It was very rarely used as it was unlikely to come up with a common reply to multiple tickets which couldn't instead be made into a Canned Reply which could then be reused on future tickets. If you were using buckets you need to note down which tickets you had in each bucket. You can tag the tickets after upgrading to ATS 5 and you can filter the display in the frontend by tag but you can no longer send a mass reply to all tickets in a bucket.

The sh404SEF plugin has been removed. This component is no longer developer for Joomla 4. Its replacement, 4SEO, does not need a plugin for Akeeba Ticket System. The URLs are generated by Joomla and ATS' SEF router helper.

We no longer ship an Installer plugin nor do we let you enter a Download ID in the component's Options. Joomla 4 manages Download Keys (what it calls the Download ID) in its Update Sites page.

We no longer provide a legacy Search plugin. The old Search component is now an optional and deprecated extension. We still provide a Smart Search (com_finder) plugin. Smart Search is the recommended and only supported search method in Joomla 4.

CLI scripts / CRON jobs

We no longer provide individual CLI scripts from CRON jobs. Joomla 4 has its own extensible CLI application, `cli/joomla.php` under your site's root. We provide a `console` plugin which adds ATS commands to Joomla's CLI application.

If you have CRON jobs using the CLI script to automate feature in Akeeba Ticket System you need to migrate them to the new Joomla CLI commands. If you were using a URL for your CRON jobs no change is necessary.

Previously optional features integrated into the component

Some features formerly included as plugins are now integrated within the component itself.

Gravatar integration is now built-in and the default fallback if no other source for a user profile image (avatar) is provided. The email address of the user submitting the ticket reply is used.

Displaying user groups is an option in the component's Options.

Sending emails (Professional version only) is integrated in the component and enabled by default. You can disable sending emails through an option in the component's Options.

Custom fields

We have removed our custom fields and our custom field plugins. Instead, we are integrating with Joomla fields, a feature introduced in Joomla 3.7. The values of the old custom fields in existing tickets **are not lost**. They are preserved. If you create Joomla fields for your tickets which have the same name as your old custom fields their contents will be automatically migrated when you next visit or edit the ticket.

Before migrating make a list of the field names, types and settings you had been using in Akeeba Ticket System 4.

After migrating to Akeeba Ticket System 5 create custom fields whose Name matches the Alias (slug) of the fields you had in Akeeba Ticket System 4.

If you were using the geolocation field type you cannot migrate its contents but you can replicate its functionality using a third party Joomla custom field plugin.

Share buttons

We have removed the shared buttons plugin. It had seen very limited use and was also very specific in the third party service it could use. Our software is now triggering the `onContentPrepare` event, the standard Joomla event which allows third party system and content plugins to add their own HTML into a component's output.

Please use a third party plugin to add share buttons. Alternatively you could use a third party module (e.g. BW Social Share [<https://extensions.joomla.org/extension/bw-social-share/>]) using our custom module positions.

User tags

ATS 1.x to 4.x inclusive had the option to set up user tags with the understanding that these are only visible to the support staff, not the users themselves. Due to the sensitive nature of these tags, when you are upgrading from ATS 4 to ATS 5 these tags are imported **unpublished** by default.

You should follow our instructions on making them visible only to the support staff before publishing them. You can publish tags from Joomla's Components, Tags page.

Kindly note that tags in Joomla are global, i.e. they will also appear in other Joomla components for users who have access to their view access level. Likewise, user tagging will also display tags you use in other Joomla components, such as core content (Articles). This is how Joomla works. There is really no way around it.

Changed options

Some of the preferences in the Options page have been removed or changed.

You no longer have options about loading the Akeeba FEF framework. Our Joomla 4 native software no longer uses Akeeba FEF. It uses Bootstrap 5 styling, this being the CSS framework Joomla 4 itself is using.

You no longer have options for Dark Mode for the same reason. If you want to implement Dark Mode on your site you can use a dark themed template for your site or the DarkMagic plugin [<https://github.com/nikosdion/DarkMagic/tree/joomla4>] if your site is using Bootstrap 5 in the frontend. Kindly note that this plugin is experimental and has only been tested with Cassiopeia, the default Joomla 4 template.

The Attachments Folder option must now include a path relative to your site's root for a directory which is under your site's root. You can no longer use a folder above your site's root. If you were using such a folder this option has changed to `media/com_ats/attachments`. Please copy your attachment files into that folder manually; it's not done automatically.

The CAPTCHA options have changed. Instead of an on/off switch for showing a CAPTCHA for Guest Tickets you can now select which of the enabled CAPTCHA plugins you want to use. Your old option is automatically migrated on upgrade. If it was off it will remain off by being set to None. If it was on it will be migrated to Use System Default which uses the CAPTCHA method defined in your site's Global Configuration. This emulates the behavior of ATS 2.x through 4.x inclusive.

The Custom Statuses have a new internal format and a new interface. Instead of a textbox with lines similar to `1=Some Label` you have a subform field presented as a table, the ID and the label being separate fields. The custom statuses are migrated automatically when you upgrade from ATS 4 to ATS 5. We recommend that you review them to make sure they are correct.

There are new options for attachments. On popular request, ATS no longer uses the same settings as Joomla's media manager. This allows you to define more lax upload permissions for Akeeba Ticket System such as different file types (e.g. you could allow users to upload ZIP files in ATS but not in the Joomla media manager), a different maximum file size etc. Please review these settings. The default settings in ATS 5 match the **default** settings in Joomla 4's media manager which may be substantially different than what you had already configured on your site.

Off-line schedule

We have removed the off-line schedule feature. The way it was working was very limited, especially if you wanted to create an offline schedule which crossed week boundaries (e.g. go off-line on Saturday afternoon and back online on Monday morning) or one which spanned more than a week e.g. an off-line schedule for the entire Christmas and New Year period, something very common in schools and other educational institutions.

Instead, we offer an integration with Joomla's CLI application which allows you to set up CRON jobs for automating the ticket system going on-line and off-line.

InstantSearch

The InstantSearch / InstantReply feature of Akeeba Ticket System has changed. In the past we had two plugins which allowed ATS to show the top 10 search results from ATS tickets and / or DocImport articles. This was cumbersome and didn't offer a realistic way to search an FAQ section / knowledge base.

ATS 5 now offers three options. Turning this feature off, using Joomla's Smart Search or using an external search engine. By default, the Smart Search method is enabled and searches your entire site. You can use custom, hidden Smart Search menu items to limit the scope of the search to specific sections of your site. Alternatively, you could use a third party search engine, be it a commercial search engine such as DuckDuckGo, Google etc or a self-hosted search engine such as Apache Solr.

One related feature removed is sending the results of InstantSearch by email when a new ticket is filed by email (InstantReply). This is no longer possible because ATS does not receive the search results in a machine-readable way; it simply displays the Joomla or third party search results in an IFRAME.

View template overrides

We no longer use Blade templates for our view templates. We use standard Joomla .php view template files.

The internal data structures have also changed. We no longer use Akeeba FOF as our backend framework, we use Joomla's MVC.

As a result you will need to rewrite any template overrides you may have made.

On the bright side, our default view templates use the standard Joomla 4 CSS framework (Bootstrap 5) which means that most sites will not need to do any customisation. The display of the ticket system will conform to the customisation you or your template developer has made on Bootstrap 5. You will only need to do template overrides if you are using a template with a different CSS framework.

Kindly note that we CAN NOT help you create template overrides.

4. Installing Akeeba Ticket System

Installing Akeeba Ticket System is no different than installing any other Joomla!™ extension on your site. You can read the complete instructions for installing Joomla!™ extensions on the official help page [https://help.joomla.org/proxy?keyref=Help40:Extensions:_Install&lang=en]. Throughout this chapter we assume that you are familiar with these instructions and we will try not to duplicate them.

4.1. Installing or manually updating the component

Just like with most Joomla! extensions there are two ways to install or manually update Akeeba Ticket System on your site:

- Install from URL. This works only with the Professional release of our component. It is the easiest and fastest one, if your server supports it. Most servers do support this method.
- Upload and install. That's the typical extension installation method for Joomla! extensions. It rarely fails.

Please note that installing and updating Akeeba Ticket System (and almost all Joomla! extensions) is actually the same thing. If you want to update Akeeba Ticket System please remember that you **MUST NOT** uninstall it before installing the new version! When you uninstall Akeeba Ticket System you will lose all your settings. This is definitely something you do not want to happen! Instead, simply install the new version on top of the old one. Joomla! will figure out that you are doing an update and will treat it as such, automatically.

Tip

If you find that after installing or updating Akeeba Ticket System it is missing some features or doesn't work, please try installing the same version a second time, without uninstalling the component. The reason is that very few times the Joomla! extensions installer infrastructure gets confused and fails to copy some files or

entire folders. By repeating the installation you force it to copy the missing files and folders, solving the problem.

Finally, please bear in mind that extension installation and updates are handled by Joomla itself. Akeeba Ltd does NOT have any control over the code which performs extension installation, discovers updates, downloads updates or installs updates.

4.1.1. Install from URL

The easiest way to install Akeeba Ticket System Professional is using the Install from URL feature in Joomla!

Important

This Joomla! feature requires that your server supports fopen() URL wrappers (`allow_url_fopen` is set to 1 in your server's `php.ini` file) or has the PHP cURL extension enabled. Moreover, if your server has a firewall, it has to allow TCP connections over ports 80 and 443 to `www.akeeba.com` and `cdn.akeeba.com`. If you don't see any updates or if they fail to download please ask your host to check that these conditions are met. If they are met but you still do not see the updates please file a bug report in the official Joomla! forum [<http://forum.joomla.org/>]. In the meantime you can use the manual update methods discussed further below this page.

First, go to our site's download page [<https://www.akeeba.com/download.html>]. Make sure you are logged in. Look for Akeeba Ticket System on that page. If you see two rows, for Joomla 3 and Joomla 4, make sure that you download the version listed under Joomla 4. Click on the Release notes and all files for version x.y.z link, where x.y.z is the latest version number for Akeeba Ticket System for Joomla 4. On that page you will find both Akeeba Ticket System Core and Professional. Next to the Professional edition's Download Now button you will see the Direct Install Link link. Right click on it and select Copy link address or whatever your browser calls this.

Now go to your site's administrator page and click on System. On that page find the Install card and click on the Extensions link. Click on the Install from URL tab. Clear the contents of the Install URL field and paste the URL you copied from our site's download page. Then click on the Install button. Joomla! will download and install the Akeeba Ticket System update.

If Joomla cannot download the package, please use the method described below. If, however you get an error about copying files, folder not found or a cryptic "-1" error please follow our installation troubleshooting instructions [<https://www.akeeba.com/documentation/troubleshooter/abinstallation.html>].

4.1.2. Upload and install.

You can download the latest installation packages our site's download page [<https://www.akeeba.com/download.html>]. If you see two rows, for Joomla 3 and Joomla 4, make sure that you download the version listed under Joomla 4.

If you have an older version of Joomla! or PHP please consult our Compatibility page [<https://www.akeeba.com/compatibility.html>] to find the version of Akeeba Ticket System compatible with your Joomla! and PHP versions.

In either case click on the version you want to download and install.

If you are a subscriber to the Professional release, please make sure that you have logged in first. You should then see an item on this page reading Akeeba Ticket System Professional. If you do not see it, please log out and log back in. Click on the Professional item to download the ZIP installation package.

All Akeeba Ticket System installation packages contain the component and all of its associated extensions. Installing it will install all of these items automatically. It can also be used to upgrade Akeeba Ticket System; just install it *without* uninstalling the previous release.

In any case, do not extract the ZIP files yet!

Attention macOS users! Safari, the default web server provided to you by Apple, is automatically extracting the ZIP file into a directory and removes the ZIP file. In order to install the extension through Joomla!'s extensions installer you must select that directory, right-click on it and select Compress to get a ZIP file of its contents. This behaviour was introduced in Mac OS X Mountain Lion. To disable this behavior go to Safari, press **CMD+**, (hold down Command, press comma), click on the General tab and UNCHECK the Open “safe” files after downloading check box.

Important

DO NOT TRY TO INSTALL THE ZIP FILES INCLUDED IN THE FILE SEPARATELY. THIS WILL CAUSE PROBLEMS WHEN UPGRADING JOOMLA TO A NEW VERSION FAMILY AND WILL MAKE IT IMPOSSIBLE FOR JOOMLA TO FIND UPDATES TO AKEEBA TICKET SYSTEM. Joomla needs to know which extensions belong together in a package. Updates are applied to the package, not each individual extension. By installing the extensions separately you do not provide this information to Joomla. As a result it will be unable to find update information for the package itself. On top of not being able to show you updates for Akeeba Ticket System it will also start complaining that your currently installed version of each of the Akeeba Ticket System extensions is incompatible with the new version of Joomla every time you try to upgrade to a new Joomla version family (e.g. 4.0 to 4.1 or 5.0). If this happens to you, download and install the latest PACKAGE of our software to restore the link between extensions and package, then wait for 2 days for Joomla's update information cache to expire. We will not provide any further support beyond this notice about this issue. Thank you for your understanding.

Log in to your site's administrator section. Click on System from the side menu. Find the Install card and click the Extensions link. Click on the Upload Package File tab. Click on the green Or browse for file button in the main area of the tab. Locate the installation ZIP file you had previously downloaded and select it. After a short while, Joomla!™ will tell you that the package has been installed.

If the installation did not work, please take a look at our installation troubleshooting instructions [<https://www.akeeba.com/documentation/troubleshooter/abinstallation.html>].

5. Upgrading from Core to Professional

Upgrading from Akeeba Ticket System Core to Akeeba Ticket System Professional is by no means different than installing the component. You do not have to uninstall the previous version; in fact, you **MUST NOT** do that.

Simply follow the installation instructions to install Akeeba Ticket System Professional over the existing Akeeba Ticket System Core installation. That's all! All your settings are preserved.

Important

When upgrading from Core to Professional you usually have to install the Professional package **twice**, without uninstalling anything in between. Sometimes Joomla! does not copy some of the files and folders the first time you install it. However, if you install the package again (without uninstalling your existing copy of Akeeba Ticket System) Joomla! copies all of the necessary files and performs the upgrade correctly.

6. Automatic updates

Akeeba Ticket System can be updated just like any other Joomla! extension, using the Joomla! extensions update feature. Please note that Joomla! is fully responsible for discovering available updates and installing them on your site. Akeeba Ltd does not have any control of the update process.

Note

This Joomla! feature requires that your server supports fopen() URL wrappers (`allow_url_fopen` is set to 1 in your server's `php.ini` file) or has the PHP cURL extension enabled. Moreover, if your

server has a firewall, it has to allow TCP connections over ports 80 and 443 to `www.akeeba.com` and `cdn.akeeba.com`. If you don't see any updates or if they fail to download please ask your host to check that these conditions are met. If they are met but you still do not see the updates please file a bug report in the official Joomla! forum [<http://forum.joomla.org/>]. In the meantime you can use the manual update methods discussed further below this page.

Warning

Akeeba Ticket System Professional needs you to set up the Download ID before you can install the updates. Please consult the Entering your Download ID documentation section for more information.

You can access the extensions update feature in two different ways:

- From the icon your Joomla! administrator control panel page. By default you will find the icon in the right-hand modules area, under the Update Checks header. When there are updates found for any of your extensions you will see the Updates are available message. Clicking on it will get you to the Update page of Joomla! Extensions Manager.
- From the sidebar of your Joomla! administrator click on System. On the new page find the Update area towards the bottom of the middle column and click the Extensions link. This takes you to the Update page of Joomla! Extensions Manager.

If you do not see the updates try clicking on the Find Updates button in the toolbar. If you do not see the updates still you may want to wait up to 24 hours before retrying. This has to do with the way the update CDN works and how Joomla! caches the update information.

If there is an update available for Akeeba Ticket System tick the box to the left of its row and then click on the Update button in the toolbar. Joomla! will now download and install the update.

If Joomla! cannot download the package, please use one of the manual update methods described below. If, however you get an error about copying files, folder not found or a cryptic "-1" error please follow our installation troubleshooting instructions [<https://www.akeeba.com/documentation/troubleshooter/abinstallation.html>].

Note

If you are using Akeeba Ticket System Professional you must have a valid subscription which includes this software to download and install the update. If you get a 403 error and a message to visit a URL that includes the text `&dlid=` in it please check that you have a valid subscription on our site, you have entered the Download ID in the Update Sites page of Joomla and that the Download ID you have entered matches the one listed in the Add-on Download IDs page of our site when you are logged into our site. If all this information is correct please clear your Joomla cache and try clicking on the Find Updates button again in Joomla's Updates page; otherwise Joomla may not "see" the Download ID correctly.

If you get a white page while installing the update please try either the Built-in method (described above) or the manual update method (described below).

Updating manually

As noted in the installation section, installing and updating Akeeba Ticket System is actually the same thing. If the automatic update using Joomla!'s extensions update feature does not work, please install the update manually following the instructions in the installation section of this documentation.

When installing an update manually you **MUST NOT** uninstall your existing version of Akeeba Ticket System. Uninstalling Akeeba Ticket System will always remove all your settings. You definitely not want that to happen!

Sometimes Joomla! may forget to copy some files when updating extensions. If you find Akeeba Ticket System suddenly not working or if you get a warning that your installation is corrupt you need to download the latest version's

ZIP file and install it *twice* on your site, *without* uninstalling it before or in-between these installations. This will most certainly fix this issue.

If the error occurs again after a while, without you updating our software, please contact your host. Some hosts will delete or rename files automatically and without any confirmation as part of a (broken and unfit for purpose) "malware scanner / antivirus". Unfortunately, these scanners return a lot of false positives -innocent files mistakenly marked as malicious- but rename / delete them nonetheless, breaking software installed on the server. If you are on such a host we very strongly recommend that you move to a decent host, run by people who actually know what they are doing. It will be far less headache for you and would actually improve your site's security.

6.1. Troubleshooting the update

Like most Joomla extensions, our software relies on Joomla's built-in extensions updater. In simple terms, code written by the Joomla project, shipped with Joomla itself and running on your site is responsible for retrieving information about the latest available versions, determining whether an update is available, downloading the update package and installing it on your site. Akeeba Ltd has no control over that code.

Despite this not being our code, we do understand that our clients do come across problems with updates and need our help. The way the Joomla built-in extensions updater is written makes it prone to some easily preventable, common errors. Its error reporting ranges from unhelpful to non-existent. In an effort to help you, we've compiled and condensed all the troubleshooting we've done for years on our sites and our clients' sites.

We do understand that the instructions in this section may feel convoluted and complicated. We are afraid that this is the *simplest* form they can be reduced to. We'd like to assure you that they do not reflect on the quality of software we can produce. These instructions are necessitated by and reflect the way Joomla itself works.

If you were using our software between 2009 and 2016 you might remember that we were using our own code, Akeeba Live Update, to find, download and install updates to our software. It was a much better solution and much easier to troubleshoot; it would tell you exactly what to fix and ask you to click a button to let it verify the fix. Unfortunately, the Joomla Extensions Directory (JED) made it impossible to use our own updater code, even as an alternative to the Joomla built-in extensions updater, in 2016. Doing otherwise is a violation of JED's terms of service and would result in all of our extensions being unlisted. That's why we are now using the Joomla built-in extensions updater even though we know it's nowhere near as good as what we used to have. Sorry, folks.

6.1.1. Addressing server issues

In some cases you will see that Joomla cannot retrieve the latest version information or update package for our software, reporting it cannot connect to `cdn.akeeba.com`. Related to that, Joomla may report that it's unable to download the Professional edition's update package, saying it's unable to connect to our site `www.akeeba.com`. This can mean a few different things which all have to do with how your host is set up.

Our CDN and our site are accessible over HTTPS and use a valid, signed TLS certificate. At the time of this writing the TLS certificates are issued by Let's Encrypt and Amazon Web Services. The TLS certificates used for HTTPS on our CDN and site use the recommended SHA-256 hashing algorithm and the servers only support modern versions of the TLS protocol (at the time of this writing it's TLS 1.2 and later). If your host has an out of date Certification Authority cache or compiled PHP against an old TLS library which does not support modern versions of TLS your site will be unable to connect to our servers.

If this is not the case, please be aware that some hosts run a proxy server or a firewall which can either prevent or cache *outgoing* connections in front of their servers. Depending on how this is implemented it can cause two distinct types of problems.

The first problem is that your site might be unable to connect to our CDN and our server to retrieve the latest version information and the update package itself respectively. If this happens you need to ask your host to allow connections to TCP/IP port 443 (HTTPS) for `www.akeeba.com` and `cdn.akeeba.com`. If they ask you for an IP address please ask

them to resolve these domain names from their server. Both are served by a Content Delivery network (CDN) with hundreds of servers, the former powered by CloudFlare and the latter powered by Amazon CloudFront, meaning that its IP address depends on where you are accessing it from.

The second problem is that when Joomla tries to retrieve the latest version information or an update file from our servers your host's proxy gets in the way and returns information it has cached. We explicitly ask for that information not to be cached, using standard HTTP headers, but some hosts choose to ignore web standards and do their own thing. Also worth noting is that your host should not interfering with HTTPS (encrypted) traffic, so all the more reason to be worried about their implementation in this case. Unfortunately, we have caught a few hosts doing that over the years.

None of these issues can be addressed by you or us. You will need to contact your host about them. Before you assume any of these issues are in play and if you are using the Professional edition of our software please do check that your Download ID is valid first.

6.1.2. Check the validity of your Download ID

Note

The information in this section only applies to the Professional edition. If you are using the Core edition you can skip over it.

If you are using the Professional version of our software we need to verify that you have an active subscription that gives you access to downloads of the software you are trying to update. We do that by means of a Download ID which has the format 0123456789abcdef0123456789abcdef (Main Download ID) or 12345:0123456789abcdef0123456789abcdef (Add-on Download ID). In and by itself the Download ID does not carry any information about your subscription status. It is an identifier linked to your account on our site.

First, you need to check that you are using a valid Download ID. **Do not assume that your Download ID** is entered at all, or that it is valid. This kind of false assumption accounts for half of the update issues we are asked to help our clients with. Always check on our site.

Log into our site and go to Add-on Download IDs from the top menu. Copy the Download ID. Go to your site, click on System from the side menu. Find the Update card and click on the Update Sites link. In there search for **Akeeba Ticket System**. You can see if there's a Download Key entered and what it's that (Download Key is what Joomla calls our Download ID). If none is entered or the wrong one is entered click on the update site's name and paste the Download ID into the Download Key box, then click on Save & Close in the toolbar. If there were more than one update sites with Akeeba Ticket System in their name repeat this process for each one of them.

If you had to enter or change the Download ID but Joomla was already reporting an updated version you need to follow the instructions in the Entering or changing your Download ID after an update is available section. Otherwise installing the update will NOT work at all due to the way Joomla caches update information in its database.

6.1.2.1. Check your subscription status

As noted above, the Download ID itself does not carry any information about whether you are allowed to download an update. This check is done on our server when it receives the Download ID along with Joomla's request to download an update. The check performed is simple: do you have an *active* subscription which gives you access to the software you are trying to download?

Do not assume that your subscription is active. It is possible that you missed an email warning you about the subscription expiring and a manual action to renew it being required on your part.

Always log into our site and go to the My Subscriptions page to check your subscription status. If your subscription has expired you can renew it. Once the payment is complete and accepted by our reseller you will be able to download the updates within the next 20' or less (typically: within seconds).

6.1.2.2. Multiple Professional edition Akeeba extensions with different Download IDs

This should not be a problem in Joomla 4 since we are using Joomla's built-in support for Download Keys instead of plugins to apply the Download ID.

However, if you were using older versions of our software on your site there might be some plugins left behind which do not allow each one of our software to have its own Download ID.

Why you might need that? It's possible that you have more than one of our Professional extensions but you want to use different Download IDs for each one of them. For example, if each extension was bought by a different company working on your site or if you are trying to migrate to a new user account on our site.

In this case you need to unpublish the following plugins on your site:

- Installer - Akeeba Backup Professional
- Installer - Akeeba Ticket System Professional
- Installer - Admin Tools Professional

If these plugins do not exist: that's perfectly fine! These plugins are no longer distributed with our Joomla 4-only software.

Do remember to check the Download IDs for each and every of our extensions.

6.1.2.3. Entering or changing your Download ID after an update is available

If you had to enter or change a Download ID after Joomla finds an update for it you will need to jump through a few hoops. This happens because of the way Joomla caches update information and some design choices in the Joomla extensions updater which prevent it from working smoothly in this case.

Please follow all of the steps below in the exact order presented from a single browser tab without having any other tabs or windows to your site open. Please follow all of the steps even if you think that something is redundant or repetitive; it's not and there is a reason we tell you to do it. We understand that the process outlined below is gruelling. Unfortunately it is necessary because of how Joomla itself works. This code is NOT under our control. If it was our code you wouldn't have to do anything at all; we know how to write software with better UX than this...

1. Go to the backend of your site.
2. Go to the System menu item, find the Update section and click on Update Sites.
3. Click on the update site for our software, e.g. Akeeba Backup Professional, Akeeba Ticket System Professional or Admin Tools Professional depending on which of our software you have.
4. Enter your Download ID in the Download Key box if not already entered or if it's not valid.
5. Click on Save & Close.
6. If you have more than one Professional version of our software installed repeat the process from the beginning until here for each one of: Akeeba Backup Professional, Admin Tools Professional, Akeeba Ticket System Professional.
7. Go to the System menu item, find the Update section and click on Extensions.
8. Click on Check for Updates.
9. Go to the System menu item, find the Maintenance area, click the Clear Cache link.
10. Click on the Delete All button even if the list is empty.

11.Go to the System menu item, find the Update area, click the Extensions link.

12.Click on the Check for Updates button in the toolbar again.

You should now be able to perform the updates to our software

6.1.3. Updates are showing after installing the latest version

Sometimes you might see that Joomla reports that the version you have installed or even a previous version is available as an update. This can mean three things:

- Joomla's update cache is stuck. See the information on resetting Joomla's update cache.
- You have a server issue connecting to our CDN. See the information on addressing server issues.
- You have found a bug in Joomla's built-in extensions updater. You need to contact the Joomla! forum [<https://forum.joomla.org>]. Unfortunately there is nothing we can do about Joomla core bugs.

6.1.4. Updates not showing despite having an older version

Sometimes you may see that Joomla refuses to report the availability of a new version of our software. This can mean three things:

- The update site for our software is disabled. See the information on checking the update site.
- Joomla's update cache is stuck. See the information on resetting Joomla's update cache.
- You have a server issue connecting to our CDN. See the information on addressing server issues.
- You have found a bug in Joomla's built-in extensions updater. You need to contact the Joomla! forum [<https://forum.joomla.org>]. Unfortunately there is nothing we can do about Joomla core bugs.

6.1.4.1. Check the update site

First we are going to check if the Update Site is disabled. Go to the backend of your site. Go to the System menu item, find the Update area and click on the Update Sites link.

On that page you will see a list of the update sites for the extensions you have installed on your site. If you see our software in that list – you may have to search for it – make sure it's published, i.e. there's a green checkmark in the Status column. If it's not already published publish it now. If you had to publish the Update Site you also need to follow the instructions under Refresh the update cache for your updates to work.

If our software does not appear on that list you will need to click on Rebuild. Watch out! Using Rebuild **removes** all Download IDs and Download Keys from all installed extensions on your site and will cause extension installation to fail. You will also need to follow the instructions under Check the validity of your Download ID for your updates to work.

6.1.4.2. Refresh the update cache

Joomla does not download the latest version information every time you ask it to tell you if there's a new version available for an extension. This would be too slow and bog down the servers of the third party developers providing this update information. Instead, it caches the updates for 1 to 24 hours (configurable), with the default being 6 hours. In very rare cases this updates cache may get “stuck” beyond this time limit and needs to be manually refreshed.

First try the normal way to get the update cache refreshed. Please follow all of the steps below in the exact order presented from a single browser tab without having any other tabs or windows to your site open. Please follow all of the steps even if you think that something is redundant; it's not and there is a reason we tell you to do it.

1. Go to the backend of your site.
2. Go to the System menu item, find the Update area, click the Extensions link.
3. Click on the Check for Updates button in the toolbar.

This will tell Joomla to get the up-to-date information about available versions for all extensions installed on your computer. If you do not see anything changing to the better you have hit a rare issue which involves another Joomla cache that is invisible even to the Super Users: the Joomla database query cache. Please follow all of the steps below in the exact order presented from a single browser tab without having any other tabs or windows to your site open. Please follow all of the steps even if you think that something is redundant or repetitive; it's not and there is a reason we tell you to do it.

1. Go to the backend of your site.
2. Go to the System menu item, find the Update area, click the Extensions link.
3. Click on the Check for Updates button in the toolbar.
4. Go to the System menu item, find the Maintenance area, click the Clear Cache link.
5. Click on the Delete All button even if the list is empty.
6. Go to the System menu item, find the Update area, click the Extensions link.
7. Click on the Check for Updates button in the toolbar again.

If you still cannot retrieve updates for our software you need to check if you have a server issue. If that's not the case you need to check the update site since it might have been automatically unpublished by Joomla.

6.1.5. Miscellaneous troubleshooting and information

6.1.5.1. The update fails to download

If you are trying to update a Professional edition please check your Download ID. Typically you will get an error message telling you that an error 403 or 500 was received when trying to download the update package. Whether you see that message or a generic download failure message depends on the version of Joomla you have installed on your site.

If this doesn't help you need to check if you have a server issue.

6.1.5.2. Updating with a third party service fails

Typically, third party site management services ask Joomla to provide the update information and install update on your behalf. Therefore the troubleshooting information in this section would solve both in-site and remote (via a service) extension updates.

If you can install an update by logging into your site's backend but NOT through a service you need to contact the third party site management service and report this issue. Unfortunately we cannot help with it. Third party services DO NOT ask us for permissions to implement an updater for our software.

6.1.5.3. Manual update

As noted earlier in the documentation, a manual update is the same as installing the extension. Download the latest version from our site and install it on your site **without** uninstalling our extension.

6.1.5.4. Update installation problems

If your update does download but fails to install try the manual update method (installing the new version on top of the old one). If that fails, too, you should follow the instructions on the installation troubleshooting section you can read earlier in this documentation.

7. Entering your Download ID

Note

If you are using Akeeba Ticket System Core, the free of charge edition of Akeeba Ticket System, you do not need to and must not enter a Download ID. The Download ID is only required for the for-a-fee Akeeba Ticket System Professional edition.

Akeeba Ticket System Professional is the for-a-fee edition of Akeeba Ticket System with additional features. Downloading it, either for installation from scratch or as an update to an already installed but older version on your site, requires confirming that you have an active subscription which gives you access to Akeeba Ticket System Professional downloads. When you download the installation ZIP file from our site this means that you need to log in to our site first. However, when downloading updates through Joomla you really don't want to and usually cannot be asked to log in to our site.

Using your Download IDs on your clients' sites

Our software license allows you to use your Download IDs on the sites of your clients. However, you must tell your clients that:

- Downloads and support for the software covered by the Download ID is provided by you, not Akeeba Ltd.
- If they want to receive support and / or downloads directly from Akeeba Ltd they need to purchase a qualifying subscription on our site. In this case they do not qualify for the renewal discount.
- They are not allowed to use the Download ID on any other site or use the Download ID to download the software for any reason other than updating or reinstalling the covered software on the same site the Download ID was entered in. In other words, they cannot use the Download ID to install or update our software on any other site.

If you are no longer administering a site where you have entered a Download ID you must revoke or regenerate that Download ID. You need to do the same if you believe that your Download ID is being used by third parties in an unauthorized manner. Please note that unauthorized use of Download IDs could have consequences on your subscription with us.

Finding your Download ID

Download IDs come in two flavors, your main Download ID and Add-on Download IDs.

You can find your main Download ID in the My Subscriptions [<https://www.akeeba.com/my-subscriptions.html>] page of our site. We recommend using this Download ID only on your own site(s). This Download ID cannot be revoked, it can only be regenerated. If it's regenerated you will need to enter the new Download ID on all of your sites which can be a significant hassle.

You can generate an unlimited number of Add-on Download IDs without additional charge in the Add-on Download IDs [<https://www.akeeba.com/download/add-on-dlid.html>] page. Unlike the main Download ID you can revoke (disable) any Add-on Download ID at any time. As long as you only use one Add-on Download ID per site revoking or regenerating it will not affect the other sites' ability to download and install updates.

Enter or view your Download ID

From the main administrator page of your site click on System on the sidebar.

Click on the Update Sites link towards the bottom of the middle column on the System page.

Find the Akeeba Ticket System Professional entry on the list and click on it to open the edit page.

Enter your main or Add-on Download ID in the Download Key area. Click on the Save & Close button on the toolbar to apply the Download ID.

If Akeeba Ticket System or Joomla! was already showing you that an update for Akeeba Ticket System is available please follow the steps under "Further steps if a download was available before entering a new Download ID" below.

Further steps if a download was available before entering a new Download ID

Joomla applies the Download ID when it *is looking for* updates, not when it tries to download updates. Unfortunately, this means that entering a new Download ID after Joomla shows that updates are available does NOT apply the new Download ID immediately and causes updates to fail. You need to follow all of the steps outlined below, in this specific order, without doing anything else on your site's administrator area in between to let Joomla see the new Download ID. Please do not file support tickets or bug reports about this with us; extension updates are managed by Joomla core code and we have no influence over it.

- Go to Joomla's extension update page. From the sidebar of your Joomla! administrator click on System. On the new page find the Update area towards the bottom of the middle column and click the Extensions link.
- Click on the Check For Updates button in the toolbar.
- Next up, we need to clear the Joomla administrator cache. This is required because Joomla caches the database queries which retrieve the Download ID in its administrator cache. If you do not clear the cache it won't see the new Download ID.

From the sidebar of your Joomla! administrator click on System. On the new page find the Maintenance area towards the top of the left column and click the Clear Cache link. Click on Delete All in the toolbar.

- Go back to Joomla's extension update page. From the sidebar of your Joomla! administrator click on System. On the new page find the Update area towards the bottom of the middle column and click the Extensions link.
- Click on the Check For Updates button in the toolbar again, even if you already see updates.
- You should now see the update to our software and be able to install it.

Troubleshooting updates to the Professional release

If you still cannot install our software please check that the Download ID is entered correctly. If it's not entered correctly enter the correct Download ID and follow all of these instructions again.

If the Download ID is entered correctly but it's not active in the Add-on Download IDs page you will need to enable it. After enabling it you will be able to download and install the update *without* having to follow these instructions again.

If the Download ID is correct please make sure that you have an *active* qualifying subscription on our site. If your subscription has expired you need to purchase a renewal on our site. Once the renewal is active you will be able to

download and install the update *without* having to follow these instructions again, as long as you have not changed your Download ID.

If you still cannot download updates despite having the correct Download ID and an active subscription try waiting for 24 to 48 hours. In very rare cases Joomla's update cache gets stuck despite following the instructions above and you just need to wait until Joomla decides it has to reload it.

If the updates are still not downloading please make sure that you are using a version of Joomla and PHP that is supported by the new version of our software. If you are not sure please consult our Compatibility page [<https://www.akeeba.com/compatibility.html>].

If you've followed all these troubleshooting steps and the update is not downloading at all you need to contact your host and ask them to allow traffic to www.akeeba.com and cdn.akeeba.com over port TCP 443 (HTTPS), make sure that the PHP cURL module is installed and activated on the version of PHP your site is using and that finally the libcurl and libssl system libraries the cURL module is compiled against are up-to-date versions. If your host cannot help you with any of these requests (despite this being literally what you are paying them to do) you can install updates manually. Kindly note that Akeeba Ltd is not responsible for your hosting environment and that the requirements for downloading updates from our site are met by server software released around 2015. If your host cannot provide at the very least several years old software and open ports in their firewall you should probably be migrating your site to a more up-to-date, competent host.

8. Requesting support and reporting bugs

Support is provided only to subscribers. If you already have an active subscription which gives you access to the support for Akeeba Ticket System you can request support for it through our site. You will need to log in to our site and go to Support, Akeeba Ticket System and click on the New Ticket button. If you can't see the button please use the Contact Us page to let us know of the ticket system problem and remember to tell us your username.

(Yes, we are using Akeeba Ticket System for the Support section of our site. We eat our own dogfood, as the saying goes.)

If you want to report a bug, please use the Contact Us page of our site. You don't need to be a subscriber to report a bug. Please note that unsolicited support requests sent through the Contact Us page will not be addressed. If you believe you are reporting a bug please indicate so in the contact form.

Important

Support cannot be provided over Twitter, Facebook, email, Skype, telephone, the official Joomla! forum, our Contact Us page or any other method except the Support section on our site. We also cannot take bug reports over any other medium except the Contact Us page and the Support section on our site. Support is not provided to non-subscribers; if you are using the Core version you can request support from other users in the official Joomla! forum or any other Joomla!-related forum in your country/region. We have to impose those restrictions in support to ensure a high level of service and quality. Thank you for your understanding.

9. Uninstallation

Akeeba Ticket System can be uninstalled just like any other Joomla extension.

Warning

Uninstalling Akeeba Ticket System will delete your tickets and attachments. This process is **IRREVERSIBLE**. If you lose your support tickets by uninstalling Akeeba Ticket System we cannot help you retrieve them, they are gone forever.

Uninstalling Akeeba Ticket System will NOT remove the categories you create for the support ticket system. This is a limitation of Joomla's category management system. You need to delete the categories yourself before uninstalling Akeeba Ticket System.

First, go to the extensions manager page. From the sidebar of your Joomla! administrator click on System. On the new page find the Manage area towards the top of the middle column and click the Extensions link.

In the Search box type Akeeba Ticket System package. It will show you a single item called "Akeeba Ticket System package" whose Type is Package.

Important

Only ever try to uninstall the Package type extension. DO NOT try to uninstall the component, its plugins or module individually. It will leave stuff behind.

Select the item's checkbox and click on the Uninstall button in the toolbar. The extension and all its dependencies will be automatically uninstalled.

Chapter 2. Overview and Quick Setup

1. Overview

Note

Some of the features described in this Overview and Quick Setup chapter may only apply to the for-a-fee Akeeba Ticket System Professional edition.

Akeeba Ticket System (ATS) is a support ticket system component. It allows users of your site to submit support tickets. Typically it's used to provide support for software and hardware sales, but it can also be used for a plethora of other uses such as providing paid expert opinions (consultancy), answering pre-sales requests in any business sector (even in retail) and so on. The key difference to a forum is who gets to view the request and who gets to reply. With a typical forum everyone with access to a forum category can see and reply to all posts. With a ticket system only the user who filed the ticket can view it, unless it is public (public tickets are readable by everyone). In any case, only the user who filed the ticket and the designated support staff on your site can reply to tickets. This allows for finer control than a forum when an authoritative reply is required to the request.

In fact, we developed Akeeba Ticket System because of the limitations we kept on bumping on with several forum systems. The major limitation was that everyone could reply to a thread. This had the usual outcome that User A was talking about Problem X and, on the same thread, User B was talking about Problem Y. This had historically led to great deals of frustration for users and support staff alike. Akeeba Ticket System solves this by incorporating the concept of ticket ownership (the user who submitted the ticket get posting access to it, other users who are not member of the support staff don't).

In order for this to work, ATS is deeply integrated with Joomla! ACLs (access control lists). This is a very powerful feature and requires a little bit of reading to get it right. If your ACL setup is flawed, ATS will behave as told which would be completely different than what you intend. We have assembled a short list of very useful, free documentation on Joomla! ACL which will help you understand how the Joomla! ACL system works and how to make the best possible setup:

- ACL concepts overview [<https://magazine.joomla.org/all-issues/january-2012/joomla-1-6-1-7-and-2-5-acl-concepts-overview>] (beginners)
- Joomla! ACL: Access Levels [<https://magazine.joomla.org/all-issues/issue-february-2012/Joomla-ACL-Access-Levels>] (beginners; scroll all the way down for a very good video)
- A case for role-based ACL [<https://magazine.joomla.org/issues/Issue-Aug-2012/item/825-A-Case-for-Role-Based-ACL>] (advanced)
- Implementing role-based ACL [<https://magazine.joomla.org/issues/Issue-Sept-2012/item/856-Implementing-Role-Based-ACL>] (advanced)
- PWT ACL (formerly known as ACL Manager) [<https://extensions.joomla.org/extension/acl-manager/>] is a third party commercial component which can help you effectively managing ACLs on complex sites.

Continue reading on for a sample, albeit very simple, setup of a ticket system.

2. Quick Setup

Note

Some of the features described in this Overview and Quick Setup chapter may only apply to the for-a-fee Akeeba Ticket System Professional edition.

Before beginning please make sure that you have read the information and related links about Joomla access control. In this chapter we will implement a site using the following specifications:

- We will have two ticket categories. Pre-Sales Requests will be accessible to all users which have a user account on your site. Customer Support will only be available to users who belong in the Customers group.
- Support in the Pre-Sales Requests category will be provided by Super Administrators and users in the Sales Staff group.
- Support in the Customer Support category will be provided by Super Administrators and users in the Tech Support group.
- We will create a single menu item for the support ticket system which will be accessible by all registered users. Naturally, only the customers will see the Customer Support category thanks to Joomla's access control.

In order to implement these specifications we will have to create Joomla user groups, Joomla viewing access levels and ATS categories.

We will begin by creating the Joomla! User Groups requires for all of the above. Go to Users, Groups in the back-end of your site and click on New. Create three user groups:

- Customers with parent group Public
- Sales Staff with parent group Public
- Tech Support with parent group Public

Please take care to select the correct parent group! Using the Public group as the parent decouples the Akeeba Ticket System privileges from the account type. This is very important!

Then go to Users, Access Levels. You have to create one new viewing access level:

- Customer Support Access, selecting the user groups Customers and Tech Support.

The idea is that if you are a customer or a member of the tech support team you will be granted access to the Customer Support ticket category.

Now go to Components, Akeeba Ticket System, Categories. You will need to create two categories.

The first category has a title of Pre-Sales Requests. Find the Access drop-down and set it to Registered. In the Category Permissions tab click on the Registered group and give it the following permissions:

- Support Staff: Inherited
- Create: Allowed - Enables the users to submit new tickets
- Delete: Inherited
- Edit: Inherited
- Edit State: Inherited
- Create Private: Allowed - Enables the users to submit private tickets
- Create Attachment: Allowed - Enables the users to submit attachments (files) with their tickets

DO NOT set any permission to Denied. If a user belongs to a group where a permission is Denied or that has a group in any parent level that has this permissions Denied then you can never, ever give that permission to this user. This

is the most common mistake. If you want to deny a permission there's a 99% chance that the Inherited option is what you really need.

Then click on the Sales staff group and give them the following permissions:

- Support Staff: Allowed - Makes them managers of this Akeeba Ticket System category, enabling special actions and notifications on new posts
- Create: Allowed
- Delete: Allowed - Allows the managers to delete tickets and posts
- Edit: Allowed - Allows the managers to edit posts
- Edit State: Allowed - Allows the managers to publish/unpublish posts and entire tickets
- Create Private: Allowed
- Create Attachment: Allowed

Save this category and create a new category with a title of Customer Support. Find the Access drop-down and set it to Customer Support Access. In the Category Permissions tab click on the Customers group and give it the following permissions:

- Support Staff: Inherited
- Create: Allowed - Enables the users to submit new tickets
- Delete: Inherited
- Edit: Inherited
- Edit State: Inherited
- Create Private: Allowed - Enables the users to submit private tickets
- Create Attachment: Allowed - Enables the users to submit attachments (files) with their tickets

DO NOT set any permission to Denied. If a user belongs to a group where a permission is Denied or that has a group in any parent level that has this permissions Denied then you can never, ever give that permission to this user. This is the most common mistake. If you want to deny a permission there's a 99% chance that the Inherited option is what you really need.

Then click on the Tech Support group and give them the following permissions:

- Support Staff: Allowed - Makes them managers of this Akeeba Ticket System category, enabling special actions and notifications on new posts
- Create: Allowed
- Delete: Allowed - Allows the managers to delete tickets and posts
- Edit: Allowed - Allows the managers to edit posts
- Edit State: Allowed - Allows the managers to publish/unpublish posts and entire tickets
- Create Private: Allowed
- Create Attachment: Allowed

Do note that now the Support Staff doesn't have access to the Customer Support category and the Tech Support users don't have access to the Pre-Sales requests.

Now let's create the menu item. Go to the Menus menu and select the menu you want to create a link to Akeeba Ticket System. Create a new menu item of the type Akeeba Ticket System, Categories. Make sure the Access is set to Registered.

That's it! You now have to assign users to groups and you're ready to start offering support services through your site:

- Add your customers to the Customers group.
- Add the support staff to answer pre-sales requests to the Sales Staff group.
- Add the support staff to answer customer support requests to the Tech Support group.

A user may belong to many groups at once. For example if Alice needs to be able to answer both pre-sales and customer support requests she must be assigned to both the Sales Staff and Tech Support groups.

Chapter 3. Deep dive into advanced features

In this chapter we will discuss how Akeeba Ticket System integrates with the more advanced Joomla core features and how you can use them to accomplish advanced features without much anguish. In other words, this chapter documents the non-obvious, awesome things you can do with Akeeba Ticket System and core Joomla 4 features.

It may sound counter-intuitive to list these features before the main component documentation. There is a good reason. The main component is fairly straightforward to use, even without documentation. Ticket categories management is handled by Joomla itself, giving it a familiar experience. Ticket management and even features like canned replies and automatic replies are mostly self-documenting; at worst, a little prodding by a curious user is enough to make them work. The features we discuss here require combining a few core Joomla features and ATS features to implement something which is not immediately obvious to the casual observer.

1. Permissions

Permissions in Joomla define what a user group can and cannot do. Joomla has excellent documentation on how Permissions work [https://docs.joomla.org/J3.x:Access_Control_List_Tutorial].

Here are some quick takeaways:

- Permissions control what users do, not what they see. If you are interested in who can see what this is what Access Levels are for.
- Permissions cascade in two directions: 1. user groups, from Public towards the specific user group; and 2. from the Component and into the Categories tree.
- Permissions have one of three states: Inherited, Allow and Deny. Inherited is a “transparent” setting; it falls back to what is inherited from a previous cascade. If nothing is found it's treated as a Deny. Allow means that this permission is explicitly allowed. Deny means that this permission is explicitly denied.
- Deny trumps Allow and Inherited. If anywhere within the two permissions cascades there's an explicit Deny then the permission is Denied even if there is a more specific Allow. Think of permissions as voting with veto system. If you are not explicitly given the votes to do something you cannot do it (Inherited without explicit Allow). If anyone uses their veto power (explicit Deny) you are not allowed to do it.

Akeeba Ticket System uses Permissions in the component and each of its categories to control what the user can and cannot do. While most of them are standard Joomla permissions some are specific to Akeeba Ticket System or have a particular meaning. We are going to go through all of them.

Configure ACL & Options	Gives access to the component's Options page and the Permissions tab. This is equivalent to giving someone full and unfettered access to the component since they can change the permissions which apply to them!
-------------------------	---

Support Staff	Is the user a Support Staff / Manager? These users have additional implied privileges. They can reply to tickets submitted by any user. They can close and reopen tickets. They can view private tickets submitted by any user. They can view and edit the ticket metadata (title, state, ...) and custom fields. They can assign and be assigned to tickets. They can submit and view Manager's Notes on each ticket.
---------------	--

Only give this permission to your support personnel who's authorised to answer and manage tickets.

Create	Allows the creation of new tickets.
Delete	Allows deleting any ticket, post and attachment regardless of who submitted it.
Edit	Allows editing any ticket, post and manager's note regardless of who submitted it.
Edit Own	Allows a user to edit the ticket metadata and custom fields of ticket submitted by them, as well as any and every post submitted by them.
Edit Own Ticket	Allows a user to edit the ticket metadata and custom fields of ticket submitted by them but NOT their own posts.
Edit Own Post	Allows a user to edit every post submitted by them, but NOT the ticket metadata and custom fields.
Edit State	Allows the user to change the ticket state (Open, Closed, Pending and any custom state). This is NOT required for clients to close their tickets. This permissions would, in fact, allow them to re-open their ticket. Moreover, this allows the user to publish / unpublish tickets, posts and attachments.
Create Private	Allows the user to create a Private ticket. Private tickets are only visible to the user who submitted them and users with the Support Staff permission. The user also needs the Create privilege. If the Create privilege is given but not the Create Private one, then the user can only submit Public tickets which are visible by everybody.
Create Attachment	Allows the user to upload one or more attachments with their ticket and ticket replies. It requires the Create privilege.

2. Configuring Joomla's WYSIWYG editor (TinyMCE)

Joomla's WYSIWYG (What You See Is What You Get) editor a.k.a. the visual editor is implemented by the Editor - TinyMCE plugin. You probably knew that. What you probably *didn't* know is that it has profiles which are activated automatically per user group. In other words, you can change the amount of functionality exposed by the TinyMCE editor depending on who is logged into your site!

This is very useful for presenting a simple, intuitive and functional editor to your clients filing tickets.

It is recommended that you create a configuration set for the user groups your clients belong in.

Go to System, Manage, Plugins.

Edit the plugin Editor - TinyMCE.

Go to the Advanced tab and set the Number of Sets option to 4.

Click on Save.

Click on the Set 3 side tab. In the Assign this Set to field clear all of its contents and select the user groups your clients belong in.

Right above the sample toolbar click on Use simple preset. By default this only adds the following buttons:

- Bold

- Underline
- Strikethrough
- Undo
- Redo
- Unordered list
- Ordered list
- Clear formatting

You might want to also add the Add link and Remove Link buttons by dragging them into the toolbar from the All available menus and buttons area at the top of the plugin configuration page.

Do NOT add the Insert/edit image button! This is a very inconvenient interface which requires the user to type in an image URL. Instead, keep reading the next section of this documentation for a very convenient and simple way to add in-line images without exposing your entire Media Manager gallery to your users.

Also note that you do not see a very familiar button: the CMS Content dropdown. This is added automatically by Joomla itself if there are any buttons to show — like the Media icon which allows users to insert images and other media files in the body text.

2.1. In-line images

The most common use case for attachments is your users attaching screenshots or other image data necessary to clarify their point.

Images as attachments are perfectly possible but inconvenient. The support person has to download them to their computer first to inspect them. It is more convenient if the client can upload an image and display it in-line, in the body of their ticket text or ticket reply.

Unfortunately, Joomla's default use case is that images are only meant to be uploaded by people with the authority to create content (articles) on the site, i.e. Authors, Editors, Publishers, Managers, Administrators and Super Users. The image button in Joomla's TinyMCE editor opens the Media Manager which gives the user access to all of the site's images. They could delete or overwrite existing images or even upload images in places you don't want them to.

The traditional approach to that was disabling the image button for regular users or using a third party editor, such as JCE (JoomlaContentEditor), which supports per-user image directories.

With Joomla 4 this is no longer the case. There is a free of charge plugin called RestrictedFS [<https://restrictedfs.dgrammatiko.dev>], written by Dimitris Grammatikogiannis, a regular code contributor to Joomla itself, which limits each user's Media Manager access to a specific subdirectory (`images/user/username` where `username` is the actual username of the currently logged in user).

Note

Caveat: images uploaded through the Media Manager, even in a folder restricted by RestrictedFS, do not count as Akeeba Ticket System attachments. As a result they are a. publicly visible to anyone with the URL to the image and b. not removed when you run the scheduling command to remove old attachments. You need to weigh these two factors before enabling this feature.

First download and install the RestrictedFS plugin.

Go to System, Manage, Plugins and look for the plugin called Restricted File System. Enable the plugin and edit its options.

In the sole options, Jail User Groups, remove the Registered group which is there by default and add the user groups your support clients belong to. If you were following along our quick setup tutorial that'd be the Customer Support group. Click on Save & Close.

Caveat: Any user, including your support staff which is assigned to one of the user groups selected in the Jail User Groups option (or any of its children groups) will be unable to select and/or manage images outside their own user directory under `images/users`. In most cases this is perfectly fine; you don't want to give your support staff too much latitude. We are mentioning this because it's so obvious that it is easily overlooked — it happened to us while writing this documentation.

Go to Content, Media and click on the Options button, the the Permissions tab.

Find the user group your clients belong in and click on it. Set the Access Administration Interface, Create, Delete and Edit permissions to Allowed. Yes, you really need to add the “Access Administration Interface” permission; it allows the user to view the images on their restricted folder. If you do not explicitly allow this permission the user is unable to upload, delete or change any image!

Repeat if you have more than one user groups where clients belong to.

Click on Save & Close.

3. Attachments

Akeeba Ticket System allows user to optionally upload attachments with their tickets and ticket replies. These files are stored on your site, in the directory you configure in the component's Options page. The default is `media/com_ats/attachments`.

Attachment files are stored two directory levels deep, using a mangled name which is derived using at least one random component. This is a security measure. The filenames are unpredictable and created sparsely in a very large search space which means that an unauthorised user would not be able to predict or brute force their name to access them directly. Furthermore, they have no extension to make them non-executable. Finally, the default directory is protected against direct web access with a `.htaccess` (Apache) and `web.config` (IIS) file. This prevents a number of attack modes such as IDOR (Indirect Object Reference), uploading malicious code and executing it on your site, etc.

Attachments have, by default, the same visibility as the ticket itself. Therefore, attachments in public tickets are visible to everyone and can be downloaded by anyone who has their URL. You can optionally make all attachments private. This means that attachments in public tickets will only be visible to and can be downloaded by the person submitting the ticket they belong to and users with the Support Staff permission.

Akeeba Ticket System has its own set of upload permissions; it does not use the same ones used by the Media Manager. This is intentional. The Media Manager is set up to allow the upload of files which need to be publicly accessible on your site, mostly images, videos, audio files, PDFs and office application files. Attachments in a ticket system tend to be archives, log files, even executables depending on the nature of the support you are offering through the ticket system. Therefore it makes sense to have a different set of upload permissions to allow for files of a different nature to be uploaded. You can find these settings in the component's Options page.

4. User custom fields

Joomla 3.7 and later — and of course Joomla 4 — include a very powerful feature called Fields. This allows you to define custom fields in Joomla's interface which will apply to different areas of the CMS such as user profiles,

categories, articles etc or third party extensions. In this section we are interested in custom fields you can define for the user profile and which Akeeba Ticket System can use to display additional information next to the user's post to give you more context, namely the user's Avatar image and the user's Country.

4.1. Avatar (profile image)

An avatar (profile image) can make communicating with a user feel more personal. Instead of an impersonal username you can now see the face of the user or at least another graphic they chose to represent them.

Akeeba Ticket System can use any custom field defined in the user profile as the source of the avatar image as long as the field renders an `IMG` tag with its `src` attribute set to the avatar image's URL or a field which returns a URL to an image. This means that you can use the built-in Media or URL field, a third party media / image selection field, a third party gallery field, a third party plugin which integrates with an external avatar service etc.

For the purpose of this tutorial we will assume you want your users to upload an image to your site and use that as their avatar.

First, we need to allow the user to upload their avatar image to our site. We do NOT recommend using the built-in Joomla "Media" field as it gives the user full access to overwrite files in the media gallery of the site which is definitely not something you want. The easiest solution we've found is a third party plugin called Image Upload Field [<https://extensions.joomla.org/extension/core-enhancements/image-upload-field/>]. Even though it says it's for Joomla 3 it will work just fine on Joomla 4. After installing it please remember to publish both of the Ajax - Image upload and Fields - Image upload plugins on your site.

Next up, we'll create a folder for the user profile pictures. Go to Content, Media. Click on Create New Folder. Type **avatar** and click on Create.

Now we'll create the user profile field. From the side menu of your site go to Users, Fields. Click on the New button in the toolbar. Use the following settings:

- Title: Avatar
- Type: IMAGEUPLOAD
- Name: avatar
- Label: Profile Picture
- Description: Please upload a photo to use with our support ticket system. Supported file types JPG, PNG, GIF or BMP. Maximum file size 512Kb. Minimum image dimensions 32x32 pixels, maximum image dimensions 512x512 pixels.
- Required: No
- Only Use In Subform: No
- Destination folder: `avatar` (that's the folder you created in the previous step)
- Accepted image formats: select `jpg`, `jpeg`, `png`, `gif`, `bmp`
- Maximum image width: 512
- Minimum image width: 32
- Maximum image height: 512

- Minimum image height: 32
- Maximum file size (kb): 512
- Filename format: randomchar(12)

As-is, the custom field cannot be edited by the user which makes it rather useless! Click on the Permissions tab. Click on the Registered group. Set Edit Custom Field Value to Allowed. Click on the Manager group and do the same.

Finally click on Save & Close.

When a user edits their profile in the frontend they will now be able to upload their avatar image.

Now we need to tell Akeeba Ticket System to use this field.

From your site backend's side menu click on Components, Akeeba Ticket System, Control Panel. Click on Options. Click on the Common tab. From the Avatar field dropdown choose Avatar (avatar). This is the field we created above.

Whenever Akeeba Ticket System displays a user's post it will also display the image uploaded in the Avatar field above the user's username. If the user has not provided an avatar image it will fall back to the Gravatar image using the email address in the user's profile.

4.2. Country

Our on-line businesses tend to be inherently global. This means that we get to interact with people coming from different countries and cultures. What is innocuous or even expected in one culture may be very offensive in another. Knowing which country the person you're talking to comes from can help avoid this kind of misunderstandings.

Akeeba Ticket System can display a small flag with the user's country (and a tooltip with the full name of the country) next to the user's username when displaying their posts in a ticket thread. To do so it needs a user profile custom field which contains the country's ISO 3166 two letter code e.g. DE for Germany, GR for Greece and so on.

While you could use the built-in List field type, meticulously entering the ISO country codes and country names, we do not think this is a particularly practical approach. Instead, we recommend using the Country custom field included in Tassos Marinos' Advanced Custom Fields [<https://www.tassos.gr/joomla-extensions/advanced-custom-fields>]. The free of charge version will do. The two main reasons for the recommendation is that you do not need to enter 200+ country codes and names by hand and that it also does IP geolocation to pre-fill the field with the best guess of the user's country (the latter only in the paid version).

Now we'll create the user profile field. From the side menu of your site go to Users, Fields. Click on the New button in the toolbar. Use the following settings:

- Title: Country
- Type: ACF - Country
- Name: country
- Label: Your country
- Description: Tell us which country you live in. This gives us cultural context which will make it easier to provide support for you.
- Required: No

- Only Use In Subform: No
- Return Value: Country Code
- Multiple Country Selection: No

As-is, the custom field cannot be edited by the user which makes it rather useless! Click on the Permissions tab. Click on the Registered group. Set Edit Custom Field Value to Allowed. Click on the Manager group and do the same.

Finally click on Save & Close.

When a user edits their profile in the frontend they will now be able to select their country.

Now we need to tell Akeeba Ticket System to use this field.

From your site backend's side menu click on Components, Akeeba Ticket System, Control Panel. Click on Options. Click on the Common tab. From the Country Field dropdown choose Country (country). This is the field we created above.

Whenever Akeeba Ticket System displays a user's post it will show a flag with the user's country next to the username. If the user has not selected a country you will of course not see a flag at all.

5. Custom fields

One of the most powerful and most underrated features in Joomla is the support for fields a.k.a. “custom fields”. We strongly believe that Fields together with view template overrides are THE killer features in Joomla. We have been able to implement features on our site which previously required creating custom extensions using just core features. We have seen incredibly complex sites — even one which was essentially a mobile, touch-first showcase application — being realised with nothing more than core Joomla, custom fields and relatively simple view template overrides.

In a nutshell, the Joomla Fields feature allows you to extend content types in core Joomla and third party extensions, like Akeeba Ticket System, adding fields with additional information. The field content can be automatically displayed in a rather clumsy way or used in the code of view template overrides to provide functionality not envisioned in the core or third party component itself. Fields can be locked for editing or display to certain user groups and even limited for editing only in the backend, only in the frontend, or both.

Naturally, Akeeba Ticket System offers full integration with Joomla's Fields feature, both in ATS categories and tickets.

For example, you could use custom fields to display videos or image galleries in the description of a ticket category, style a ticket (think about marking a public ticket “important” and rendering it with a standout color scheme to draw attention) or simply request additional information from your clients when filing a ticket without relying on them remembering to type everything in the ticket text — something we all know never worked reliably.

In this section we assume that you have a modicum of familiarity with the Joomla fields feature in general. If not, or if you want to take a deep dive into what's possible with custom fields, we recommend reading the following excellent article series in the Joomla Community Magazine written by Marc Dechèvre:

- How far can we go with Joomla! and Custom Fields? [<https://magazine.joomla.org/all-issues/march-2018/how-far-can-we-go-with-joomla-and-custom-fields>] This is an introduction and a teaser of what you can do with custom fields.
- Explore the Core! Play with Custom Fields to enrich your content or your design [<https://magazine.joomla.org/all-issues/may-2021/explore-the-core-play-with-custom-fields-to-enrich-your-content-or-your-design>] This is a revisited version of the article above, going deeper into the field types built into Joomla.

- Custom Fields - Episode 2 : how to manage your Custom Fields [<https://magazine.joomla.org/all-issues/may-2018/custom-fields-episode-2-how-to-manage-your-custom-fields>]
- Custom Fields - Episode 3 : all the parameters one can wish for [<https://magazine.joomla.org/all-issues/october-2018/custom-fields-episode-3-all-the-parameters-one-can-wish-for>]
- Custom Fields - Episode 4: a step by step tutorial [<https://magazine.joomla.org/all-issues/april-2020/custom-fields-episode-4-a-step-by-step-tutorial>]
- Custom Fields - Episode 5 - List of Resources [<https://magazine.joomla.org/all-issues/july-2021/custom-fields-list-of-resources>] A treasure trove of further tutorials, presentations and information on custom fields in English, French, German and Spanish/
- Custom Fields - Episode 6: Make more with extensions [<https://magazine.joomla.org/all-issues/august-2021/custom-fields-episode-6-make-more-with-extensions>] A fantastic collection of third party custom fields types to do everything you imagined and even more!
- Custom Fields - Episode 7 Part 1: one Custom Field to rule them all [<https://magazine.joomla.org/all-issues/october-2021/custom-fields-episode-7-part-1-one-custom-field-to-rule-them-all>] A tutorial on the subform field type, including how to manage its content in a view template override

On top of these resources we are going to demonstrate how to do some really neat tricks in Akeeba Ticket System using custom fields.

5.1. Custom fields in Categories

Joomla Fields are fully supported in Akeeba Ticket System categories. You can do pretty much the same things you can do with Joomla Fields in Joomla's content categories — with a twist!

ATS Categories are displayed in two separate use cases:

- Category View. When you are displaying a list of categories (the main Categories page in Akeeba Ticket System) or the subcategories part of the category display page.
- Tickets View. When you are listing the tickets of a category.

Depending on the custom fields you are using you may want to only display them in one use case but not the other. For example, you may want to use a custom field to display a gallery of video tutorials at the top of the Tickets View page, helping your clients get help faster without even submitting a ticket.

Akeeba Ticket System categories allow you to select when to display custom fields: Never (you only get to use them in template overrides), Category View, Tickets View or both.

If you want to have different fields displayed in different use cases you will have to do template overrides and set the category's custom fields display option to None. Unfortunately, this display trick can only be implemented per category for all fields, not per each individual field (that's a limitation of how Joomla fields work).

Custom fields in template overrides

If you want to use custom fields in view template overrides just remember that the category has the same `jcfields` property you get in Joomla core content categories. This property includes the field values of all fields which would be visible to the user regardless of their Automatic Display setting i.e. it includes fields whose Automatic Display is set to Do Not Display.

For those of you implementing view template overrides we have added a handy dandy helper to make your life managing fields much easier. Joomla keys the `jcfields` array by field ID which is cumbersome and

practically unusable — you typically want to get a custom field *by name*. To help with that we implemented the `getFieldsByName` helper.

If you have the category in a variable named `$cat` and you want to get the information for a field called `example` on this category all you need to do is `$exampleField = $this->getFieldsByName($cat, 'example');`. This returns the object with the field information and value. The raw field value is accessible as `$exampleField->rawvalue` whereas the fully rendered HTML of the field display is accessible as `$exampleField->value`.

If the field you are looking for does not exist the field helper will return `null`.

If you want to get the entire `jcfields` array keyed by the field name instead of the field ID you just need to omit the second parameter. For example: `$allFields = $this->getFieldsByName($cat);`

So, if you want to display the rendered version of a field named `video` only when the custom field with the name `showVideo` is set to 1 you can do this in your template override:

```
<?php if ($this->getFieldsByName($cat, 'showVideo')->rawvalue == 1) {
    echo $this->getFieldsByName($cat, 'video')->value;
} ?>
```

To make it more safe, checking if both fields are defined before using them, you can do the following:

```
<?php
$customFields = $this->getFieldsByName($cat);

if (($customFields['showVideo'] ?? null) && ($customFields['video'] ?? null)
    && $customFields['showVideo']->rawvalue == 1) {
    echo $customFields['video']->value;
} ?>
```

5.2. Custom fields in Tickets

Important

If you rely on custom fields to collect mandatory, additional information when a user is submitting a new ticket you **MUST NOT** enable Akeeba Ticket System's new ticket by email feature (it's okay to allow replies by email). The reason should be pretty obvious: when a new ticket is created by email the custom fields **CAN NOT** be populated since there is no standard way to extract this information from a ticket.

Likewise, if you want to use custom fields as conditional fields (a set of custom fields “opens” for editing by the client when the support staff toggles another staff-only custom field) you should disable ticket replies by email **OR** make it clear to your users that they can only enter this information when they edit the ticket on your site.

Custom fields are very powerful but they only work in the web interface. Creating and replying to tickets over email is very convenient but cannot use custom fields. Power or convenience — or something in between? Choose wisely. It's ultimately up to you. You know your use case better than we possibly can!

And no, there really is no way to reliably populate custom fields from emails. We experimented with various ideas but they are all as good as a bouncy ball filled with nitroglycerin. That is to say, it sounds like a cool idea until you try it for real and regret it instantly...

Akeeba Ticket System has full support for Joomla Fields in tickets. Before you start using them you need to keep in mind a few basic concepts.

Custom fields can be limited for editing in the backend, frontend or both sides of your site. This is controlled by the field's Editable In option. If you want your support staff or your clients to be able to change this field's value in the frontend of your site you need to set this option to Site or Both. If you want your backend support staff to also be able to edit this field's value you need to set this option to Both.

Custom fields are by default only editable by Super Users. This makes them kinda useless. Remember to go to the Permissions tab in each field and select which user groups can edit them by allowing the Edit Custom Field Value permission. This is an extremely powerful setting with a very interesting repercussion. You can make a field editable ONLY by clients, ONLY by support staff, neither or both! A field only editable by a client would be ideal for a privacy consent or NDA agreement field which needs to be legally guaranteed to be set only by the affected party (the client). A field only editable by the support staff would be ideal for implementing conditional fields. We will see example of these further down this documentation.

Read-only fields are, by default, visible in the ticket edit form in the frontend and backend of your site. You can hide these fields by setting the Display When Read-Only option of the field to No. This would be necessary for fields meant to be only accessed by your support staff or generally for fields you don't want to clutter the edit ticket interface, something which could overwhelm the client.

Fields have an access level which controls when they are displayed. This applies to rendering the field value (and also making it accessible through the ticket's `jcfields` property for template overrides) when displaying the posts of a ticket AND when editing a ticket. By default this is set to Public which means that everyone, including Guest users, can see the fields. Please note that Joomla CANNOT limit the display of custom fields only to ticket staff and the ticket owner i.e. it's impossible to implement Akeeba Ticket System's "Private" visibility. The best you can do is EITHER set the field's Access to a view access level which only includes the support staff OR set the ticket's Automatic Display to Do not automatically display. In the former case only support staff will see the custom field contents, even if the client can edit it. In the latter case neither will see the custom field contents unless they edit the ticket. This kind of limited display is great when asking for privileged information like server access details, social security numbers etc.

Note

Super Users are special in Joomla. They can view and edit any field, regardless of the Access and Permissions of the field itself.

We understand that this is confusing but it's not a bug in Akeeba Ticket System. It's how Joomla itself works.

We strongly advise AGAINST using a Super User account to manage tickets. Even if you are the sole privileged user on your site we recommend creating a new, unprivileged user account and assign it to a group which has its Support Staff privilege allowed. Using this new account to reply to and manage tickets will be much easier when it comes to custom fields — and more secure, too!

Fields can optionally belong to a Field Group. You can have as many field groups as you want. When submitting a new ticket or editing a ticket the fields are displayed per field group. Each field group renders a card (a bordered area within the form) whose title is the field group's title. If you have a field group description it will be displayed as an information message below the group's title and before any fields. This is a convenient way to include a brief explanation of why you are asking for this information and/or short instructions for filling out the fields in the group.

Finally, fields can be assigned to one or more ATS categories. The default is All which means that the custom field appears in all ATS categories. You can change that so only the most pertinent fields appear in each category. Please note that field groups cannot be assigned in a category. However, empty field groups ARE NOT displayed.

Conditional fields

By default, Joomla fields do not allow you to display them conditionally i.e. based on the value of another field. We found this to be rather limiting. We addressed this shortcoming with the Content – Akeeba Ticket System Custom

Fields Show-on Behavior plugin. This plugin is installed and enabled by default when you install Akeeba Ticket System.

The plugin adds a new option in the General tab of the fields you define *in Akeeba Ticket System only*. It does not apply to fields defined for any other component, including the build-in Joomla components such as Articles, Banners, Contacts etc.

The plugin merely exposes to you a built-in Joomla feature. That's right, Joomla already has a "show on" behavior, it just doesn't expose it for custom fields. The format of this option's contents and how it works can be found in the Joomla Showon documentation page [https://docs.joomla.org/Special:MyLanguage/Form_field#Showon]. Please remember that you need to use the field names, not titles or labels, in the Show On field.

For example, if you have a List custom field with the name `showme` and you want to make another field display only when the `showme` field is set to 1 you need to enter the following Show On option value in the other field `showme : 1`. We will see a practical example of this in the conditional fields tutorial further below.

Important

What you enter in this field is **CODE, NOT PROSE**. As a result, the letter case and the use of whitespace (or lack thereof) matters!

If you have a field the **name** of which is `showme` (all lowercase) then you must enter the Show On option as `showme : 1` *in all lowercase, without spaces*. The following things will not work:

- `Show Me:1` There is no such field as capital S followed by how followed by space followed by capital M followed by e.
- `ShowMe:1` There is no such field as capital S followed by how followed by capital M followed by e.
- `showme : 1` There is no such field as `showme` followed by space.

Likewise the bitwise operators [OR] and [AND] must be entered exactly like that: opening square bracket followed by uppercase OR followed by closing square bracket; and opening square bracket followed by uppercase AND followed by closing square bracket. No spaces, no other whitespace, no lowercase characters, no non-Latin characters. The latter is very important because of homoglyphs (different characters which look alike) such as the letters A (capital Latin letter a) and Α (capital Greek letter alpha) or O (capital Latin letter o), Ο (capital Greek letter omikron) and 0 (number zero). You must not have spaces before or after [OR] and [AND].

Finally, remember that you are entering the field's name, not its title or label. The field's title is something you see in the backend Fields page and it's just for your information. The field's label is what is displayed in the form itself. The field's name is a unique identifier. Different fields can have the same title and/or label but they CAN NOT have the same name. That's why Joomla's Show On feature operates on the field's name, not its title or label.

Fields in template overrides

If you want to use custom fields in view template overrides just remember that the ticket object has the same `jcfields` property you get in Joomla core articles. This property includes the field values of all fields which would be visible to the user regardless of their Automatic Display setting i.e. it includes fields whose Automatic Display is set to Do Not Display.

For those of you implementing view template overrides we have added a handy dandy helper to make your life managing fields much easier. Joomla keys the `jcfields` array by field ID which is cumbersome and

practically unusable — you typically want to get a custom field *by name*. To help with that we implemented the `getFieldsByName` helper.

If you have the ticket in a variable named `$ticket` and you want to get the information for a field called `example` on this ticket all you need to do is `$exampleField = $this->getFieldsByName($ticket, 'example');`. This returns the object with the field information and value. The raw field value is accessible as `$exampleField->rawvalue` whereas the fully rendered HTML of the field display is accessible as `$exampleField->value`.

If the field you are looking for does not exist the field helper will return `null`.

If you want to get the entire `jcfields` array keyed by the field name instead of the field ID you just need to omit the second parameter. For example: `$allFields = $this->getFieldsByName($ticket);`

So, if you want to display the rendered version of a field named `video` only when the custom field with the name `showVideo` is set to 1 you can do this in your template override:

```
<?php if ($this->getFieldsByName($ticket, 'showVideo')->rawvalue == 1) {
    echo $this->getFieldsByName($ticket, 'video')->value;
} ?>
```

To make it more safe, checking if both fields are defined before using them, you can do the following:

```
<?php
$customFields = $this->getFieldsByName($ticket);

if (($customFields['showVideo'] ?? null) && ($customFields['video'] ?? null)
    && $customFields['showVideo']->rawvalue == 1) {
    echo $customFields['video']->value;
} ?>
```

5.2.1. Support staff-only fields

The more one talks to help desk agents and their supervisors the more apparent it becomes that being able to provide an amount of structured information not visible to the client may not just be a good idea but an essential part of being able to run a help desk efficiently or even possibly within regulatory compliance. Collecting and managing this information can be implemented in Akeeba Ticket System creating support staff-only fields.

Let's give an example to make it easier to understand. A travel agency allows its clients to file support tickets to sort out issues before, during or after the guided tour trip they have booked. Because an entire family may be booking a trip it's not possible to have an 1:1 assignment of Joomla users on the company's site and the client account numbers in the company's booking system — the entire family booking a trip would be a single client. A support agent would have to find out which booking reference and client account number in the company's booking system the support case refers to and log it for future reference, either the support agent's future reference as they continue the conversation with the client or a supervisor's when they are reviewing a sample of support cases for the yearly performance evaluation.

At the very least this requires two fields which need to only be able to be filled in by the support agent and only be visible to the support agent: Client Number and Booking Number.

We will assume that support agents are members of the Support Staff user group and that there is an access level called Support Access which only includes the Support Staff user group. We will also assume there are two ATS categories, Pre-Sales Inquiries and After-Sales Care. The fields we create will only be accessible in the After-Sales Care category. Finally, we will need to put them in a Field Group called Booking System Reference.

First we go to Components, Akeeba Ticket System, Field Groups to create our field group. Make sure the drop-down at the top left reads Tickets and click on the New button. Enter the following information:

- Title: Booking System Reference
- Description: Cross-reference the client information with the Booking System and enter the respective references in the fields below.
- Access: Support Access

Click on Save & Close.

Now go to Components, Akeeba Ticket System, Fields. Make sure the drop-down at the top left reads Tickets and click on the New button. Enter the following information:

- Title: Client Number
- Type: Text (text)
- Name: client-number
- Label: Client Number
- Required: Yes
- Default Value: n/a
- Filter: Text
- Maximum Length (characters): 15 — we assume the client IDs are a maximum of 15 characters in the company's booking system
- Field Group: Booking System reference
- Category: remove the “All” entry and select the “After-Sales Care” category.
- Access: Support Access

Click on the Options tab and set the following:

- Editable In: Site
- Automatic Display: Before Display — we want this information to be displayed in the ticket

Click on the Permissions tab. Select the Support Staff user group. Set the Edit Custom Field Value permissions to Allowed. Click on the drop-down next to Save & Close and select Save & New to start defining the next custom field. Enter the following information:

- Title: Booking Number
- Type: Text (text)
- Name: booking-number
- Label: Booking Number
- Required: Yes
- Default Value: n/a

- Filter: Text
- Maximum Length (characters): 30 — we assume the booking IDs are a maximum of 30 characters in the company's booking system
- Field Group: Booking System reference
- Category: remove the “All” entry and select the “After-Sales Care” category.
- Access: Support Access

Click on the Options tab and set the following:

- Editable In: Site
- Automatic Display: Before Display — we want this information to be displayed in the ticket

Click on Save & Close.

The client never sees these two fields but the support staff does. The fields are displayed right above the posts in the ticket (i.e. above the conversation).

When a ticket is first submitted these will both display as n/a, hinting the support agent they need to do some work. They can click on the Edit Ticket button to enter this information which presumably they queried in the company's booking system. If the client had not provided enough information they can instead ask them for clarification (“What's your full legal name so I can look you up in our system?”) without editing the ticket at this point.

Once this information is entered the support agent will see it above the conversation, letting them quickly pull up information about the client's booking to better help them.

5.2.2. Client-only fields

Sometimes there is information that only the client is allowed to enter. The support agent can see them but they must NOT be able to change them, typically for legal or regulatory reasons. Let's give a practical example.

An indie developer is publishing applications on the Apple App Store and the Google Play Store. These stores do not provide a help desk feature. The developer does not want to deal with the unmitigated hellscapes that's doing support over email for hundreds of people. Instead, they want all their support to go through their site's ticket system.

Since they do not want to have their clients jump through unnecessary hoops they decide to allow Guest Tickets so that a client can simply file a ticket entering their name, email address and what they want help with. However, because of GDPR and similar legislation, they can't *just do that*. A name and an email address is Personally Identifiable Information. The user must provide explicit consent that they agree to the Privacy Policy of the site before being allowed to submit the ticket.

It sounds like a chicken and egg issue. Luckily, there's a way to implement that with a custom field.

Here's our battle plan. We will use the third party PrivacyCheckbox custom field plugin [<https://extensions.joomla.org/extension/authoring-a-content/custom-fields/privacycheckbox/>], written by one of the Joomla core contributors. This field's value will only be editable by the Guest group i.e. only when someone has not logged in and filing a guest ticket. This field cannot be changed by support staff or by the user after they log in. The only drawback is that since acceptance of the privacy policy happens when the user is still a guest user it's not recorded to the Joomla Privacy component. When the user logs in they will have to go through Joomla's privacy consent.

First, install the third party plugin and remember to publish it — it's the Fields - Privacy Checkbox plugin. You do not need to configure it.

Now go to Components, Akeeba Ticket System, Fields. Make sure that the drop-down at the top left reads Tickets and click on the New button. Use the following information:

- Title: Privacy Policy
- Type: Privacy Checkbox (privacycheckbox)
- Name: privacy-policy
- Label: Privacy Policy
- Required: Yes
- Checkbox Note: By checking this box I agree to the site's Privacy Policy (link to Privacy Policy here).
- Access: Guest (so it only shows for Guest users)

Click on the Options tab and use the following settings:

- Label: Hide
- Automatic Display: Do not automatically display — This is not strictly necessary. Since the field has Guest access it will not show up for staff and the logged in client anyway. This is just a reminder to you that the field will be invisible.

Next up, click on the Permissions tab. Click on the Guest group. Set the Edit Custom Field Value permission to Allowed.

5.2.3. Conditional fields (or how to ask clients for additional information)

Those of us doing support for web software have often found ourselves in need to ask our clients for connection information. The problem with that is two-fold. If we ask for that information as custom fields when the ticket is first submitted we create the false expectation that we will get into the site ourselves and do all the work for the user meaning a lot more work for us and the client having no idea what we did to fix their problem (which means many more tickets from that client). If we don't ask for this information in custom fields we are effectively trusting that the user will somehow follow our instructions and provide the information we are asking them for. Unfortunately, even with very detailed instructions on what we need, this is rarely the case. This causes support delays.

Is it a lost cause? Are we bound by a chicken and egg issue which make our work harder? Not with the power of custom fields AND the nifty Content – Akeeba Ticket System Custom Fields Show-on Behavior plugin we ship with Akeeba Ticket System.

This is going to be a longer tutorial. We will need to create *five* custom fields:

1. A Site Administrator Login URL where the user will give us the URL to their site's administrator login.
2. A Username field for the administrator login.
3. A Password field for the administrator password.
4. An Additional Information text field where the user can communicate additional security measures which may be in place (e.g. an administrator directory password, Two Factor Authentication settings etc)
5. An Ask For Additional Information field only visible to the support staff controlling the display of the previous four fields.

Everything will be placed in a Site Connection Information field group to keep it neat and tidy.

The first four fields will be displayed in the ticket view to support staff and the client who owns the ticket *only*. If the ticket is public, a guest or another client who can view the ticket will NOT be able to view these fields. These fields will be editable by both the client and the support staff. The last field will not be displayed at all in the ticket view and will only be editable by the support staff.

We assume that your support staff belongs in a group title Support Staff and there is a view access level on your site called Support Access which only contains the Support Staff group. We also assume that your clients belong to a user group called Clients and there is a view access level called Client Access which only contains the Clients user group. Moreover, we assume there is a view access level called Ticket Access which includes both the Clients and Support Staff groups. Finally, we assume that you only want these fields to be available to an Akeeba Ticket System category called Software Support.

The custom field group

Go to Components, Akeeba Ticket System, Field Groups. Make sure the drop-down at the top left has Tickets selected. Click on the New button.

- Title: Site Connection Information
- Description: Please enter the connection information to your site. After you're done, click on Save & Close. Please remember to send a reply to ticket otherwise we will NOT know that you provided the connection information.
- Access: Public

Click on Save & Close.

The support staff-only field

The purpose of this field is to act as a toggle. By default it will be set to No, meaning that users will NOT be able to see the fields under the Site Connection Information — this includes the time when they are submitting a new ticket. When the support staff edits the ticket and set the toggle to Yes the fields will become visible to the client who will be able to fill them in.

Go to Components, Akeeba Ticket System, Fields . Make sure the drop-down at the top left has Tickets selected. Click on the New button.

- Title: Ask For Additional Information
- Type: Radio (radio)
- Name: access-info-toggle
- Label: Ask For Additional Information
- Description: Does the user need to provide access details?
- Default Value: 0
- Field Group: Site Connection Information
- Category: only select the Software Support category
- Access: Support Access

In the Radio Values area create two settings:

1. Text: JNO Value: 0
2. Text: JYES Value: 1

In the Options tab set the following:

- Field Class: btn-group
- Automatic Display: Do not automatically display

In the Permissions tab, select the Support Staff group, set the Edit Custom Field Value to Allowed.

The client-facing fields

The next four fields will be visible to and editable by the users.

Normally, there's a caveat. Joomla uses the Access level to determine both who can view the field values in the ticket view and who can edit their values in the ticket edit page. Here's the problem. If we set the access to Support Access the client can neither see *nor* edit these fields which beats the purpose of having them. If we set the access to Ticket Access then both the client and the support staff can see and edit these fields. However, all other clients will *also* be able to see these fields on public tickets which creates a privacy and security issue.

Akeeba Ticket System's Content – Akeeba Ticket System Custom Fields Show–on Behavior plugin has a solution for that. It exposes a custom field parameter called Private Display. When this is enabled, Akeeba Ticket System will only display the values of these fields in the ticket view to the support staff and the ticket owner. Other clients will NOT be able to see them. Therefore you can ask for connection information securely even on a public ticket!

Furthermore, you will notice that we are NOT making these fields required. This is on purpose. Remember that the support staff will set the staff–only Ask For Additional Information field to Yes to “open” the access to the client–facing fields. Because of what we described above, these fields will *also* be editable by our support staff. If we make them Required then the support staff will have to fill them in to save the ticket when they set Ask For Additional Information to Yes. However, the whole point is that the support staff does not have this information and needs to request it from the user. Unfortunately, we have not found a good solution to this problem so we are just making all fields optional (i.e. Required is set to No).

Without further ado let's start creating our fields.

Go to Components, Akeeba Ticket System, Fields . Make sure the drop–down at the top left has Tickets selected. Click on the New button.

- Title: Site Administrator Login URL
- Type: URL (url)
- Name: admin-login-url
- Label: Site Administrator Login URL
- Description: The URL to log into your site's administrator backend.
- Schemes: HTTP, HTTPS
- Relative URLs: No
- Show On: access-info-toggle:1

- Private Display: Yes
- Field Group: Site Connection Information
- Category: only select the Software Support category
- Access: Client Access

In the Options tab set the following:

- Placeholder: <https://www.example.com/administrator/index.php?foobar>
- Automatic Display: Before display

In the Permissions tab, select the Support Staff group, set the Edit Custom Field Value to Allowed. Then select the Clients group, set the Edit Custom Field Value to Allowed.

Click the drop-down menu next to Save & Close. Click on Save & New.

- Title: Username
- Type: Text (text)
- Name: username
- Label: Username
- Description: The username of a Super User
- Filter: Text
- Maximum Length (characters): 100
- Show On: `access-info-toggle:1`
- Private Display: Yes
- Field Group: Site Connection Information
- Category: only select the Software Support category
- Access: Client Access

In the Options tab set the following:

- Placeholder: `admin_user`
- Automatic Display: Before display

In the Permissions tab, select the Support Staff group, set the Edit Custom Field Value to Allowed. Then select the Clients group, set the Edit Custom Field Value to Allowed.

Click the drop-down menu next to Save & Close. Click on Save & New.

- Title: Password
- Type: Text (text)

- Name: password
- Label: Password
- Filter: Raw
- Maximum Length (characters): 192
- Show On: `access-info-toggle:1`
- Private Display: Yes
- Field Group: Site Connection Information
- Category: only select the Software Support category
- Access: Client Access

In the Options tab set the following:

- Placeholder: password
- Automatic Display: Before display

In the Permissions tab, select the Support Staff group, set the Edit Custom Field Value to Allowed. Then select the Clients group, set the Edit Custom Field Value to Allowed.

Click the drop-down menu next to Save & Close. Click on Save & New.

- Title: Additional Information
- Type: Text Area (textarea)
- Name: additional-info
- Label: Additional Information
- Description: Do we need to know anything else to access your site (administrator directory password, two factor authentication settings, geographic IP blocking etc)?
- Rows: 5
- Columns: 50
- Maximum length (characters): 1000
- Filter: Text
- Show On: `access-info-toggle:1`
- Private Display: Yes
- Field Group: Site Connection Information
- Category: only select the Software Support category
- Access: Client Access

In the Options tab set the following:

- Automatic Display: Before display

In the Permissions tab, select the Support Staff group, set the Edit Custom Field Value to Allowed. Then select the Clients group, set the Edit Custom Field Value to Allowed.

Click on Save & Close.

Putting it all together

A client submits a ticket. They do not see the “Site Connection Information” fields at all.

When replying to the client it becomes apparent that the support staff will need access to the client's site to help them out.

The Support Staff edits the ticket and sets the Ask For Additional Information field to Yes.

Then the support staff tells the client that they need to edit their ticket to enter their site connection information.

The client comes to the site and clicks the Edit button for their ticket. They enter the site connection information and clicks on Save & Close.

This causes an automatic message to be posted to the ticket thread, saying that user such-and-such has edited the ticket information. The ticket status is automatically set to Open.

The support staff now comes back to the site. The connection information appears above the conversation tab. This information only appears for the support staff and ticket owner, even if the ticket is Public.

5.2.4. Replacing the geolocation field

Older versions of Akeeba Ticket System had a custom field plugin called “Geolocation”. It would detect the geographic location of your client and show it on a map.

With the integration of Joomla custom fields we no longer have to provide our own field. You can instead use a third party Joomla custom field plugin which replicates the same function.

For example, you can use the `plg_fields_osm` by CoolCat Creations [https://github.com/coolcat-creations/plg_fields_osm] to let the client enter their address (it auto-completes) and display it on a map powered by OpenStreetMap. This is very similar to what our old plugin did.

6. Tags

Akeeba Ticket System allows you to tag tickets and users.

Keep in mind that Joomla has a *global* tagging system. This means that all tags on your site are available for all content types be it core content (articles), article categories, contacts, banners, Akeeba Ticket System tickets or users. This is by design. We cannot make it so that only “ticket tags” appear when tagging tickets, for example.

Tagging tickets allows you to keep all related tickets together. You can access all tickets — and any other content types — by clicking on the tag's name in the frontend of your site on any ATS view which shows you a list of tickets or a single ticket. In the backend you can filter by tags and only see those tickets which are tagged with that specific tag.

Tagging users will only display the user tags in the backend and frontend ticket view, in the user information area displayed next to a user's tickets. This can be used to convey information which does not necessarily warrant using a

user group such as VIP status or using tags to indicate that some users may have specific requirements when providing support.

6.1. Support staff–only tags

As noted above, Joomla has a global tag system. This means that any tickets you use for tickets or users will be publicly visible in the core Joomla tags component. They will also be shown everywhere someone can select a tag, either submitting content (e.g. Editors submitting articles) or filtering content (e.g. Smart Search).

In most cases you want to make sure that the tags you are using for tickets and users are only displayed to Support Staff and are not made available to other users of your site. You can do that from your site's backend.

Go to Components, Tags.

Select the tags you want.

Click on the Actions drop–down and select Batch.

In the Set Access Level drop–down select an access level which only grants access to the Support Staff of your site.

Click on Process.

These tags will now be only visible to your site's support staff.

Handling new tags

If you create new tags by typing them into the tag field yourself — or as the result of migrating from ATS 4 or earlier — you will need to go to the backend of your site and repeat this process for these new tags.

Organising your tags

We strongly recommend using *nested tags*. Create a top level tag called Ticket Tags and a top level tag called User Tags. Create the respective ticket and user tags under the top level tags. This makes organising your tags much easier than having everything at the top level.

7. Nested Categories

Akeeba Ticket System uses Joomla's category manager for its tickets categories. That is to say, the ticket categories are managed by Joomla's com_categories, just like Joomla's own article categories.

One of the main features of Joomla's category manager is the support for nested categories. One category can be under a parent category which is under a grandparent category and so on.

In most cases you are not going to use nested categories for tickets. Typically, you will only have a small handful of ticket categories. Larger organisations with thousands of clients and dozens of support staff may need to use nested categories to make it easier for clients to locate the support category they need and for the staff to provide support.

If you are using nested categories the same considerations apply as with any other Joomla component, core or third party, which has nested categories.

Permissions are cascaded across categories first, then by user groups. When Joomla encounters an explicit Deny on any given permission throughout the category and user groups cascade it will be denying that privilege even if a child category and/or other user group has an explicit Allow for the same permissions. The rule is simple: Deny is stronger

than Allow which is stronger than Inherit. We recommend using a third party extension to understand how permissions cascade across categories and user groups on your site.

Access levels do not cascade but may make it impossible to reach a category. If the parent category has its Access set to Special and the child category has its access set to Public then a Registered user can only access the child category but not the parent category. Normally, the user would need to go into the parent category to navigate into the child category. Therefore you are making it hard (but NOT impossible!) for a user to navigate into the child category. Do note that a user could enter the URL to the child category directly to access it. Therefore setting the Access of a parent category does NOT protect children categories from being accessed.

Conversely, if you want to allow easy access to the child category you should create a menu item to that category. Otherwise your user would need to know the child category's alias and understand how Joomla URL routing works to construct a URL similar to `https://www.example.com/tickets/parent/child.html` to view the child category.

Some **category options** have a Use Global option value. When you select that special option value you are telling Joomla to use the value from the parent category. If the parent category does not have a value it will keep searching until it finds the root category. If it hits the root category it will fall back to the setting you specified for that option in the component's Options page. We recommend setting all options to explicit values (Show / Hide, Yes / No, ...) to avoid any misunderstandings about what Joomla and ATS do when trying to determine which is the actual option value they need to use for a Use Global option value.

8. InstantSearch

We have offered an InstantSearch / InstantReply feature since the very first version of ATS released back in 2011. The concept is simple and powerful. As your client types the title of their ticket ATS will perform a search using that title, presenting the user with relevant results. Typically, this reduces the number of tickets submitted because clients are very likely to find an answer in public tickets, documentation etc on your site.

Older versions of ATS only supported searching for public tickets or articles in Akeeba DocImport. This was very limiting as many sites do not use public tickets and very few people are using DocBook XML, the requirement to use Akeeba DocImport.

Instead, ATS 5 and later uses either Joomla's Smart Search or a third party search engine. This allows you to show search results from many different data sources to best help your clients find what they are looking for.

8.1. Smart Search menu items

By default, Smart Search shows search results from your *entire* site. In most cases this is not very convenient. Ideally, you'd want to show search results from select data sources, e.g. your public tickets and an FAQ, Knowledge Base or documentation core Joomla articles category.

Luckily, this is perfectly possible with Akeeba Ticket System and a hidden Joomla feature: Smart Search filters.

Creating Smart Search filters

First, we need to create a Smart Search filter. Go to the backend of your site and click on Components, Smart Search, Filters. Click on the New button in the toolbar. The full documentation of this features can be found by clicking the Help button in the toolbar when adding or editing a Filter. The following paragraphs are meant to be additional notes on what we have found as Joomla users to be useful to know. It's not meant to substitute the Joomla documentation nor does it mean that we will provide support for a core Joomla feature.

You will need to the Search by Type and Search by Category areas of the Filter. The Search by Type tells Joomla's Smart Search what kind of content items to return. This is up to core and third party extensions to register with Joomla.

For example, Akeeba Ticket System registers the content type Ticket. Therefore if you want to search through *public* tickets you need to check the Ticket type under Search by Type.

Note

Only *public* tickets will be searched by Smart Search. The reason is that Smart Search can only limit the display of tickets by Joomla view access level. All clients who can post to a category would have the same access level. If we allowed Smart Search to search private tickets it would return results from other people's tickets, including potentially private or privileged information. Therefore the search is limited to public tickets only.

Other useful content types are Articles (Joomla core articles) and Documentation (Akeeba DocImport articles).

Remember that this is just the type. By default, all content items of the selected types will appear in the search results. You probably don't want that; most likely you want specific categories to be searched. This is possible through the Search by Category options. Select the categories you want.

Note

DocImport does not use Joomla categories. As a result its categories do not appear on that list. We will work on this issue in a later version of DocImport. You are unlikely to even notice; DocImport is used by all of ten or so Joomla extensions developers, including us. For most real world users of Akeeba Ticket System what matters is core Joomla articles whose categories do, indeed, appear here.

Please note that Joomla lists all categories across all core and third party extensions without any indication of which category is which. We agree with you that this is far from optimal. Please lodge your complaint to the Joomla forum and the Joomla Issue Tracker. You can mention @nikosdion (our lead developer) who will chime in and offer a solution. The thing is, unless someone other than us complains that this is far from being reasonable / convenient there is of course no impetus for the core Joomla maintainers to change it or take up our offer to make changes. We can't blame them for that; if nobody complains the understanding is that this feature works fine so why touch it?

Using Smart Search filters in Akeeba Ticket System

Now that you have a Smart Search Filter you can tell ATS to use it for InstantSearch queries when a client is submitting a new ticket. You can assign the filter either component-wide or to a specific category.

Note

Category options for InstantSearch DO NOT cascade across nested categories. You need to explicitly define them in each category. If the Use Global value is used for InstantSearch (default) then ATS will fall back to the component's Options for InstantSearch.

Either in the component's Options or a ticket category's Ticket Options tab there's the InstantSearch Method drop-down. Set it to Smart Search. This makes the Search Filter drop-down show up below. Select the Filter you created there. ATS will apply that filter when returning InstantSearch results.

Using Smart Search menu items in Akeeba Ticket System

An alternative to using Search Filters directly is selecting which Smart Search menu item you want to use. Just like the Smart Search Filter above you can select it either in the component Options or in the Ticket Options of a tickets Category.

This feature can be used instead of or additionally to a search filter. In the latter case the search filter you have selected in ATS will override the filter selection in the menu item.

The menu item allows you to set up how the search results will be displayed by Joomla, e.g. whether you want to show advanced search options, date filters, the results taxonomy, the description of each search result, pagination, alternative search term suggestions etc.

Combining both options allows you to create a very customised search experience which will funnel your clients into past tickets, documentation, knowledge base articles etc instead of submitting new tickets. As a result clients will be happy that they got a solution to their problem quickly and you will have to deal with a smaller volume of support tickets.

8.2. External search engine

Smart Search is a pretty decent search solution but it's limited by what is possible with a PHP-based search engine. Most commercial search engines (e.g. DuckDuckGo, Google etc) allow you to run a custom search for your site only.

Akeeba Ticket System supports that. Just like Smart Search, you can set this up either per category or in the component's Options page.

Either in the component's Options or a ticket category's Ticket Options tab there's the InstantSearch Method drop-down. Set it to URL. This makes the Search Engine URL field show up below. You can enter the URL to your search engine there. Use the variables {site} for your site's domain name (optional) and {search} for the URL-encoded search term (mandatory). For example, to use DuckDuckGo use the URL:

```
https://duckduckgo.com/search.html?site={site}&q={search}
```

Likewise, to use Google Search use the URL:

```
https://google.com/search?q={search}+site%3A{site}
```

For Google we use a compound search query. We combine the ticket's title with the suffix `site:yourdomain.com` to tell the search engine to only return results from your site. This trick can work on most other search engines, e.g. for Bing:

```
https://bing.com/search?q={search}+site%3A{site}
```

You can also use the external search engine URL feature in ATS to return search results from a custom search engine you use in your Intranet / Extranet, e.g. an instance of Apache Solr, Elasticsearch etc or even a custom search component installed on your site. Anything where you can access a URL with a search query and get an HTML page back with search results can be used with Akeeba Ticket System.

9. Guest tickets

Akeeba Ticket System allows, under certain conditions, Guest users to file tickets. Guest users is Joomla! lingo for site visitors who are not logged in, i.e. they do not have a user account yet. Please make sure you understand the "How does this feature work?" section below before trying to use this feature. It tells you how this feature works and which settings, in ATS and Joomla! itself, will conflict with it and make it not work as you think it should.

If you do not want this feature, you don't need to do anything. Guest tickets will only be allowed if you set up one or more ATS categories to allow Guest users to create tickets.

How does this feature work?

Unregistered (Guest) users can create tickets as long as the ATS category has the Create privilege for the Guest user group. Joomla's user registration must be enabled in the Options of the Users component in the backend; or the Force user creation when filing Guest tickets option in Akeeba Ticket System's Options page must be set to Yes.

The users will need to provide an email address, a name and a desired username when filing a ticket from the web. You can optionally tell Akeeba Ticket System, through its Component options, to show a CAPTCHA for Guest users. This is a good idea since it will cut down on spam ticket submissions. Please note that you must have one or more plugins in the "captcha" group activated in Joomla's Plugins page and select the CAPTCHA plugin to use. If either of these conditions is not met a CAPTCHA will not be shown to Guest users trying to file a new ticket. Also note that, obviously, the CAPTCHA will not apply to tickets created by sending an email!

When filing a ticket by email we obviously already have their email address. If they have provided their full name as part of the From address of their email we will use that. Otherwise we will also use the email address as the user's full name. Akeeba Ticket System will create a Joomla! username derived from the user's full name or, if that's not possible, their email address. Users can change their email address and full name in Joomla's user profile edit page whenever they want, as long as you've provided such a page in your site's frontend. Users can optionally be allowed to change their username themselves if you enable the respective feature in Joomla's Users component's Options page. By default, this feature is disabled. Do note that this is a feature of Joomla! itself, not a feature of Akeeba Ticket System.

As implied above, a new user account is created for the guest user and assigned to the ticket. That user account has a random password. The username and the password are both emailed to the user by Joomla! itself. If you have configured Joomla! to not send the password to the user then the user will very obviously not receive their password which makes it impossible for them to log in. In this case they will either have to use the Forgot Password feature of Joomla! itself or contact you and ask for a password to be issued. For this reason we recommend that you configure Joomla! to send the password to new users over email.

A special note about what happens when the user submits a ticket but required fields are missing or the CAPTCHA check fails. Akeeba Ticket System needs to create the user *before* checking for required fields and CAPTCHA (CAPTCHA is technically a required Joomla form field). This is a limitation of how Joomla works. This means that in this case a user is created and immediately deleted. If your site is configured to send an email to the user on new user registration (default Joomla behavior) your guest user will receive a confusing email telling them a user account was created but no such user account exists on your site until the finish submitting their ticket. This is NOT a bug, it's the only way this can be implemented. We strongly recommend NOT using Guest tickets. Instead, let the users register a user account and *then* let them file tickets. You could even use something like Akeeba SocialLogin to provide frictionless registration with the users logging in (and automatically creating a user account) using their social media accounts such as Facebook, Twitter, GitHub, Google account, Microsoft account or Apple ID.

The user account created for the guest ticket may need to be activated by the user, per the Options in Joomla's backend. If, however, you have enabled Force user creation when filing Guest tickets the user registration settings of Joomla! will be ignored and the user account will be activated. As noted above, enabling this option will also allow Akeeba Ticket System to create new Joomla! users even if you have disabled user registration in the Options page of Joomla's Users component.

The user will need to log into your site with the created user account to reply to the ticket or file more tickets. If they try to file a ticket as a Guest with an email address already in use by a user account on the site they will be asked to log in to file a new ticket.

Why not implement guest tickets just by email address?

Because it's woefully insecure.

Email addresses are supposed to be handled as public information. You do give it to people you know or even people you don't know, e.g. printing it on your business card. Anyone who knows or guesses your email address could access your guest tickets and reply to them as if they were you. This is ridiculously insecure.

That's the reason why commercial helpdesk SaaS like ZenDesk require people contacting you to create a user account to view the entire ticket thread and/or send their replies by email. Akeeba Ticket System also does the same, the latter if you enable the ticket replies by email feature. It's the only way to have a modicum of assurance that the people who view or reply to tickets are those who are in control of a specific email address.

10. Receiving emails

Akeeba Ticket System is primarily designed as a helpdesk component where submitting new tickets and replying to tickets takes place on the site itself. This allows for intricate, time-saving features like the ones described in the earlier sections about custom fields and InstantSearch.

There are, however, many sites where the ticket workflow is best suited to be managed via email, e.g. there are very few categories (possibly even just one), there is only a small number of people managing tickets (possibly even just one), there are no custom fields, clients are expected to prefer sending an email over logging into the site and so on and so forth. Akeeba Ticket System makes it possible to cater for any combination of the following email-to-ticket workflows:

- Support staff can reply to existing tickets by email.
- Clients can reply to their existing tickets by email.
- Clients can submit new tickets by email.
- Guest users can submit new tickets by email.

These workflows are implemented through the Akeeba Ticket System - Fetch Email plugin and CRON jobs.

Caveats

Email messages are far more limited than full blown websites. The entirety of the information you are receiving over an email message can be summed up like this: name and email address of the sender; subject; email body; attachments.

As a result the following features cannot possibly work over email:

- **Ticket custom fields, Ticket tags, User Tags, Time Spent.** There is no reliable way of parsing free text into the structured format of these fields.
- **User information, avatar, total time spent, user groups, previous tickets** shown next to each user's reply in a ticket thread. Obviously, these are features which can only be implemented on your site. While some of that information COULD be sent via email it becomes problematic when it comes to attachments, as will be discussed below.
- **InstantSearch.** InstantSearch is designed to make search queries and display their results before the client submits their ticket. By definition, once we receive an email the client has already submitted their ticket. Therefore it makes no sense providing them with a link to search results.
- **Canned replies.** These are snippets of text which can be inserted in your ticket reply as a Support Staff member. By definition, if you are using an email client application or a webmail site to send replies via email you do not have access to them.
- **Signatures.** These are only displayed on the site, they are not included in the email message.

Moreover, receiving new tickets and/or ticket replies over email requires making sense of the subject and body of the email.

The subject is pretty straightforward... unless it contains any character which cannot be encoded in 7-bit ANSI. You see, the original email specification (RFC 822 [<https://datatracker.ietf.org/doc/html/rfc822>]) was written back in 1982 and is in fact a standardization of the hodgepodge of message exchange protocols which sprang in the 1970s, back at the dawn of networked computers. Computers were not capable of displaying characters with diacritics beyond some *very* basic ones, let alone characters from different character sets like Cyrillic, Greek etc. There is a way to encode full Unicode characters in a way that is sent by email and most email clients support it. ATS will try to parse the

subject using those standards and do its best to represent it as real Unicode information which can be stored in your site's database.

The body text is even more complicated. You see, email was designed to send short, plain text messages. Most of the email you see today is actually HTML email, i.e. an HTML document sent as an inline attachment, with or without an alternative, text-only representation. ATS will try its best to divine if the email is plain-text only (very popular among Linux and German users in particular), HTML-only or HTML-with-plain-text. If it's just plain text, ATS will convert it to a basic HTML representation since all ATS posts to tickets are HTML. This works for messages sent from the vast majority of mail clients but it all depends on whether the mail client uses the standard convention of attaching the subject as the *first inline attachment* in the message. Yes, there is no standard for that, just a convention.

The other problem is that every single image you include in a mail message, including inline images, is in fact an **attachment**. ATS will process all attachments as such, even if they are inline. This means that the email sender's user account needs to have the Attachments permission to allow these to be attached to their ticket. Furthermore, images anywhere in the document, even those inlined below the “reply above this line” which indicates the end of the client's reply, will be processed as attachments — even if they are part of their signature. On top of that, for security reasons, ATS will NOT display attachment inline. Any inline attachments in mail messages are shown as broken images or empty text. This may make communication with clients problematic if they are using inline images to communicate something. You will have to figure out which one of their attachments corresponds to each inline image they reference in their ticket.

Finally, you need to set up CRON jobs for ATS to periodically check for email. This may be a bit frustrating in the sense that it is not only an additional administrative burden for you as a site owner but also confusing for the client who were they to visit your site would not see their new ticket or ticket reply appear immediately but only after the CRON job has run. Running the CRON jobs too often could bog down your server, running the CRON jobs too infrequently would make clients wonder if their ticket / reply didn't make it through and possibly resend it.

Generally speaking, receiving tickets or ticket replies by email messages should be avoided when possible. The user experience visiting your site to submit a new ticket or reply to an existing ticket will be far superior to using email.

10.1. The plugin options

Displayed in the Joomla! Plugin Manager as Akeeba Ticket System - Fetch Email

This plugin allows your site to retrieve email from a mail server and create new tickets or replies to existing tickets based on your settings. It makes the *mailfetch* CRON command available.

CRON jobs are required

This plugin only provides the infrastructure required to retrieve emails but **does not automate the process, i.e. publishing it is required byt NOT sufficient to retrieve emails.**

You will need to create a CRON job, either using the CRON CLI script or the CRON URL. Please note that the System - Akeeba Ticket System reply by email plugin has been removed because it had inherent consistency issues when fetching email.

If you are creating a CRON job use the *mailfetch* CRON command to retrieve emails.

These are the parameters which allow users to reply or create new tickets by sending emails to your site. Please remember to enable the respective plugin or the CRON job script.

Notes on the technology used

Starting with Akeeba Ticket System 3.2.0 we are no longer using the legacy PHP IMAP extension to retrieve and parse email because it has not been maintained since the late 2000's and is no longer up to the task. Instead, we are using

the Horde IMAP library which is written in pure PHP and does not require any PHP extensions. It's also faster and much more reliable. Nothing changes from your perspective.

Unless you are using Gmail / G Suite. Because of changes made by Google to their server we can no longer offer the old I am using GMail option and you can not keep on using Gmail / G Suite accounts over IMAP with regular password authentication either. This is NOT because of the library change explained above but because Google decided to discontinue support for a web standard. The reason we are now using the Horde library is that it's the only way we could support the only authentication method Google supports. It is complicated to set up but does work once set up.

The plugin options

Reply by email	When enabled allows users to reply to their tickets by email.
Only for managers	When enabled the Reply by email feature is only available for managers (support staff)
Create ticket by email	When enabled allows users to create new tickets by emails. Obviously, the user needs to send the email from the email address they used to register on your site.

Warning

This is dangerous! A user with an auto-reply may cause an endless amount of tickets being created all the time. We strongly recommend NOT using this option.

Track email UID	When enabled Akeeba Ticket System will track the unique identifier of each email message it has processed to prevent double posting of tickets e.g. when an error occurs after a new ticket is created but before it's marked as read / deleted. This is recommended to be enabled on production sites. If you change email providers or mailboxes this might cause new email to be dropped if the new email provider / mailbox uses the SAME unique IDs as the old one. In this case you will need to disable this option and enable the Delete after post creations option, documented below.
Text version only	When enabled Akeeba Ticket System will try to retrieve the plain text version of the email and use it as the post content. If the user's mail client only sends an HTML version of the email this could cause empty tickets / replies to be created.
Default category	The category where the new tickets created by the "Create ticket by email" feature will be placed.
Mail server type	Your incoming mail server type (IMAP, POP3 or GMail). If you are using a GMail / G Suite mail account please refer to the "GMail / G Suite Initial Configuration" section of this documentation.
Mail server host	The hostname or IP address of your mail server
Port	The port of your mail server
Use SSL	If your server uses SSL set this to Yes
Use TLS	If your server uses TLS set both this to either When Available or Always and also set Use SSL to Yes
Validate certificates	If you have enabled Use SSL and Use TLS select this option to have ATS verify the SSL certificates presented by your mail server. If the certificates are self-signed this WILL fail. We recommend using this option only when your mail server is using a commercial SSL certificate.
Username	The username you use to connect to your incoming mail server. Usually this is either your email address or the part of your email address before the @ sign. Please consult your email provider.

Password	The password you use to connect to your incoming mail server.
Mail folder	If you have an IMAP / Gmail / G Suite mail server select the folder where the messages are stored. This is usually INBOX (all capital letters)
API Client ID	Only applies if you are using a Gmail / G Suite mail account. Please refer to the "Gmail / G Suite Initial Configuration" section of this documentation.
API Secret Key	Only applies if you are using a Gmail / G Suite mail account. Please refer to the "Gmail / G Suite Initial Configuration" section of this documentation.
Authorize Gmail	Only applies if you are using a Gmail / G Suite mail account. Please refer to the "Gmail / G Suite Initial Configuration" section of this documentation.
Access Token	Only applies if you are using a Gmail / G Suite mail account. Please refer to the "Gmail / G Suite Initial Configuration" section of this documentation.
Refresh Token	Only applies if you are using a Gmail / G Suite mail account. Please refer to the "Gmail / G Suite Initial Configuration" section of this documentation.
Delete after post creations	Should the retrieved email messages be deleted from the server after the post/ticket has been created? If this is set to No then the emails are only marked as read. If you set this option to No we strongly advise that you set Track email UID to Yes.

10.2. Integrating with Gmail / G Suite

Starting March 2020 we have removed the I am using Gmail checkbox from our software due to a change in Google's Gmail and G Suite email services.

If you want to use a Gmail or G Suite email account with Akeeba Ticket System you will need to undergo the following process *once per site and email address you want to use with Akeeba Ticket System*.

Please keep in mind that this is not a limitation of Akeeba Ticket System. It is a consequence of changes effected by Google without prior notice or an easier alternative, as explained in detail in the Frequently Asked Questions below.

Overview

Akeeba Ticket System needs to retrieve your emails from the mail server and process them for the Reply by Email and New Ticket by Email features to work. The standard way to do that is connecting to your mail server with either the IMAP or the POP3 protocol using a username and a password for authenticating (logging into) the server. For most mail server the username is your email address and the password is, well, the password you use to login to your email account.

Starting in the early 2010's, Google introduced Two Factor Authentication (TFA) for its Gmail and G Suite Mail accounts (collectively called "Gmail" in this documentation). Due to the higher security standard inherent to TFA it was no longer possible to use your password for IMAP / POP3 authentication. They introduced application-specific passwords.

Starting in 2019 and enforced throughout 2020, Google completely discontinues standard IMAP password authentication, either with a regular password (for non-TFA accounts) or an application-specific password (for TFA accounts). Instead, they have migrated to a proprietary authentication method called XOAUTH2.

Instead of entering your email address and your password / application-specific password you need to instead login to your Google account associated with your Gmail mail account in your browser. Google will send back to your site two long pieces of text called **tokens**.

The [Definition: *Access Token*] is what is used for authenticating to IMAP. It is also set to expire frequently, approximately once every hour. The second token is called the [Definition: *Refresh Token*] and can be used to request a new Access Token when the current Access Token has expired.

What this oversimplified overview doesn't address is that tokens *are not issued for your Gmail user account* in general. In other words, they are not like app-specific passwords which were valid for any and all applications accessing Gmail. They are issued *for your Gmail user account to an OAuth2 API application registered with Google* – and only this application can ever use these tokens. This is meant to be a double assurance measure. On one hand, the Gmail user can revoke the API application's access to their Gmail account whenever they see fit, without having to change their Gmail password and disturb *every other* connected application. On the other hand, Google has a tight control over who can ask Gmail users for access to their email account.

10.2.1. Initial Configuration

Creating an OAuth2 API application

For reasons explained in the Frequently Asked Questions below it is ultimately not possible for Akeeba Ltd to create an API application which allows you to connect your Gmail account to your site without having to undergo this complicated process. You will need to create an API application yourself, free of charge, in Google Cloud Platform [<https://console.cloud.google.com>]. You can then have Akeeba Ticket System use the connection information of that API application to request access to your Gmail account. **This process only needs to be carried over once per Gmail account / G Suite organization.**

Important

The information under the “Creating an OAuth2 API application” section are provided for information purposes only. They are not meant to be definitive. They are subject to change without prior notification by Google. If you find that the steps have significantly changed since the publication of this document please contact Google for support. Google requests that third parties, like us, do not provide support for its products. As a result we are NOT allowed to help you with your Google Cloud Platform issues.

Creating an OAuth2 API application

First, you need to go to Google Cloud Platform [<https://console.cloud.google.com>]. Make sure that you are logged in with the correct user account, depending on what you want to connect to your Akeeba Ticket System installation:

- If you are going to connect **your Gmail account (@gmail.com)** you must be logged in with the same Gmail account you are going to connect.
- If you are going to connect **a G Suite email account** you must be logged in as an *administrator* of the same G Suite organization where the email you want to connect is registered with.

Check the currently logged in account the same way as you do in all Google web products, from the icon at the top right corner of your screen.

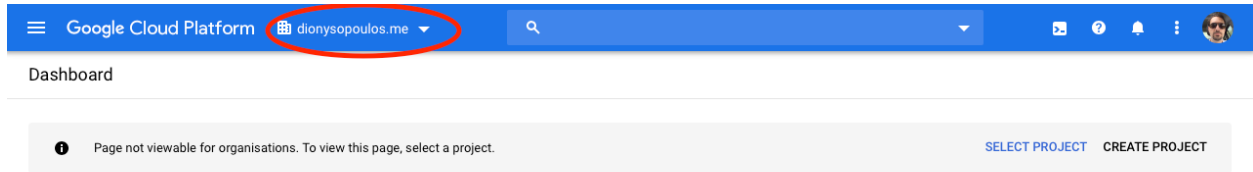
Enable Google Cloud Platform for your Google account

This is described in Google's Getting Started page [<https://cloud.google.com/apis/docs/getting-started>]. The takeaway is that you may have to enable billing *even though what you are going to be using is free of charge*. That's a weird requirement of Google and is ABSOLUTELY NOT under Akeeba Ltd's control. Thank you for your understanding.

Create an API project

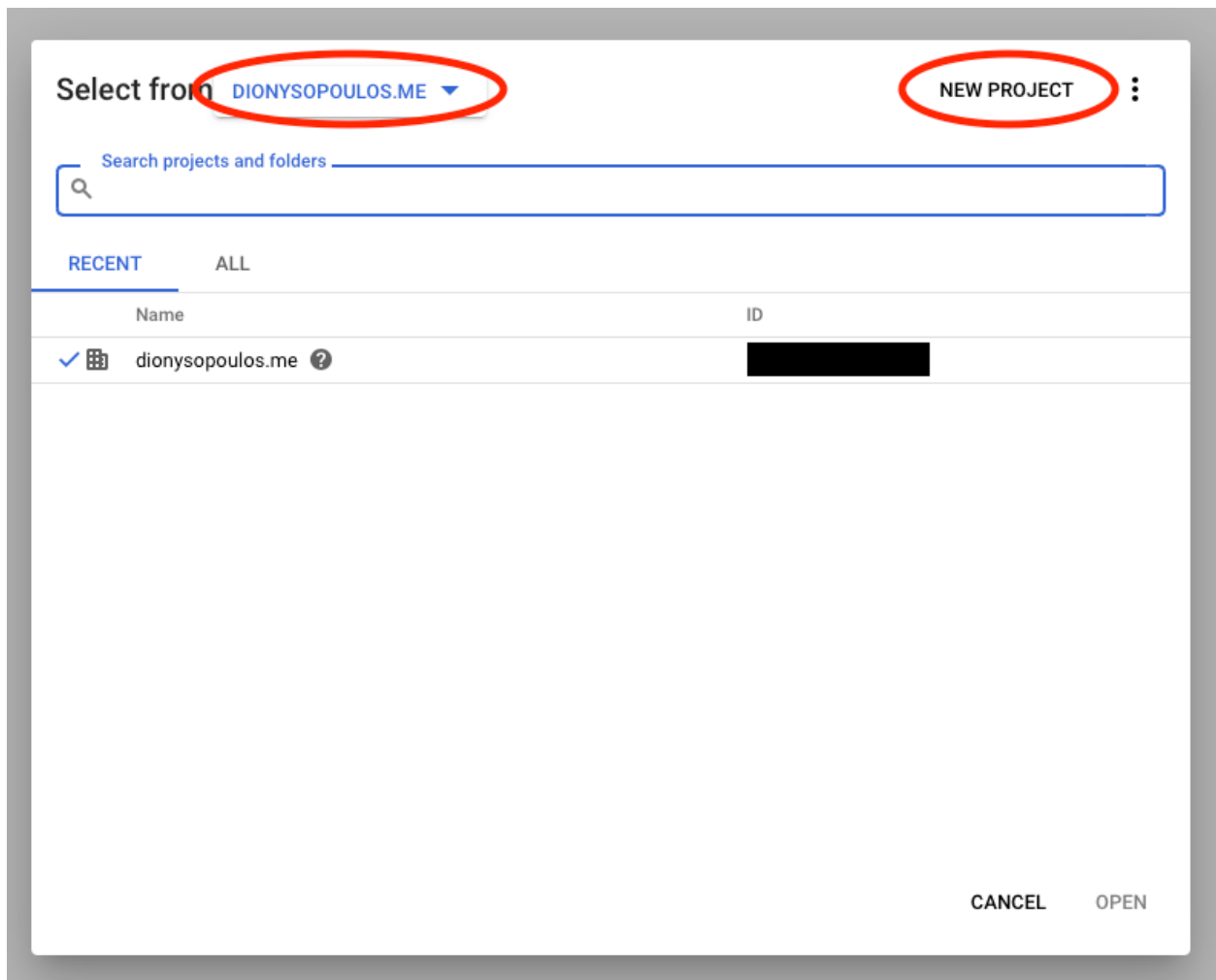
From the top of the page, next to the “Google Cloud Platform” text, there is a dropdown with your projects.

Show the projects management popup



Click on it to display the projects management popup.

Projects management popup



From the leftmost popup make sure that you select your G Suite organization. This does not apply if you are using your personal Gmail account.

Then click on the NEW PROJECT button in the top, right hand corner of the page.

New project, first step

The screenshot shows the 'New Project' form in the Google Cloud Platform interface. At the top, there is a blue header with the Google Cloud Platform logo and a search icon. Below the header, the title 'New Project' is displayed. The form consists of three main sections:

- Project name ***: A text input field containing 'Akeeba Ticket System demo'. Below the field, it states 'Project ID: akeeba-ticket-system-demo. It cannot be changed later.' with an 'EDIT' link.
- Organisation**: A text input field containing 'dionysopoulos.me'. Below the field, it states 'This project will be attached to dionysopoulos.me.'
- Location ***: A dropdown menu showing 'dionysopoulos.me' with a grid icon. A 'BROWSE' button is located to the right of the dropdown. Below the dropdown, it states 'Parent organisation or folder'.

At the bottom of the form, there are two buttons: a blue 'CREATE' button and a white 'CANCEL' button with a blue border.

In the Project name field enter something descriptive, e.g. “Support System Integration”. This is just for you. Keep it short.

Click on the blue CREATE button.

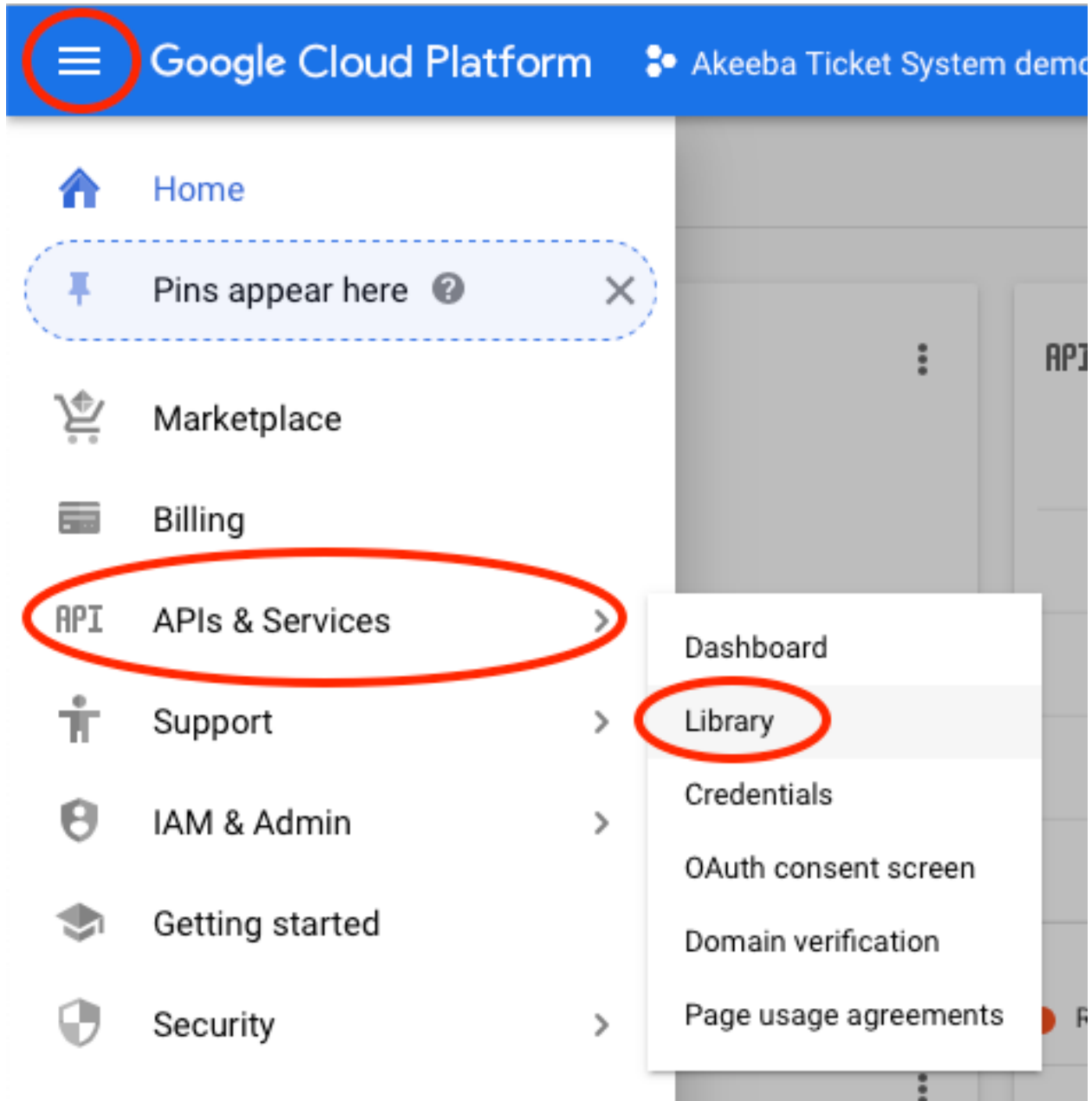
Wait for a few seconds until Google informs you that the project has been created.

Enable the Gmail API on your API project

Note

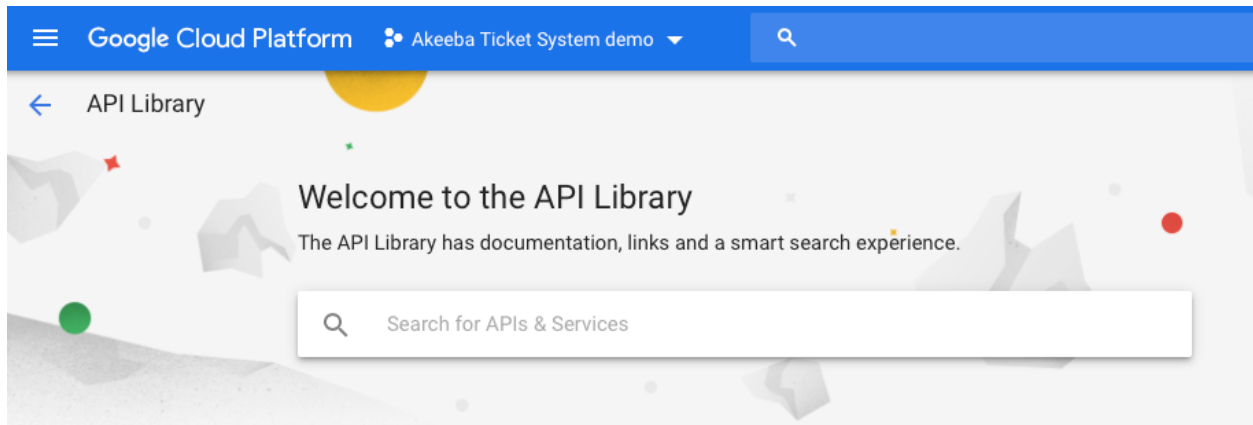
Yes, this also applies if you are using G Suite. As far as Google is concerned, Gmail and G Suite email are one and the same.

Navigating to the APIs & Services Library



From the new page click on the hamburger menu icon, hover over APIs & Services and click on Library.

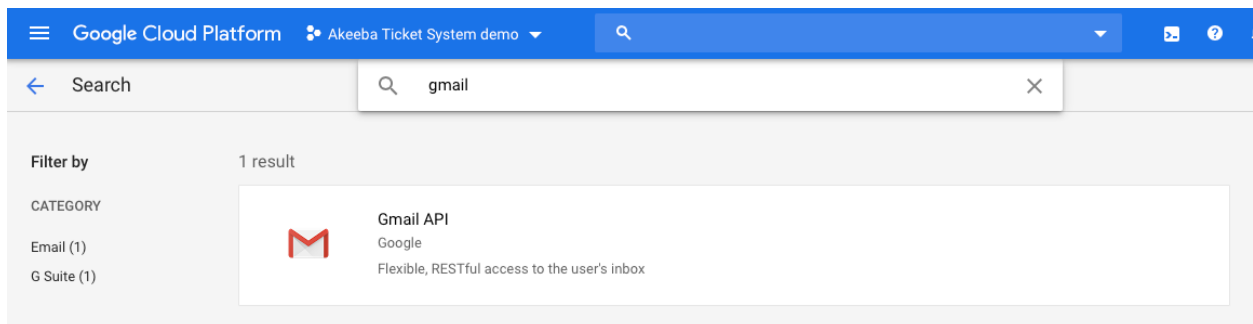
The APIs & Services Library page



In the APIs & Services Library page click on the "Search for APIs & Services" and type **gmail**

After a few moments you will see only one result called "Gmail API".

The Gmail API result



Click on it.

The Gmail API page

Google Cloud Platform Akeeba Ticket System demo

API Library

Gmail API

Google

Flexible, RESTful access to the user's inbox

[ENABLE](#) [TRY THIS API](#)

Type
APIs & services

Last updated
10/12/2019, 02:34

Overview
The Gmail API lets you view and manage Gmail mailbox data like threads, messages, and labels.

About Google

Wait a few moments for the ENABLE button to appear. Then click on the ENABLE button and wait until its control panel page appears.

The Gmail API control panel page

Google Cloud Platform Akeeba Ticket System demo

APIs & Services
Gmail API

Overview [DISABLE API](#)

To use this API, you may need credentials. Click 'Create credentials' to get started. [CREATE CREDENTIALS](#)

Details

Name
Gmail API

By
Google

Service name
gmail.googleapis.com

Overview
The Gmail API lets you view and manage Gmail mailbox data like threads, messages, and labels.

Activation status
Enabled

Tutorials and documentation

[Learn more](#)

[Try in API Explorer](#)

[Maintenance & support](#)

Traffic by response code

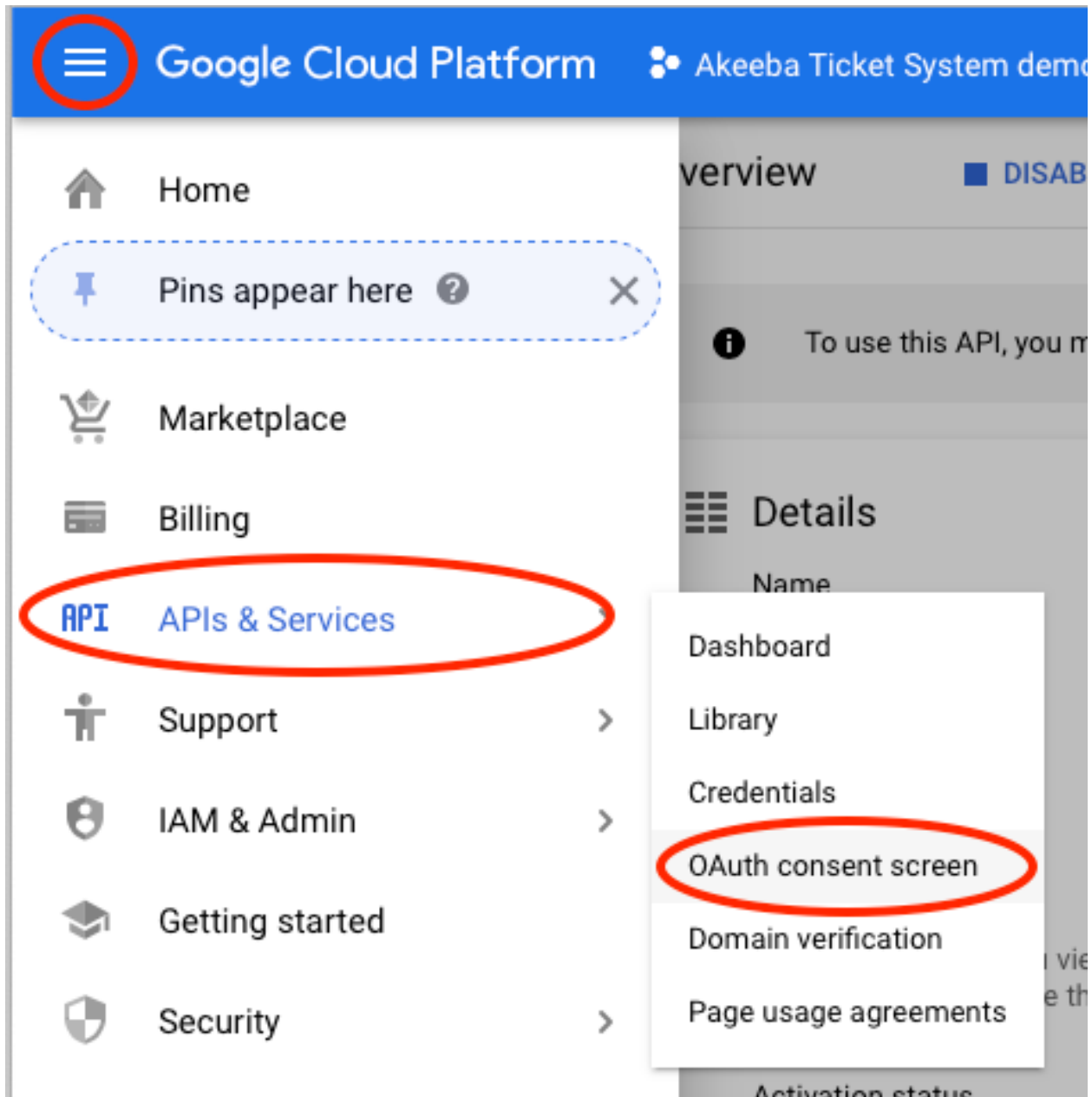
Request/sec (2 hr average)

No data is available for the selected time frame.

[View metrics](#)

Create an OAuth2 consent screen

Navigating to the APIs & Services Consent Screen



From the new page click on the hamburger menu icon, hover over APIs & Services and click on OAuth Consent Screen.

Setting up an Internal project

Google Cloud Platform Akeeba Ticket System demo

APIs & Services OAuth consent screen

Choose how you want to configure and register your app, including your target users. You can only associate one app with your project.

User Type

Internal ?

Only available to users within your organisation. You will not need to submit your app for verification.

External ?

Available to any user with a Google Account.

CREATE











In the new page select **Internal** as your project type.

Important

You **MUST** select Internal. This project type only allows your personal Gmail address / any email address in your G Suite organization to be authorized. This is what you want and the only way that you will not need to go through Google's verification process. If you accidentally select External you will need to go through the verification process which will take a massive amount of time and effort on your part only to be declined since you don't have a valid use case for an External project that allows anyone, with any Gmail / G Suite email address, to authorize your application to access their account.

Click on the blue CREATE button to proceed.

OAuth2 consent page setup, top section

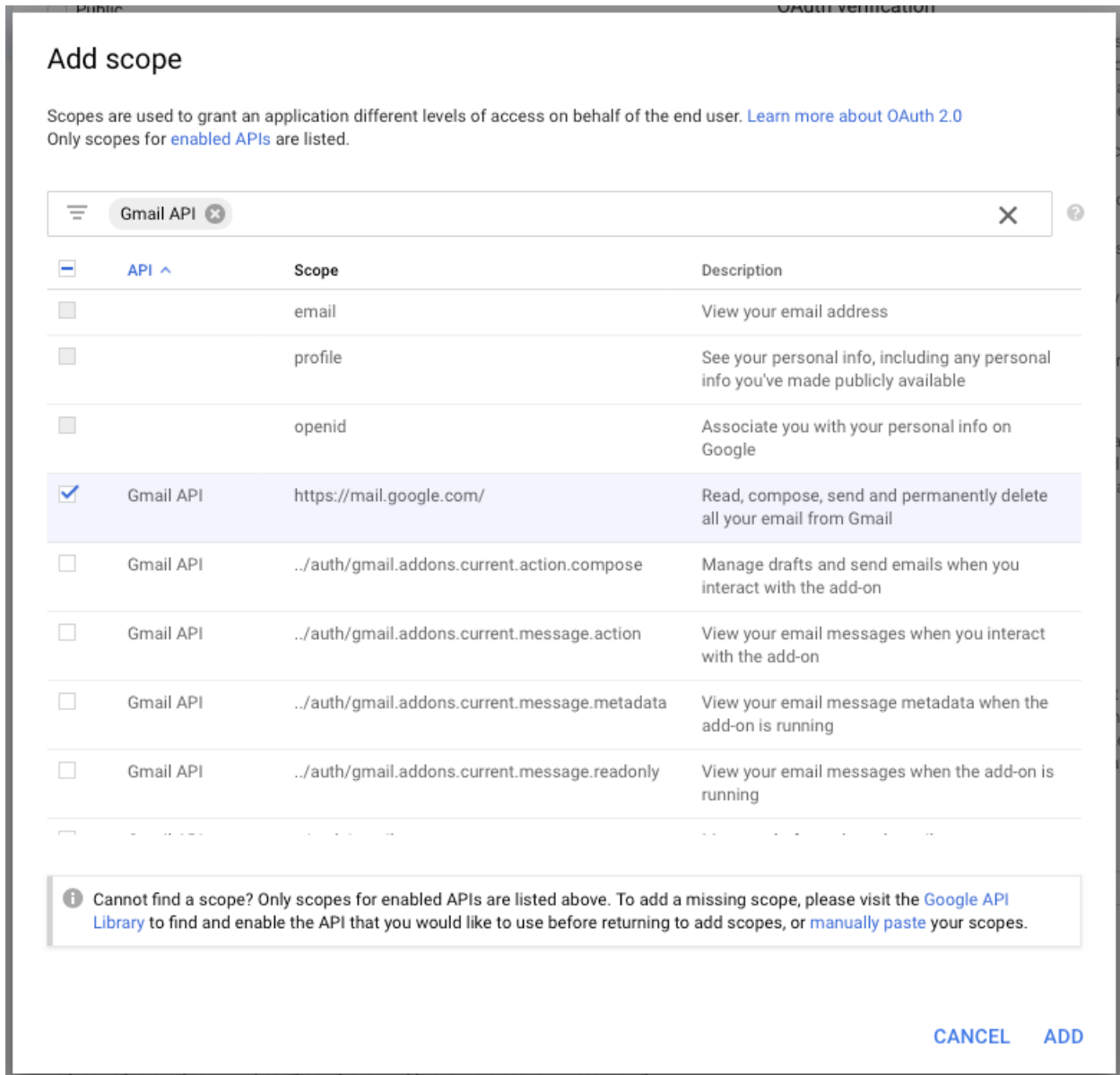
API APIs & Services	OAuth consent screen
<ul style="list-style-type: none">  Dashboard  Library  Credentials <li style="background-color: #e0e0e0;"> OAuth consent screen  Domain verification  Page usage agreements 	<p>Before your users authenticate, this consent screen will allow them to choose whether they want to grant access to their private data, as well as give them a link to your terms of service and privacy policy. This page configures the consent screen for all applications in this project.</p> <p>Application type</p> <p><input type="radio"/> Public Any Google Account can grant access to the scopes required by this app. Learn more about scopes</p> <p><input checked="" type="radio"/> Internal Only users with a Google Account in your organisation can grant access to the scopes requested by this app.</p> <p>Application name  The name of the app asking for consent</p> <input style="width: 100%;" type="text" value="Akeeba Ticket System"/> <p>Application logo  An image on the consent screen that will help users recognise your app</p> <input style="width: 90%;" type="text" value="akeeba-ticket-system-512.png"/> <input type="button" value="Browse"/>  <p>Support email  Shown on the consent screen for user support</p> <input style="width: 100%; background-color: black;" type="text"/> <p>Scopes for Google APIs Scopes allow your application to access your user's private data. Learn more</p> <p>If you add a sensitive scope, such as scopes that give you full access to Calendar or Drive, Google will verify your consent screen before it's published.</p> <ul style="list-style-type: none"> <input type="checkbox"/> email <input type="checkbox"/> profile <input type="checkbox"/> openid <p style="text-align: right;"><input type="button" value="Add scope"/></p>

In the new page enter an Application Name. This will be displayed when authorizing Akeeba Ticket System to your Gmail account. We recommend entering something along the lines of “ATS on example.com” where example.com is the domain name of your site.

In the Application Logo select any square image over 128x128 pixels. You can use your logo, for example.

Then click on the Add scope button. This opens a popup dialog.

OAuth2 scope popup



In the search box at the top enter **Gmail API** to filter the results.

From the list below select the Gmail API entry with the Scope <https://mail.google.com/>

Note

Yes, this also applies if you are using G Suite. As far as Google is concerned, Gmail and G Suite email are one and the same.

Click on the ADD button.

OAuth2 consent page setup, bottom section

Scopes for Google APIs

Scopes allow your application to access your user's private data. [Learn more](#)

If you add a sensitive scope, such as scopes that give you full access to Calendar or Drive, Google will verify your consent screen before it's published.

email

profile

openid

https://mail.google.com/ 

Add scope

Authorised domains

To protect you and your users, Google only allows applications that authenticate using OAuth to use Authorized Domains. Your applications' links must be hosted on Authorized Domains. [Learn more](#)

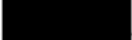
.com 

example.com

Type in the domain and press Enter to add it

Application Homepage link

Shown on the consent screen. Must be hosted on an Authorized Domain.

https://www.com/ats-info.html

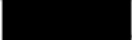
Application Privacy Policy link

Shown on the consent screen. Must be hosted on an Authorized Domain.

https://www.com/privacy-policy.html

Application Terms of Service link (Optional)

Shown on the consent screen. Must be hosted on an Authorized Domain.

https://www.com/tos.html

Save

Cancel

Under Authorized Domains enter your site's domain name. **It is VERY important that the domain name you enter here matches EXACTLY the site you are going to be using to link Akeeba Ticket System with your email address.** Do remember that `www.example.com` and `example.com` are *different domains*.

Note

If you have not yet linked the domain name to your Google Cloud Platform account you will be asked to do so. The easiest way is to be using Google Analytics on your site.

In the Application Homepage Link enter the URL to your site. **It is VERY important that the domain name in this URL matches EXACTLY one of the Authorized Domains you set up above.** Do remember that `www.example.com` and `example.com` are *different domains*.

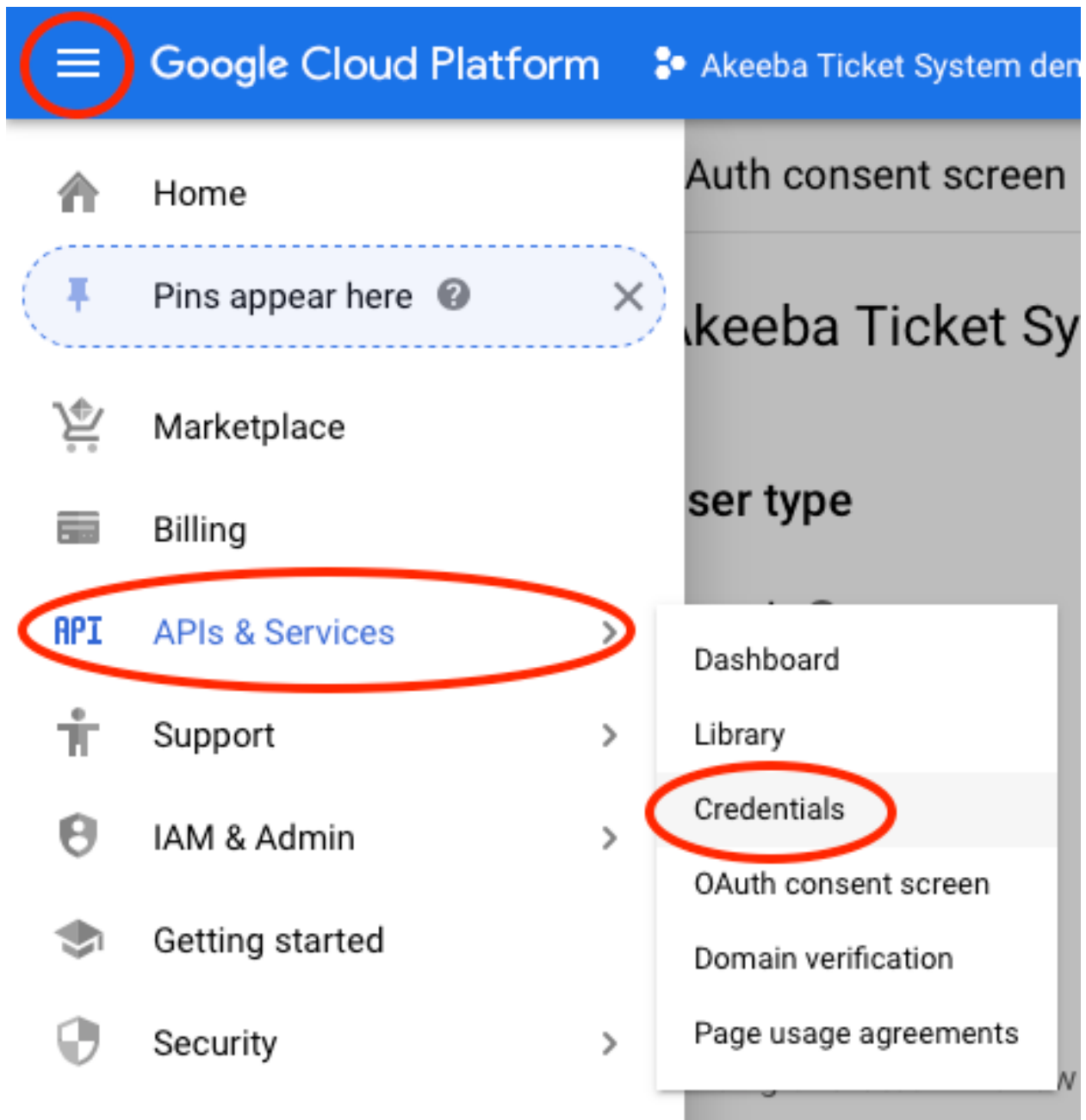
In the Application Privacy Policy Link enter a URL on your site. This can be a dummy page, as long as it doesn't return a 404. **It is VERY important that the domain name in this URL matches EXACTLY one of the Authorized Domains you set up above.** Do remember that `www.example.com` and `example.com` are *different domains*.

You don't need to enter an Application Terms of Service Link.

Finally click on the Save button.

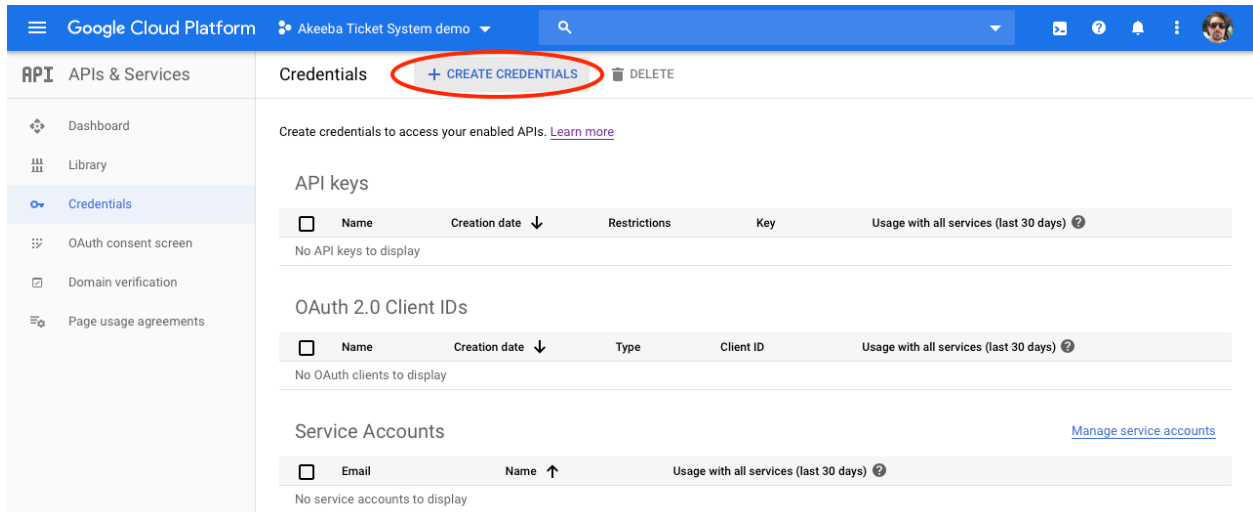
Get your API credentials

Navigating to the credentials page



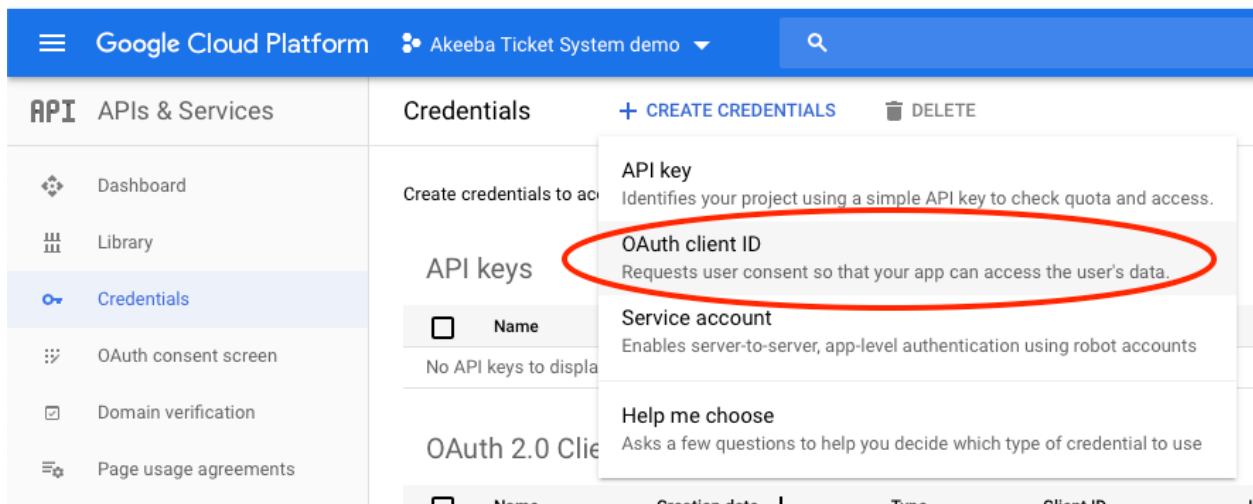
From the new page click on the hamburger menu icon, hover over APIs & Services and click on Credentials.

The credentials page



On the credentials page click on CREATE CREDENTIALS at the top of the page.

New credentials popup



From the drop-down, select OAuth client ID.

New credentials setup page

☰
Google Cloud Platform
🔗 Akeeba Ticket System demo ▼
🔍

← Create OAuth client ID

For applications that use the OAuth 2.0 protocol to call Google APIs, you can use an OAuth 2.0 client ID to generate an access token. The token contains a unique identifier. See [Setting up OAuth 2.0](#) for more information.

Application type

Web application
 Android [Learn more](#)
 Chrome App [Learn more](#)
 iOS [Learn more](#)
 Other

Name ?

ATS demo

Restrictions

Enter JavaScript origins, redirect URIs or both [Learn more](#)

Origins and redirect domains must be added to the list of authorised domains in the [OAuth consent settings](#).

Authorised JavaScript origins

For use with requests from a browser. This is the origin URI of the client application. It cannot contain a wildcard (https://*.example.com) or a path (<https://example.com/subdir>). If you're using a non-standard port, you must include it in the origin URI.

https://www. [REDACTED].com
🗑️

https://www.example.com

Type in the domain and press Enter to add it

Authorised redirect URIs

For use with requests from a web server. This is the path in your application that users are redirected to after they have authenticated with Google. The path will be appended with the authorisation code for access. Must have a protocol. Cannot contain URL fragments or relative paths. Cannot be a public IP address.

https://www. [REDACTED].com/administrator/index.php?option=com_ajax&group=ats&plugin=AtsMailfetchGmailCode&format=raw
🗑️

https://www.example.com

Type in the domain and press Enter to add it

Create
Cancel

Select the Web Application under Application Type.

Enter any Name you want; this is only for your convenience.

Under Authorized JavaScript origins enter your site's URL, e.g. `https://www.example.com`. **It is VERY important that the domain name in this URL matches EXACTLY one of the Authorized Domains you set up in the OAuth consent page setup earlier.** Do remember that `www.example.com` and `example.com` are *different domains*.

The Authorized Redirect URIs is the most important part of this process. It needs to be your site URL plus `/administrator/index.php?option=com_ajax&group=ats&plugin=AtsMailfetchGmailCode&format=raw`

For example, if your site is `http://www.example.com/mysite` you would need to enter

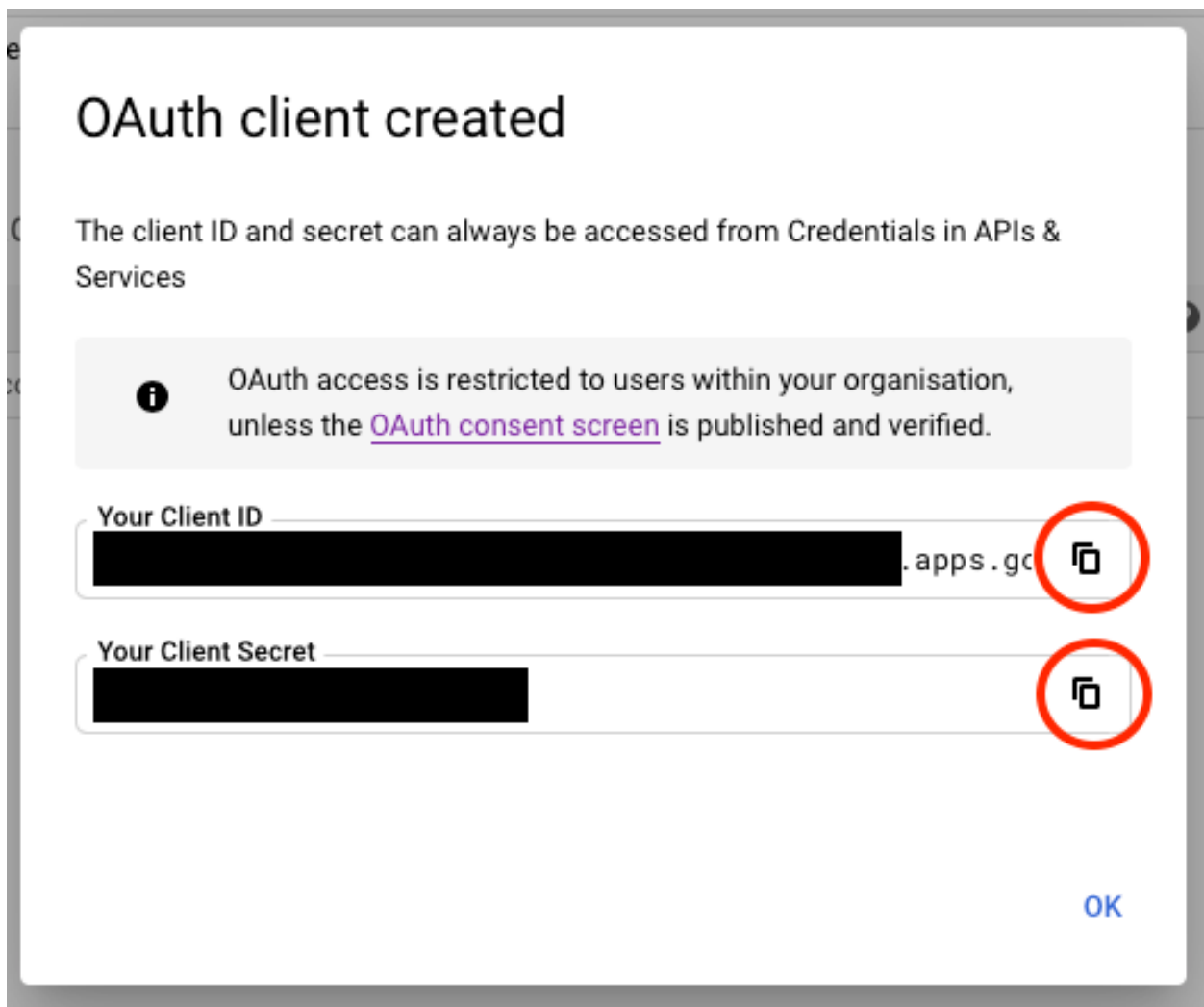
```
http://www.example.com/mysite/administrator/index.php?option=com_ajax&group=ats&plugin=AtsMailfetchGmailCode&format=raw
```

Warning

If you get this URL wrong you will NOT be able to connect Akeeba Ticket System to your email address. You will get an error 400 from Google. If that happens to you, you will need to come back to this page and fix the Redirect URI.

Click on Create.

The API credentials popup



You will now see a popup with your API credentials. Before doing anything else **copy these credentials to a safe place**. You need to copy **BOTH** the Client ID AND Client Secret. Use the copy buttons next to each one to copy them to your clipboard, do **NOT** try to highlight them (you cannot highlight the entire Client ID).


Use your API credentials in Akeeba Ticket System

Go to your site's administrator backend.

Click on Extensions, Plugins from the top menu.

Find and click on the Akeeba Ticket System - Fetch Email plugin.

Editing the Akeeba Ticket System - Fetch Email plugin

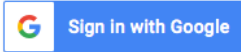
 Plugins: Akeeba Ticket System - Fetch Email

Plugin

Akeeba Ticket System - Fetch Email

ats / mailfetch

Downloads and processes email to create new tickets or ticket replies. Makes available the `mailfetch` CRON task. You need to set up your own CRON job to run this task.

Reply by email	<input type="radio"/> No <input checked="" type="radio"/> Yes
Only for managers	<input checked="" type="radio"/> No <input type="radio"/> Yes
Create ticket by email	<input type="radio"/> No <input checked="" type="radio"/> Yes
Track email UID	<input type="radio"/> No <input checked="" type="radio"/> Yes
Text version only	<input checked="" type="radio"/> No <input type="radio"/> Yes
Default category	<input type="text" value="Account issues"/> ▼
Mail server type	<input type="text" value="GMail / G Suite"/> ▼
Username	<input type="text" value="support@dionysopoulos.me"/>
API Client ID	<input type="text" value="418078923559-163ad561drjrptq5vs"/>
API Secret Key	<input type="text" value="mRBKOMd-PSbO8VKCnFkS4b6b"/>
Authorize GMail	
Access Token	<input type="text"/>
Refresh Token	<input type="text"/>
Delete after post creations	<input type="radio"/> No <input checked="" type="radio"/> Yes
InstantReply on new ticket	<input checked="" type="radio"/> No <input type="radio"/> Yes

On the screen, under Mail server type please choose GMail / G Suite

Under Username enter your email address.

Important

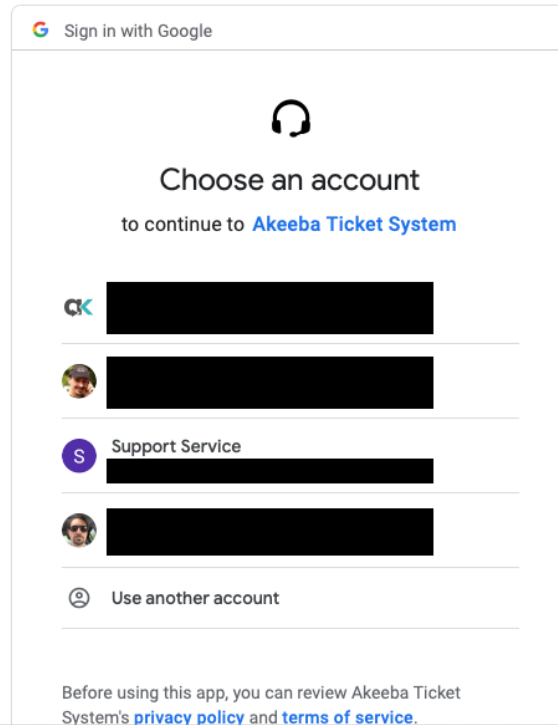
The email address you enter here **MUST** be the same one you will be authorizing in the next couple of steps. Otherwise ATS will be unable to retrieve any email.

In the API Client ID field paste the Client ID you copied from Google Cloud Console in the steps further above.

In the API Secret Key field paste the Secret Key you copied from Google Cloud Console in the steps further above.

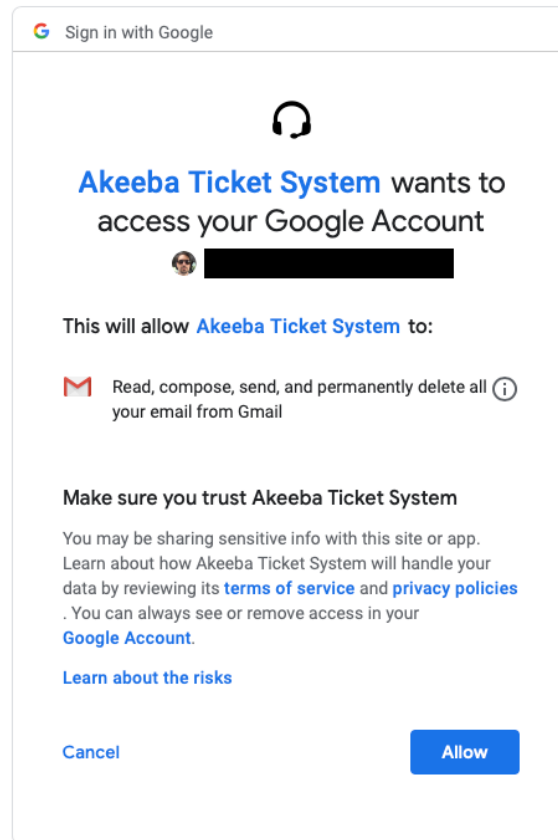
Next to Authorize GMail there is a Sign in with Google button. Click on it. A popup opens.

Gmail popup, select an account



Select the email account you want to link to Akeeba Ticket System. **It must match exactly the email address you typed in the Username field in the plugin configuration page.**

Gmail popup, final authorization



Click on Allow to complete the link.

The popup will redirect back to your site and you will see a "One moment please" message momentarily. The popup will automatically close and the Access Token and Refresh Token fields will be filled in.

Now click on Save & Close in the toolbar. You are done.

Important

If you have multiple sites DO NOT copy the Access Token and Refresh Token between them. It will NOT work.

10.2.2. Troubleshooting

Possible error on the popup: You are not authorized to view this resource

Have you entered the Client ID and Secret Key before clicking the Login with Google button?

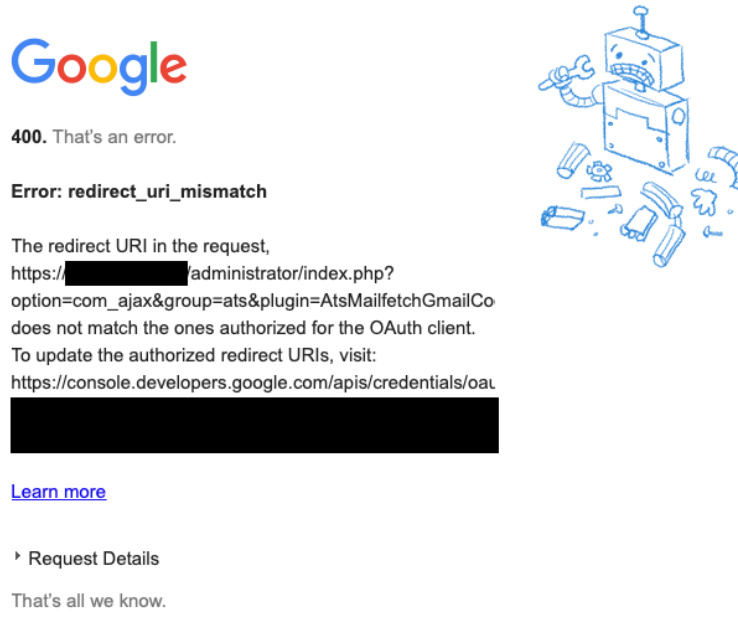
Are the Client ID and Secret Key correct, created for a web application OAuth2 flow as described above and they really, *really*, **REALLY** belong to the API application you created on Google Cloud Platform? Before dismissing this question as too basic – these are examples of mistakes made writing this documentation.

Has the user you are currently logged in with adequate Joomla privileges to edit the plugin options?

If you answered yes to all of the above then probably your Joomla session is broken. Log out from Joomla, log back in and try again.

Possible error on the popup: Google error 400

Error 400 on the popup



Google

400. That's an error.

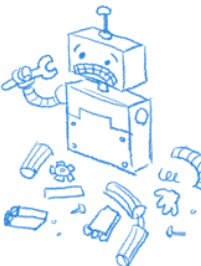
Error: redirect_uri_mismatch

The redirect URI in the request, `https://[REDACTED]administrator/index.php?option=com_ajax&group=ats&plugin=AtsMailfetchGmailCo` does not match the ones authorized for the OAuth client. To update the authorized redirect URIs, visit: `https://console.developers.google.com/apis/credentials/oa[REDACTED]`

[Learn more](#)

▸ Request Details

That's all we know.



If you receive an error 400 _from Google_ when the popup first opens the problem is that you have not entered the correct Authorized Redirect URIs as explained under “Get your API credentials”.

Please correct the redirect URL and try again.

Possible error on the popup: 403, 404 or Joomla error page

You might get an error 403 or 404 or a Joomla error page in the popup after you confirm that you want to link your email address to your application.

The problem is that your server is rejecting Google's redirection URL because it contains the URL parameter `scope=https://mail.google.com/`. As you can see, Google includes a full, unencoded URL which might trigger the server's protection.

If you are using our Admin Tools software and its .htaccess Maker, NginX Conf Maker or Web.config Maker feature please go to Components, Admin Tools, click on the .htaccess Maker, NginX Conf Maker or Web.config Maker button (depending on what you are using on your site) and set Protect against common file injection attacks to No. Then click on Save and Create .htaccess. Now you can retry linking your mail account to Akeeba Ticket System. After you're done you can set that option back to Yes if you want.

If you are not using Admin Tools or that did not help please contact your host. They might have to make a server configuration change, e.g. disable an Apache ModSecurity2 server protection rule.

10.2.3. Frequently asked questions

10.2.3. Why did you remove the old Gmail integration? It seemed to work fine.

Yes, it *did* work fine. However, Google decided to phase out the IMAP password authentication for Gmail and G Suite mail. This means that starting June 2020 it no longer works fine – it doesn't work *at all*. Implementing XOAUTH2 is a requirement imposed by Google, not Akeeba Ltd. If you are unhappy with that please let them know; we have done so already.

10.2.3. Other applications don't require me to go through that process to login to Gmail / G Suite. They just show a consent page. Why didn't you implement something like this?

This is exactly the approach we initially pursued. However, Google wouldn't allow us to do it.

As explained earlier, Access and Refresh Tokens are issued to an API application registered with Google. There are two kinds of applications. An Internal application – like the one created with the process explained above – only allows getting tokens for an email account that is the same as the one creating the account (Gmail) or belongs to the same organisation (G Suite). This wouldn't work for our clients. This meant that we had to create the other kind of API application, an External one. The External application is able to request tokens for any Gmail or G Suite account. However, External applications need to be verified by Google.

Normally, External applications are used for desktop applications, mobile apps or hosted web services. Akeeba Ticket System is none of the above. It's a self-hosted web application meant to be installed on your sites. The tokens are meant to be stored on our clients' sites, not our server. The emails are retrieved directly by our clients' sites, not through our server. However, Google requires us to host an application on our domain to retrieve the Access Token and, when it expires, exchange the Refresh Token with a new Access Token.

The only solution to that is something similar to what desktop / mobile apps like Thunderbird or Microsoft Outlook for Android do – or like what we are already doing in Akeeba Backup to integrate with Google Drive: a "mediator" application. That's a barebones application hosted on our site which handles the OAuth2 consent flow, retrieves the Access and refresh Tokens and conveys them to your site, without storing them on our server. This guarantees your privacy.

We wrote and tested that code and submitted our External API application to Google for verification. Unfortunately, Google decided to decline the verification of our External API application on the grounds that the site processing the email is not the same as the domain name registered with the application. Therefore they are rejecting the use case of a self-hosted ticket system using an easy method for authenticating to Gmail / G Suite.

Google gave us only one alternative: have an unverified application which could be whitelisted by G Suite administrators manually. This alternative is bad for two reasons. First and foremost, it makes it impossible for Gmail clients (with an @gmail.com email address) to use Akeeba Ticket System; that's the majority of our clients interested in connecting Akeeba Ticket System with a mail account hosted by Google. Second, the manual whitelisting process is just as complicated as creating your own Internal API application.

This left us with two options. One, we would have to stop supporting Gmail / G Suite in Akeeba Ticket System. Two, we would have to make you create your own Internal API applications with Google since they do not require going through the verification process. We chose the latter as the lesser of two evils.

We are fully aware that this process is complicated and hostile to the users. We did raise that point with Google, repeatedly, throughout the onerous verification process. Unfortunately, Google does not want to consider software like Akeeba Ticket System as valid consumers of email. In other words, email accounts hosted by Google come with artificial limitations imposed by Google. They are not real email accounts that can be used for any purpose their owners see fit. They are crippled by anti-features which are controlled by a commercial entity's opaque and arbitrary decision making process.

We strongly recommend not using a Google-controlled, crippled email account whenever possible. If your organisation / client is tied to G Suite then you will have to go through the complicated process of creating an Internal API application. Unfortunately, we can't be of much help because Google *doesn't allow third parties providing support for its products either*. Worse, yet, because of the nature of an Internal API application we cannot perform that process for you, not for free or even for a fee; it would require administrator access to your Gmail / G Suite account that you cannot reasonably provide and we cannot possibly accept to be given for reasons that have to do with GDPR and similar privacy legislation.

10.2.3.1.3 I'm not a G Suite admin (e.g. I am a site integrator working on behalf of my client). Can I set up Akeeba Ticket System to use the G Suite account?

No, you can't. That's a limitation due to Google's decision not to authorize our External project as described above.

10.2.3.1.4 I do not have access to my client's Gmail account. Can I set up Akeeba Ticket System to use their Gmail account?

No, you can't. That's a limitation due to Google's decision not to authorize our External project as described above.

10.2.3.1.5 I believe there is an easier way to connect to Gmail / G Suite email

Yes, you are right, there is. Unfortunately, Google denied verification for it. See the long explanation above.

10.2.3.1.6 How about the security, privacy and GDPR compliance of this integration?

Your OAuth2 API application information (client ID and secret key) and your access and refresh tokens are **only** stored on your own site, as plugin parameters. They are NOT shared with Akeeba Ltd or any other third party, nor do they ever leave your site. This means that you have perfect privacy of your support tickets.

Regarding security, your information is as secure as your Joomla! site itself. If your site gets hacked you should unlink your email address from your API application, disable the OAuth2 API credentials of your API application and create new ones, then relink ATS to your email account. A more secure alternative is forwarding your emails to a non-Gmail, non-G Suite email account that's accessible over regular IMAP with password authentication. In case of compromise you can simply change the password of that account and be done with it.

As for GDPR, CCPA and similar privacy legislation – please ask your lawyer. We cannot give a legal opinion. Our non-legal advice is that you should disclose that the emails sent to a particular email address are automatically processed by software running on your site for the purpose of creating new support tickets or replies to existing ones. If that has other privacy implications, e.g. third parties employed by / affiliated with your site have access to the ticket system, that should be disclosed as well. Again, it's best to ask a qualified lawyer.

10.2.3.1.7 Can I forward my Gmail / G Suite emails to a different account and access that account over IMAP / POP3 instead?

Yes, of course you can. Google has documented the automatic mail forwarding process already [<https://support.google.com/mail/answer/10957?hl=en>].

Once you set up your automatic forwarding to a regular email account, e.g. one provided by your hosting company, you can connect ATS to that email account over IMAP / POP3.

11. Customising the frontend layout

Akeeba Ticket System has a simple, clean design using either Bootstrap 5 or a custom CSS file. To make things simpler for site integrators we only use vanilla JavaScript, without relying on jQuery or any other framework, and core

Joomla features thus minimising the need to rework our component when using different templates. We also provide an abundance of custom module positions to allow site integrators to put additional information in Akeeba Ticket System pages beyond what the template they are using allows them to do.

While this is all great and flexible, we understand that there's no such thing as one-size-fits-all in web design. One of the greatest strengths of Joomla over any other PHP-based CMS is that it allows site integrators to easily customise the frontend experience with overrides. ATS is a native Joomla component and of course supports all of these overrides without much effort. Even better, we designed ATS *expecting* overrides to take place and made sure to provide features to help you towards that goal such as plenty of CSS classes and even the source SCSS files for our custom frontend CSS.

In this section we will discuss all the ways you can use to customise the look and feel of the frontend.

11.1. Component view template overrides

The ATS component's frontend is primarily generated by standard view template files. These are the PHP files in the subdirectories of the `components/com_ats/tmpl` folder under your site's root. You can override them as explained in the official Joomla documentation page on overriding the output from the Joomla! core [https://docs.joomla.org/How_to_override_the_output_from_the_Joomla!_core].

Some other parts of its output are generated through Layouts. These are PHP files in the subdirectories of the `components/com_ats/layouts` folder under your site's root. These typically are reusable and smaller pieces of content display. They are called by the view template files using the `LayoutHelper` helper class provided by Joomla. Overriding them is similar to overriding view templates and discussed in the [Layout overrides](#) section below.

But let's go back to the view templates. Joomla offers two ways to do that.

Using the Joomla Template Manager

This method is discussed in the Joomla documentation [https://docs.joomla.org/J3.x:How_to_use_the_Template_Manager#Creating_Overrides].

Go to System, Templates, Site Templates.

Select your site's template.

Click on the Create Overrides tab.

From the Components area click on `com_ats`. You can now select the view whose files you want to override. For example, selecting `ticket` will override the `Ticket` view which is used when creating a new ticket and when viewing or editing an existing ticket.

Back to the Editor tab, navigate to `html` then `com_ats`. You will see a new directory with the same name as the one you selected on the step above. Clicking on it you can see all the files Joomla copied over from our component and you can edit them.

These are your own copy of our view template files, living inside your site's template. You can modify them freely without fear of breaking ATS. If you mess up you can delete these files (or the entire subfolder) and start over.

Copying files manually

The Joomla template manager doesn't perform any magic; it simply copies over files. You can copy the files manually and edit them in a programmer's text editor (e.g. Notepad++ on Windows, gEdit or Kate on Linux, or BBEdit on macOS) or a full blown IDE such as Visual Studio Code, NetBeans, Eclipse, phpStorm etc.

You will only need to know where your site's template lives inside your site's filesystem. All templates live under the `templates` folder in your site's root. Inspect the subfolders there to find out which one has your template. In this documentation we will call that subfolder `MY_TEMPLATE`.

Go to the folder `components/com_ats/tmpl` and find the subfolder of the view and which file(s) you want to override. Yes, that's right, you DO NOT need to copy all files; you only need to copy the ones you actually want to override.

Let's say that you want to override `components/com_ats/tmpl/ticket/default_post.php`.

1. Create the subfolder `templates/MY_TEMPLATE/html/com_ats/ticket` You may have to create parent folders to get there.
2. Copy the file `components/com_ats/tmpl/ticket/default_post.php` to `templates/MY_TEMPLATE/html/com_ats/ticket/default_post.php`

You can now edit the file `templates/MY_TEMPLATE/html/com_ats/ticket` and customise it as you wish. It will be used INSTEAD OF the file we provide with ATS.

11.1.1. Custom fields in view template overrides

All of our views run the displayed items through Joomla's content plugins. This means that the category and ticket objects made available in our view templates will have the `jcfields` property containing the custom fields *as long as* the core Joomla Content - Fields plugin is published. Please remember that publishing this plugin and having its Access set to Public is a hard requirement by Joomla itself. This is the plugin which runs the core Joomla code to make custom fields available to any extension, be it a core Joomla extension such as Articles or a third party extension such as Akeeba Ticket System. In fact, the `jcfields` property is created and populated by this plugin. Our code has NOTHING to do with it at all; that's how Joomla is designed to work and we follow the Joomla best practices to the fullest extent.

By default, the `jcfields` property is an array ordered by the numeric field ID which is not particularly helpful. As explained in the Category custom fields and Ticket custom fields sections you can use the `getFieldsByName` helper in your template overrides to get an array sorted by the field's name (the alias, not the label!).

Please keep in mind when using the helper that your field names must be unique across all field groups. Otherwise it's hard to tell which field will be returned by the helper. Technically, it's the last field that Joomla knows about by this name. However the order by which Joomla will list fields so as to figure out which one is the last one depends on too many factors to make it ever remotely predictable. As we said, use unique names or you'll have troubles nobody can help you with.

11.1.2. Layout overrides

Layout overrides are similar to template overrides. The only thing that changes is the directory they are copied from. Joomla offers two ways to do that.

Using the Joomla Template Manager

This method is discussed in the Joomla documentation [https://docs.joomla.org/J3.x:How_to_use_the_Template_Manager#Creating_Overrides].

Go to System, Templates, Site Templates.

Select your site's template.

Click on the Create Overrides tab.

From the Layouts area click on `com_ats`. You can now select the layout subfolder whose files you want to override. For example, selecting `akeeba` will override the layouts we use for rendering small pieces of information such as the user's avatar.

Back to the Editor tab, navigate to `html` then `layouts`. You will see a new directory with the same name as the one you selected on the step above. Clicking on it you can see all the files Joomla copied over from our component and you can edit them.

These are your own copy of our layout template files, living inside your site's template. You can modify them freely without fear of breaking ATS. If you mess up you can delete these files (or the entire subfolder) and start over.

Copying files manually

The Joomla template manager doesn't perform any magic; it simply copies over files. You can copy the files manually and edit them in a programmer's text editor (e.g. Notepad++ on Windows, gEdit or Kate on Linux, or BBEdit on macOS) or a full blown IDE such as Visual Studio Code, NetBeans, Eclipse, PhpStorm etc.

You will only need to know where your site's template lives inside your site's filesystem. All templates live under the `templates` folder in your site's root. Inspect the subfolders there to find out which one has your template. In this documentation we will call that subfolder `MY_TEMPLATE`.

Go to the folder `components/com_ats/layouts` and find the subfolder and file(s) you want to override. Yes, that's right, you DO NOT need to copy all files; you only need to copy the ones you actually want to override.

Let's say that you want to override `components/com_ats/layouts/akeeba/ats/common/user.php`.

1. Create the subfolder `templates/MY_TEMPLATE/html/layouts/akeeba/ats/common` You may have to create parent folders to get there.
2. Copy the file `components/com_ats/layouts/akeeba/ats/common/user.php` to `templates/MY_TEMPLATE/html/layouts/akeeba/ats/common/user.php`

You can now edit the file `templates/MY_TEMPLATE/html/layouts/akeeba/ats/common/user.php` and customise it as you wish. It will be used INSTEAD OF the file we provide with ATS.

11.2. Module view template overrides

Our module's output is generated by view template files, like our component's. Just like a component, the module's view template files are stored in the `tmpl` directory of the module itself.

Keep in mind that there are two kinds of modules: site and administrator. Site modules' view templates can only be overridden in site templates. Administrator modules' view templates can only be overridden in *administrator templates*. If you copy an administrator module's view templates in a site template's `html` folder to override them nothing will happen. This is not a bug; you asked Joomla to do something that does not make sense if you think about it. We explicitly mention that because it's something that less experienced site integrators fall for the first time they try to do template overrides.

In any case, Joomla offers two ways to do module view template overrides.

Using the Joomla Template Manager

This method is discussed in the Joomla documentation [https://docs.joomla.org/J3.x:How_to_use_the_Template_Manager#Creating_Overrides].

For frontend modules: go to System, Templates, Site Templates. For backend modules: go to System, Templates, Administrator Templates.

Click on the Create Overrides tab.

From the Modules area click on the module you want to override view templates for.

Back to the Editor tab, navigate to `html` then `mod_something`, where `mod_something` is the name of the module you selected in the previous step. You can see all the files Joomla copied over from our component and you can edit them.

These are your own copy of our view template files, living inside your site's template. You can modify them freely without fear of breaking ATS. If you mess up you can delete these files (or the entire subfolder) and start over.

Copying files manually

The Joomla template manager doesn't perform any magic; it simply copies over files. You can copy the files manually and edit them in a programmer's text editor (e.g. Notepad++ on Windows, gEdit or Kate on Linux, or BBEdit on macOS) or a full blown IDE such as Visual Studio Code, NetBeans, Eclipse, PhpStorm etc.

You will only need to know where your site's template lives inside your site's filesystem. All site templates live under the `templates` folder in your site's root and all administrator templates live under the `administrator/templates` folder under your site's root. Inspect the subfolders there to find out which one has your template. In this documentation we will call that subfolder `MY_TEMPLATE`.

Go to the folder `modules/mod_something` (frontend modules) or `administrator/modules/mod_something` (backend modules) where `mod_something` is the module you want to override view templates for. Find the `tmpl` folder and which file(s) you want to override. Yes, that's right, you DO NOT need to copy all `.php` files if there are more than one; you only need to copy the ones you actually want to override.

Let's say that you want to override the frontend module file `modules/mod_atstickets/tmpl/default.php`.

1. Create the subfolder `templates/MY_TEMPLATE/html/mod_atstickets` You may have to create parent folders to get there.
2. Copy the file `modules/mod_atstickets/tmpl/default.php` to `templates/MY_TEMPLATE/html/mod_atstickets/default.php`

You can now edit the file in `templates/MY_TEMPLATE/html/mod_atstickets` and customise it as you wish. It will be used INSTEAD OF the file we provide with ATS.

Let's say that you want to override the backend module file `administrator/modules/mod_atstats/tmpl/default.php`.

1. Create the subfolder `administrator/templates/MY_TEMPLATE/html/mod_atstats` You may have to create parent folders to get there.
2. Copy the file `administrator/modules/mod_atstats/tmpl/default.php` to `administrator/templates/MY_TEMPLATE/html/mod_atstats/default.php`

You can now edit the file in `administrator/templates/MY_TEMPLATE/html/mod_atstats` and customise it as you wish. It will be used INSTEAD OF the file we provide with ATS. Even if you are using the default Joomla administrator template (Atum) this file will NOT be overwritten when you update Joomla; it's not part of Joomla's distribution.

11.3. Media (CSS, JavaScript and image) overrides

Overriding the HTML output of the component is one thing. Most of the times you don't really want to do that, you just want to customise the CSS, an image or make small changes to the JavaScript. These are all static files and yes, it's possible to override them without making changes to the component itself. This means that any changes you make will survive an update of the component since they live in your template's directory, not your site. These are called media overrides in Joomla lingo.

Joomla currently provides one way of doing that, copying file manually.

There are two things you need to know.

1. Where are the original files. They are kept in the `media/com_ats` folder under your site's root.
2. Where to put the overrides. You need to put them in the `templates/MY_TEMPLATE/media/com_ats` folder, where `MY_TEMPLATE` is your site template's folder.

There are a few notes about how these files work.

CSS files

The custom CSS file `frontend.css` is only loaded if you set the Load custom CSS option in Akeeba Ticket System's Options page to Yes.

Do not try to edit the CSS file directly; it's been automatically generated by parsing the SCSS files in the `media/com_ats/css` folder. We strongly recommend that you work on the SCSS files in your media override directory and use an SCSS compiler to convert them to CSS. You can of course forego our custom CSS completely and create your own `frontend.css` file any way you want.

Make sure your file is called `frontend.css`, not `frontend.min.css` or anything else.

JavaScript files

You will see that there are two JavaScript files with the same name, one with the extension `.js` and one with the extension `.min.js`. For example, `CannedReply.js` and `CannedReply.min.js`. Joomla normally loads the `.min.js` file. If you set Debug Site to Yes in your site's Global Configuration it will, however, try to load the `.js` file instead. Therefore if you need to override any JavaScript remember to override BOTH files to prevent surprises.

We use Babel to minify our `.js` files into `.min.js`. If you don't know how to use Babel it's okay to just copy the `.js` file with a `.min.js` extension.

Image files

No special handling is necessary. Just make sure that the file name and extension cases matches. That is to say, `image.gif`, `Image.gif` and `image.GIF` are three different files. The files we ship have lowercase names and extensions.

11.4. Language overrides

In many cases you don't want to override the entire frontend display of the component, just change the wording. No text in Akeeba Ticket System is hard-coded. Everything is contained in standard Joomla language files. You can change specific language strings using standard Joomla language overrides [https://docs.joomla.org/J3.x:Language_Overrides_in_Joomla].

12. Custom module positions in the front-end

Akeeba Ticket System allows you to customise its pages by publishing modules in its predefined module positions. These module positions are not related to those used by your template; they are defined and rendered in the view templates of the component itself.

Using custom module positions

Go to Joomla's Content, Site Modules menu item.

Find the module you want to publish in a custom ATS module position or create a new one.

In the edit page of the module, look for the Position field. It's in the sidebar, right below the Title show/hide toggle.

Click inside the Position dropdown, not its downwards arrow. You will see that it's a text field where you can enter free text.

Type the custom module position name e.g. **ats-top** and press **ENTER** or **RETURN** on your keyboard.

That's it! You have just told Joomla to use a custom module position even though it's not listed in the Position dropdown.

Custom module position names

Below you will find out what each custom module position is and where/when it is shown.

ats-top	At the top of each page of the component
ats-categories-top	At the top of the categories list page, below ats-top
ats-mytickets-top	At the top of the My Tickets page, below ats-top
ats-newticket-top	At the top of the new ticket page, below ats-top
ats-posts-top	At the top of the post page, below ats-top
ats-tickets-top	At the top of the tickets in a category list, below ats-top
ats-categories-none-top	Above the "no categories" message when no categories are found, below ats-categories-top This module position is only shown when there are no categories defined in Akeeba Ticket System or when the privileges of the currently logged in user do not allow him to access any of the Akeeba Ticket System categories.
ats-mytickets-none-top	Above the "no tickets" message when no tickets are found in the My Tickets view, below ats-mytickets-top This module position is only shown when there are no tickets submitted yet by the currently logged in user.
ats-tickets-none-top	Above the "no tickets" message when no tickets are found in the tickets in category list, below ats-tickets-top This module position is only shown where there are no tickets posted yet in a category or when all the tickets posted in the category are private and not visible by the currently logged in user..
ats-categories-none-bottom	Below the "no categories" message when no categories are found, above ats-categories-bottom This module position is only shown when there are no categories defined in Akeeba Ticket System or when the privileges of the currently logged in user do not allow him to access any of the Akeeba Ticket System categories.
ats-mytickets-none-bottom	Below the "no tickets" message when no tickets are found in the My Tickets view, above ats-mytickets-bottom This module position is only shown when there are no tickets submitted yet by the currently logged in user.
ats-tickets-none-bottom	Below the "no tickets" message when no tickets are found in the tickets in category list, above ats-tickets-bottom

	This module position is only shown where there are no tickets posted yet in a category or when all the tickets posted in the category are private and not visible by the currently logged in user..
ats-categories-bottom	Below the categories list page, above ats-bottom
ats-mytickets-bottom	Below the My Tickets page, above ats-bottom
ats-newticket-bottom	At the bottom of the new ticket page, above ats-bottom
ats-posts-bottom	At the bottom of the post edit page, above ats-bottom
ats-tickets-bottom	At the bottom of the tickets in a category list, above ats-bottom
ats-bottom	At the bottom of each page of the component
ats-replyarea-overlay	Overlaid on top of the message area when composing a new post or replying to an existing one. If "No new tickets" or "No new replies" is enabled in the component's Options page it will also be shown above the ats-nonewtickets and the ats-noreplies positions respectively.
ats-nonewtickets	Shown instead of the New Post page when "No new tickets" is enabled in the component's Options page.
ats-noreplies	Shown instead of the reply area are when "No new replies" is enabled in the component's Options page.

13. CRON jobs to set the ticket system offline

Akeeba Ticket System 5 no longer has the Off-line Schedule feature which allowed to you to put the ticket system off-line (disallow new tickets and new ticket replies) on a schedule.

Instead, we provide integration with Joomla's command-line application, `cli/joomla.php`, and a command which allows you to change the component Options.

You can schedule the execution of these commands using CRON jobs. CRON jobs can be defined using your host's tools, typically cPanel or Plesk.

You will need to know two things before you start (if unsure, ask your host about these):

1. The absolute filesystem path to your site's root. For the purpose of this section we assume it is `/home/mysite/public_html`.
2. The absolute filesystem path to the PHP CLI binary. This is entirely host specific. Make sure to ask your host about the same PHP version you are using on the site itself — most hosts provide a wide choice of PHP versions at the same time. For the purpose of this section we assume it is `/usr/local/bin/php`.

To put the ticket system off-line you need to execute two commands. One to set No New Tickets (prevents submitting new tickets) and one to set No New Replies (prevents submitting new replies):

```
/usr/local/bin/php /home/mysite/public_html/cli/joomla.php ats:config \  
-k nonewtickets -a 1
```

```
/usr/local/bin/php /home/mysite/public_html/cli/joomla.php ats:config \  
-k noreplies -a 1
```

To put the ticket system on-line you need to execute two commands. One to disable No New Tickets (allowing the submission of new tickets) and one to disable No New Replies (allowing the submission of new replies):

```
/usr/local/bin/php /home/mysite/public_html/cli/joomla.php ats:config \  
-k nonewtickets -a 0
```

```
/usr/local/bin/php /home/mysite/public_html/cli/joomla.php ats:config \  
-k noreplies -a 0
```

Chapter 4. The Component

1. The Control Panel page

Accessible through the back-end Components, Akeeba Ticket System, Control Panel menu item.

Akeeba Ticket System's Control Panel

The screenshot shows the Akeeba Ticket System Control Panel. At the top right, there is an "Options" button with a gear icon. Below this is a line graph showing ticket activity from Oct 13 to Nov 12. The y-axis ranges from 0 to 1.0. The data points are all at 0. To the right of the graph is a table titled "Ticket Statistics".

	Open	Pending	Closed	Total
Public	4	207	4075	4286
Private	2	81	1744	1827
Total	6	288	5819	6113

Below the graph and table is a section titled "Main operations" with three buttons: "Tickets", "Time cards", and "CRON Logs". Below that is a section titled "Ticket System Configuration" with six buttons: "Categories", "Field Groups", "Fields", "Email Templates", "Auto-replies", and "Canned replies". Below that is a button for "Automation". At the bottom is a blue bar titled "Version information" containing the text "Akeeba Ticket System Professional rev85B470CC" and a "Changelog" link. Below the version information is the copyright notice: "Copyright ©2010–2021 Nicholas K. Dionysopoulos / Akeeba Ltd. All Legal Rights Reserved. Akeeba Ticket System is Free software released under the GNU General Public License, version 3 of the license or – at your option – any later version published by the Free Software Foundation."

The Control Panel page gives you an overview of your help desk and allows you to quickly reach the configuration pages for the different Akeeba Ticket System features.

The graph at the top shows you the number of new tickets and posts submitted on each calendar day over the last 30 days.

The table next to the graph shows you the number of tickets depending on their visibility (Public or Private) and status. Please note that if you are using custom ticket states they are all lumped under the Pending figure for screen space efficiency reasons.

In the area below the graph and stats tables you can see quick icons to the main ticket system features and configuration.

Finally, there is a panel displaying the current version information and copyright notice — the latter being a requirement of the license we publish our software under, the GNU General Public License version 3.

This page is one of the two pages — the other being the Tickets page — where you can find the Options button which opens the component's options page.

2. Component Options

Note

Some of the features may only apply to the for-a-fee Akeeba Ticket System Professional edition.

The Options page is accessible through the back-end Components, Akeeba Ticket System, Control Panel menu item and then clicking on the Options button in the component's toolbar. These options define how Akeeba Ticket System behaves. This page has several sections.

The Options page is NOT handled by Akeeba Ticket System itself; it's handled by Joomla itself through its `com_config` core component. This is the same core component which handles the Global Configuration of your site and the options pages for all core and third party components.

2.1. Common

These are basic options which control the overall operation of the component and its related features and extensions.

Enable ticket priorities Allows the users to set the priority of their ticket (low, normal, height). By default, Public tickets gets normal priority and Private tickets get High priority. When this option is disabled ATS will not show any ticket priority information.

Country Field Select a custom Field you have already created in the user profile which lets the user select their country. This information will be used to display a flag next to the username displayed along with a user's post in the Ticket view. For more information read our documentation page on creating and using a country user field.

Avatar Field Select a custom Field you have already created in the user profile which lets the user upload or select their user profile image (avatar). This information will be used to display the avatar above the username displayed along with a user's post in the Ticket view. For more information read our documentation page on creating and using an avatar user field.

Display user groups Displays the user groups the user belongs to under their username displayed along with a user's post in the Ticket view. This information is presented ONLY when the tickets is viewed by a user who is granted the Support Staff permission or is a Super User. This is intentional, to prevent leaking the names of the user groups on your site to the general public.

Send Emails Should ATS send e-mails? E-mails are sent when new tickets or ticket replies submitted, when the ticket information is edited by a user who does not have the Support Staff permissions, when a ticket is assigned to a manager or a post is edited.

Regardless of this setting, Akeeba Ticket System Core (free version) will NOT send e-mails.

Only email assigned managers If a ticket is assigned to a specific user the other users with the Support Staff permission on the category the ticket belongs to will not be emailed about replies to the ticket.

Custom ticket statuses You can create up to 99 extra ticket statuses on top of the default three (Open, Pending and Closed). Here you can define the custom ticket statuses.

Each row defines a custom ticket status. The ID is a number from 1 to 99 which internally represents the custom status. The custom statuses are displayed ordered by ID ascending. Please make sure that you don't reuse the same ID twice; if you do only the last instance of that custom ticket status will be taken into account. The Label is how this ticket status will be displayed to users and support staff.

Enable anonymous PHP, MySQL and Joomla! version reporting Allow Akeeba Ticket System to collect an anonymous report about the ATS, PHP, MySQL and Joomla! version you are currently using. This information is sent to us at most once a week. We DO NOT collect any identifiable information such as IP addresses, your site's URL / name, your Download ID / username or any other information which would allow us to correlate a report record with a specific site or user.

A random, unique ID generated on your site is sent along with these statistics. Akeeba Ltd does NOT have a way to correlate this ID with a specific site or user account. The ID is only used to discard all but the last report coming from each site when aggregating the version information at the end of each calendar month.

The individual data points are aggregated at the end of each month. The original report data is discarded immediately after it's processed. That is to say, the information reported by your site is stored pseudonymised for 1 to 31 days and then it's fully anonymised in accordance with the EU GDPR legislation.

We use this information to understand the server environments of our clients and decide when to drop support for older versions of PHP, MySQL and Joomla.

2.2. Frontend

These options control how the component is displayed in the frontend of your site.

No new tickets Nobody will be allowed to create a new ticket in the frontend of the site. This does not affect their ability to reply to existing tickets which have not already been closed.

No new replies Nobody will be allowed to reply to existing tickets which have not already been closed in the frontend of your site. This does not affect their ability to submit new tickets.

Maximum post size (Kb) The maximum allowed size of the ticket's body text, in Kilobytes. Tickets larger than this will be rejected.

The ticket body size is calculated without taking into account any attachments. This feature only applies to tickets and ticket replies submitted through the web interface; tickets submitted by email are not subject to this check. This feature does NOT apply to support staff; the support staff can submit tickets and replies of any length.

This feature is meant to prevent extremely large posts, e.g. a client posting an entire log file inline instead of posting as an attachment. Overly long posts not only clog the database with unnecessary data, it will also make it very slow for the browser to display the page. The default value for this option is 128Kb, determined by examining over a decade of tickets and hundreds of thousands of posts on our site, adding a bit of wiggle room for the larger posts where an attachment would not necessarily make sense. For most sites you can set this as low as 32Kb without causing problems for your clients. If you have a use case which requires large inline text you can increase it up to 10240Kb (10Mb). Please note that post sizes over 512Kb may cause severe performance issues when displaying tickets.

When a ticket is over the limit the client will receive an error with the language string `COM_ATS_POST_ERR_BODY_TOO_BIG`. You can use Joomla's Language Override feature to customise this error message.

Own Ticket/Post Editing Grace Time Only applies to user who have not been granted the Edit Own Ticket or Edit Own Post permission.

Setting this to a non-zero value allows all users to edit their ticket information and/or their post for a period of time after originally submitting it. This allows them to rectify some obvious mistake e.g. make an accidentally Public ticket Private, fix a typo in the ticket title or post, include some information they forgot to include the first time around, retry uploading an attachment that failed uploading etc. This option controls how many minutes after they first submitted their ticket / post they will be able to edit it.

Setting it to 0 disables this feature. Only users granted the Edit Own Ticket permission will be able to edit their ticket information and only users granted the Edit Own Post permission will be able to edit their posts.

Time spent field mandatory Makes the Time Spent field mandatory for Support Staff users. This means that support staff won't be able to submit a ticket reply unless they fill in a non-zero time spent answering the ticket.

If you enable this option it's worth reminding your support staff users that they can and should use the Start Timer button next to the Time Spent field to automatically count the time they are spending on a ticket.

CAPTCHA for Guest tickets Only applies to Akeeba Ticket System categories which allow the Create privilege to the Guest (not logged in user) group.

Choose which CAPTCHA plugin should be used in the New Ticket page when a Guest user is submitting a ticket. The "Use Global" option tells Akeeba Ticket System to use the CAPTCHA plugin selected in Joomla's Global Configuration. The "- None Selected -" option tells Akeeba Ticket System to not show a CAPTCHA at all.

Please remember that you need to publish the plugin selected AND make sure that its Access is set to Public or Guest. Any other access level will make it impossible for Joomla! to display a CAPTCHA to Guest users and it may make it impossible to file tickets as a Guest. This is how Joomla itself works, not a limitation of Akeeba Ticket System.

Force user creation when filing Guest tickets Only applies to Akeeba Ticket System categories which allow the Create privilege to the Guest (not logged in user) group.

Normally, Akeeba Ticket System will honor the Options of Joomla's Users component. If you have disabled user registration Guests will not be able to file new tickets. If you have allowed user registration but require account activation (either by the users themselves or by an administrator) the relevant activation emails will be sent out.

When this option is enabled Akeeba Ticket System will *ignore* Joomla's settings. Filing tickets as a Guest will always result in a new, activated user being created.

Load custom CSS Akeeba Ticket System's frontend is using the CSS classes provided by Bootstrap 5 to style its output. This was a conscious decision; Bootstrap 5 is Joomla's default CSS framework. Using Joomla's default framework means we do not need to load yet another CSS file for our extension. This makes your site load faster.

If your site's frontend template is NOT using Bootstrap 5 then Akeeba Ticket System will appear unstyled. To fix that, we have this option here. Enabling it tells Akeeba Ticket System to load the

file `media/com_ats/css/frontend.css` (or its media override) in the frontend of your site.

If you enable this option you most likely want to customise the CSS or even do template overrides for the component itself to better suit your site.

2.3. Categories View

These options apply to the main, top-level list of Categories (Categories view). It also applies at the subcategories area displayed at the top of a page displaying the tickets of a category.

These options can be overridden in the menu item's parameters.

No Categories message	Show a message notifying the user that there are no categories or they do not have the permissions to view any of the existing categories.
Title	Show the title of each category.
Linked titles	Makes the category title a link to the category, i.e. to the page which displays the tickets in that category. It obviously requires setting the Title option above to Show.
Show buttons	Display the View Tickets and New Ticket buttons next to the category's description.
Category description	Show the category description.
Category image	Show the category image. The category image will be shown next to the description; changing the image display requires a template override.
Tags	Show the category's tags below the category title, above the description.
Show subcategories	If the category has subcategories, show a tree of the subcategories below the category's description.
Subcategory levels	How many nesting levels of subcategories to display when the Show Subcategories option above is enabled. Use "All" to display all subcategory levels. Using "None" has the same effect as setting Show Subcategories to Hide. It usually makes sense to set this to 1 level.
Show subcategory description	Show the description of each subcategory under its title. Enabling this option may lead to very long, hard to read pages.

2.4. Tickets view

These options apply to the main, top-level list of Categories (Categories view). It also applies at the subcategories area displayed at the top of a page displaying the tickets of a category.

These options can be overridden in the menu item's parameters.

Title	Show the category title at the top of the page
Parent category	Show the parent category under the category title. This should be unnecessary in most cases; Akeeba Ticket System populates the breadcrumbs with the path to the category. You will realistically only need this option if you are creating menu items for each individual ticket category and/or in a way where your menu structure does not follow your nested category structure.

Tags	Show the category tags above the list of tickets.
Category description	Show the category description above the list of tickets.
Category image	The the category image next to the category description.
Show subcategories	If the category has subcategories, show a tree of the subcategories below the category's description.
Subcategory levels	How many nesting levels of subcategories to display when the Show Subcategories option above is enabled. Use “All” to display all subcategory levels. Using “None” has the same effect as setting Show Subcategories to Hide. It usually makes sense to set this to 1 level.
Show subcategory description	Show the description of each subcategory under its title. Enabling this option may lead to very long, hard to read pages.
New ticket button	Show the New Ticket button above the list of tickets.
Show tickets	Show the list of tickets in the category. You are advised to always set this to Show in the category options. You can change that globally or at the menu item if you have a very specific use case such as: <ul style="list-style-type: none"> • You have nested categories where one or more of the parent category levels is there for organisational purposes only and must not accept tickets itself. In this case change this option at the menu item level. • You do not have public tickets and you only ever want clients to access their tickets through the My Tickets page — and support staff only see the latest open tickets, not the full list of tickets ever. In this case you may want to display the categories structure but not list the tickets under each category. To do that, set this option to Hide in the component options.
# tickets to list	How many tickets to list by default per page. The “Use Global” option will use the List Limit setting in your site's Global Configuration. We very strongly recommend not showing more than 50 tickets per page for performance reasons. Kindly note that showing too many tickets per page, e.g. 100 or more, may result in PHP memory running out and your site throwing an error.
Sort table by	Choose the sort order for the tickets. The default is “Last replied, descending” showing the tickets with the most recent activity first. This is different than “ID, descending” which shows the ticket submitted most recently first, essentially letting older, still active tickets end up in later pages even if they have more recent activity!
Show No Tickets message when empty	Show a message that the category has no tickets or the user does not have permission to access any tickets when there are no tickets to show to the user.
Pagination	Display the pagination controls. These are the page links at the bottom of the page. It is generally advised to always have this set to Show.
Display select	Show a drop–down menu which lets the user select how many tickets to display per page. THIS IS DANGEROUS. Joomla includes very large numbers (100 or more) in the drop–down. As noted above, trying to display more than 50 tickets may become excessively slow or even cause a memory outage error, causing your site to display an error. The client will be unable to reset

this until the log out and back in. As a result you should NOT give them the option to change the number of tickets per page.

Pagination summary Show the “Page X of Y” message next to the page number links. It is not strictly necessary — this information can be easily inferred from the pagination links. It's just a bit clearer for your clients.

2.5. InstantSearch

Akeeba Ticket System can display Smart Search or search engine results when your clients are submitting a ticket. This can help them find help faster, without submitting a ticket.

For more information on how this all works and the options in this section please consult the InstantSearch documentation.

2.6. Attachments

These options control the behavior of attachments and attachments uploads. A lot of these options are identical to those used by Joomla! in the Media Manager Options and work exactly the same way, but they only apply to Akeeba Ticket System. This allows you to have a different set of uploads handling for ticket attachments than media files in general on your site.

Please note that attachments are only available in Akeeba Ticket System Professional.

Attachments folder Enter the path to the folder where ATS will save its attachments. The path must be relative to your site's root, it must be contained in your site's root (you cannot use a directory outside your site's root), writeable and must not contain two dots anywhere in its name.

Attachments are stored with “mangled” filenames in the form of abcdef0123456789abcdef0123456789abcdef0123456789. These file names are generated semi-randomly during upload. Further to that and to avoid filesystem performance issues the attachments are stored two directory levels deep, the directory names corresponding to a. the first two and b. the third and fourth letters of the mangled filename, e.g. ab/cd/abcdef0123456789abcdef0123456789abcdef0123456789.

Private attachments Makes all attachments visible only to support staff and the user who filed the ticket they belong to *even if the ticket itself is Public*.

Allow unsafe uploads **WARNING! THIS MAY HAVE A DETRIMENTAL IMPACT ON YOUR SITE'S SECURITY!**

This option tells Akeeba Ticket System to blindly accept any and all downloads that your server and PHP itself allowed to go through, without performing any further security checks. This could allow a user to upload a malicious file accidentally or on purpose. That could put your site at risk or may even result in your site being suspended by your hosting company.

Enabling this option disables and hides all options below it.

Maximum size (in Mb) The maximum size (in Megabytes) of each individual file. Use zero for no limit.

This option will have no effect is it's bigger than the `post_max_size` or `max_upload_size` PHP configuration values. In simple terms, you can apply a value *smaller* than either of these PHP options but NOT a value bigger than either of these PHP options.

Restrict Uploads Perform additional checks on the attachment's original name and/or MIME Type before deciding whether to accept it.

Setting this option to No disables and hides all options below it.

Allowed executable extensions	<p>Joomla blocks by default files which have an “executable” extension anywhere in their file name. Adding these extensions here will unblock uploading files with these extensions in Akeeba Ticket System. It's recommended to leave the default list. This is safe; ATS stores the files with a randomised filename without a file extension i.e. they cannot be executed on your server.</p> <p>Use a comma separated list. The default list is:</p> <p>js, exe, dll, go, ade, adp, bat, chm, cmd, com, cpl, hta, ins, isp, jse, lib, mde, msc, msp, mst, pif, scr, sct, shb', sys, vb, vbe, vbs, vxd, wsc, wsf, wsh, html, htm, msi, php, phps, pht, phtml, php3, php4, php5, php6, php7, asp', php8, phar, inc, pl, cgi, fcgi, java, jar, py, aspx</p>
Allowed extensions	<p>Only files with one of these extensions can be uploaded. Extensions in this list are case-sensitive i.e. ZIP, Zip and zip are three different file extensions.</p>
Check MIME Types	<p>Use MIME Magic or Fileinfo to verify the file type on top of checking the file extension. It generally not recommended using this option for ticket attachments.</p> <p>Setting this option to No disables and hides all options below it.</p>
Ignored Extensions for MIME Type checks	<p>Files with these extensions will NOT have their MIME type checked.</p>
Legal MIME Types	<p>A comma separated list of MIME types. Files matching this list will be allowed to be uploaded. Files NOT matching this list will be rejected (won't upload).</p>

2.7. Security

These options affect the security of the ticket system.

Filtering method	<p>Akeeba Ticket System allows your clients to submit their ticket and reply content using Joomla's WYSIWYG editor which returns HTML. Likewise, accepting new tickets or ticket replies by email means that a client can submit an HTML-formatted ticket. If this HTML is left unfiltered there's a very chance that either accidentally or out of malice someone will exploit this to launch an XSS (Cross Site Scripting) attack which could compromise your site. ATS deals with it by filtering the incoming HTML. There are three filtering options:</p>
HTML Purifier (best protection)	<p>This uses the third party HTML Purifier library. It's marginally slower but provides the very best protection you can get.</p>
Joomla!	<p>This uses Joomla!'s own HTML sanitiser. It's good, it's fast. It's not as good as HTML Purifier, though. Only use this option if HTML Purifier seems to remove content you really want to allow but you do not understand how to configure it.</p>
I want my site to be hacked (no protection)	<p>This options disables protection. The name of this option is deliberately provocative — and accurate. Unless you have</p> <p>This option is reserved for people who want their site to get hacked and developers who believe they've found a better filtering method than HTML Purifier and don't mind being hacked to disprove their point. No joking here. This option turns off all filtering. It's like jumping off a plane without a parachute. DON'T DO IT! It's not a question of whether you're going to get hacked. It's a simple question of when you'll get hacked.</p>

HTML Purifier allowed tags For advanced users only. You get to specify which tags and attributes will be kept by the HTML Purifier filter. The default value is:

p,b,a[href],i,u,strong,em,small,big,span[style],font[size],font[color],ul,ol,li,br,img[src],img[width],img[height],code

Do not change unless you know what you are doing. If you remove everything from the list the default value will be used (otherwise all posts would end up blank).

2.8. Automation

These options modify the way Akeeba Ticket System's custom CRON URL works.

We strongly recommend using our integration with Joomla's CLI application (`cli/joomla.php`) instead of the CRON URL whenever possible. If you are only using a CLI-based CRON script you do not need to change any of the options here.

Secret Key Enter a secret key that consists of letters a-z (without accents or diacritics) and numbers 0-9. Make sure it's at least 24 characters long. This key will need to be present in the CRON URL for it to work.

If unsure, you can use the Random.org password generator [<https://www.random.org/passwords/?num=1&len=24&format=html&rnd=new>] to create a truly random secret key.

Warning

Do not use special characters, such as those delivered by pressing SHIFT and the numbers keys at the top of a US layout keyboard (e.g. !, @, # and so on, especially the characters ?, & , % and =), Latin characters with diacritics (e.g. ü, ê, ø, ç, š, # and so on), compound characters / ligatures (e.g. æ, ß, œ, # and so on) or non-Latin characters (e.g. μ, # and so on). These characters would need to be URL-encoded to work; failing to do so could cause your CRON URL to not work.

CRON URL time limit The execution of CRON commands using the CRON URL will be limited to approximately this many seconds. The default value, 10, is good enough for most sites. If you get server errors when accessing the CRON URL set this lower, e.g. 5. Anything below 3 is practically unusable.

Do note that the **mailfetch** command may take more than this many seconds. After retrieving all unread email it will process at least one email before checking the time limit. This might be longer than the time limit defined here and it's deliberate. Fetching email may take a few or several seconds. If the time limit was checked before processing at least one email and the time limit had already been reached while fetching email messages you'd end up never processing any email at all. This would make the new ticket by email and reply by email feature essentially non-functional.

Why not integrate with Joomla's Scheduled Tasks?

At the time we wrote Akeeba Ticket System 5.0 (November 2021) Joomla's Scheduled Tasks was still a feature proposed for inclusion in Joomla 4.1. However, it wasn't ready and we could not write an integration for it. We will work on integrating Joomla's Scheduled Tasks in Akeeba Ticket System after February 2022, when Joomla 4.1 is released.

2.9. Permissions

In this section you can determine the default permissions of each Joomla! User Group for all Akeeba Ticket System categories. If you prefer to define these settings per category remember to NEVER use a Deny rule here. A Deny rule

here will override Allow in children user groups and categories. If you want to deny access just leave the default value, Inherited. Inherited (denoted by a faded "no entry" symbol next to it) is also known as a "soft deny" and will deny access unless you provide an explicit Allow in a child User Group or a category.

Configure ACL & Options	Allows the user to access the Options page of the interface and change the Permissions.
Support Staff	<p>The user is a member of the support staff a.k.a. a ticket manager. It allows the user to view and reply to any ticket, even Private tickets; submit attachments; submit new tickets, public and private; edit, delete, publish and unpublish tickets, posts and attachments; make tickets public or private; change the ticket status; edit ticket properties; access Canned Replies; assign and be assigned to tickets; receive email notifications.</p> <p>When a user has this permission it overrides the settings for the Create, Delete, Edit, Edit Own, Edit Own Ticket, Edit Own Post, Edit State, Create Private and Create Attachment permissions.</p> <p>When this permission is allowed in the component options (as opposed to just in a category) it also allows the user to access the backend interface of Akeeba Ticket System i.e. it acts exactly like the "Access Administration Interface" permission in other components.</p> <p>Since this permission corresponds to Joomla's core.manage permission by default only the Administrator and Super User groups (or other groups explicitly granted the Access Administrator Interface / core.manage permission globally) will be able to manage tickets in Akeeba Ticket System.</p>
Create	The user can create new tickets and reply to their own tickets in the frontend. The user can create new items of any type in the backend.
Delete	The user can delete tickets, posts and attachments.
Edit	The user can edit items, including tickets and posts, created by anyone — be it themselves or a different user.
Edit Own	The user can edit items other than tickets and posts which they created themselves in the backend interface of Akeeba Ticket System. For example, canned replies and automatic replies.
Edit Own Ticket	The user can edit the ticket information on tickets they submitted themselves (the limits set in the Frontend options do not apply to these users).
Edit Own Post	The user can edit the posts on tickets they submitted themselves (the limits set in the Frontend options do not apply to these users).
Edit State	The user can publish/unpublish any item in the backend or frontend of Akeeba Ticket System.
Create Private	The user is allowed to create private tickets. Requires the Create permission.
Create Attachment	The user is allowed to upload attachments. This permission only takes effect in the Professional version.
Support Staff	<p>The user is a member of the support staff. It allows them to view and reply to any ticket, even Private tickets; submit attachments; submit new tickets, public and private; edit, delete, publish and unpublish tickets, posts and attachments; make tickets public or private; change the ticket status; edit ticket properties; access Canned Replies; assign and be assigned to tickets; receive email notifications.</p> <p>When this permissions grants user the ability to do something (such as editing a ticket) but the other permissions deny it this permission wins. That is to say, giving someone the support staff</p>

permission will override a hard Deny setting on Create, Delete, Edit, Edit Own, Edit Own Ticket, Edit Own Post, Edit State, Create Private and Create Attachment privileges.

3. Categories

Note

Some of the features described may only apply to the for-a-fee Akeeba Ticket System Professional edition.

This is accessible through the back-end Components, Akeeba Ticket System, Categories menu item.

Categories

<input type="checkbox"/>	▲	Status ↕	Title ↕	✓	✕	🗑️	🗑️	Access ↕	Language ↕	ID ↕
<input type="checkbox"/>	⋮	✔️	Pre-sales requests Alias: presales	82	0	0	0	Public	🇬🇧 English	13
<input type="checkbox"/>	⋮	✔️	Γενικές ερωτήσεις Alias: γενικές-ερωτήσεις	97	0	0	0	Public	🇬🇷 Greek	14
<input type="checkbox"/>	⋮	✔️	Data Compliance Alias: data-compliance	3228	0	0	0	Public	🇬🇧 English	15
<input type="checkbox"/>	⋮	✔️	Contact Us Alias: contact-us	2706	0	0	0	Public	🇬🇧 English	16
<input type="checkbox"/>	⋮	✔️	Foo Alias: foo	0	0	0	0	Public	🇬🇧 English	22
<input type="checkbox"/>	⋮	✔️	- Bar - Alias: bar	0	0	0	0	Public	🇬🇧 English	23
<input type="checkbox"/>	⋮	✔️	:- Baz :- Alias: baz	0	0	0	0	Public	🇬🇧 English	24

The basic organisational element of your ticket system is the Categories. Tickets are created inside a category and the category's Permissions define who can do what in the category and its sub-categories.

Please keep in mind that Akeeba Ticket System categories are managed by `com_categories`, the Joomla core component for managing categories in all core Joomla and third party components. We have very limited control over what is being displayed — namely, the Ticket System tab — and no control over how this core component works.

Category Permissions

Category Options Ticket Options Fields Publishing **Permissions**

Permissions

▶ Expand for notes about setting the permissions.

Public	Action	Select New Setting	Calculated Setting
– Cookies Accepted	Support Staff	Inherited	Not Allowed (Inherited)
– Cookies Declined	Create	Inherited	Not Allowed (Inherited)
– Guest	Delete	Inherited	Not Allowed (Inherited)
– Manager	Edit	Inherited	Not Allowed (Inherited)
⋮ – Administrator	Edit Own Ticket	Inherited	Not Allowed (Inherited)
– Registered	Edit Own Post	Inherited	Not Allowed (Inherited)
⋮ – Author	Edit State	Inherited	Not Allowed (Inherited)
⋮ – Editor	Create Private	Inherited	Not Allowed (Inherited)
⋮ – Publisher	Create Attachment	Inherited	Not Allowed (Inherited)
– Subscriber			
⋮ – Contact Us Subscriber			
⋮ – Data Compliance Subscriber			
– Super Users			

Each ATS category, just like any other Joomla! category, allows you to set up Permissions per user group. These permissions work in the same way discussed in the Options page's Permissions documentation section.

Keep in mind that Permissions in Joomla are inherited across two directions. First, there's the inheritance through the user group structure. Then there is the inheritance of permissions through the category structure. Also keep in mind that when there's an explicit Deny setting across the inheritance path the permission will always be denied, even if there's also an explicit Allow — the rule is that Deny wins over Allow.

Important

We have assembled a collection of links explaining how ACLs work in the Overview section of our documentation. Please read them before trying to set up your ticket system. Joomla! ACLs are extremely powerful and very easy to misuse. Even experienced users like our developers resort to planning groups and permissions using pen and paper before working on a site. As Joomla! co-founder Brian Teeman puts it, Joomla! begins with a P! [<http://brian.teeman.net/joomla/786-joomla-begins-with-a-p>].

The Ticket Options tab

Each ATS category has the standard Joomla category options but also some options specific to Akeeba Ticket System. The latter can be found in the Ticket Options tab.

Ticket Options

Category Options **Ticket Options** Fields Publishing Permissions

Ticket Options

Category email
 This is used with the Reply By Email and New Ticket By Email features of Akeeba Ticket System. You must either leave it blank (if you are not using these features) or specify a different email per category. Something like support-myCategoryAlias@example.com works best. Make sure all these email addresses end up being delivered to the same mailbox (e.g. set them up as aliases of a single email address)

Auto-select Private * Yes
 When enabled and the user is allowed to post private tickets, the new ticket form will default to creating a private ticket. Otherwise the default ticket type will always be Public.

Allowed visibility ▼
 Should this category allow **only** private or **only** public tickets?

Default priority ▼
 The pre-selected priority for new tickets in this category. Only applies if you enabled ticket priorities in Akeeba Ticket System's Options page.

Notify managers ▼
 Select which managers should be notified for new items in this category

Exclude managers ▼
 Select which managers shouldn't be notified for new items in this category

Default new post text
 Edit Insert View Format Table Tools
B I U Paragraph ▼ Paragraph ▼ sans-serif ▼ 16px ▼ ...
 p 0 WORDS
 The default text to show in the post area when a new ticket is being filed
Toggle Editor

Show custom fields * Never Category View Tickets View Both Views
 When should I display category custom fields? Category view is when displaying a list of categories — including the list of subcategories in a tickets view. Tickets view is when displaying the tickets submitted in this category.

InstantSearch method ▼
 When a new ticket is being created you can display search results from your site below the ticket title. The ticket title is used as the search query. There are two methods, Smart Search (built-in Joomla search) and URL (third party search engine such as DuckDuckGo, Google etc). Choosing None disables this feature.

Category email This is used with the Reply By Email and New Ticket By Email features of Akeeba Ticket System. You must either leave it blank (if you are not using these features) or specify a different email per category. Something like support-myCategoryAlias@example.com works best. Make sure all these email addresses end up being delivered to the same mailbox (e.g. set them up as aliases of a single email address).

This option only applies to Akeeba Ticket System Professional; the core version cannot receive new tickets and ticket replies by e-mail.

Important

There are two ways to set up the Reply by Email / New Ticket by Email features of ATS:

1. All email replies are delivered in the same email address. All new tickets are created in the Default Category specified in ATS' Options page. In this case you must leave the Category Email option blank and your users will be replying to the Email From address set up in your site's Global Configuration.
2. Email replies are sent to a different email address per category. Replies sent to the wrong email address will be ignored. New tickets will be created in a category based on the email address they were sent to. If you have multiple categories sharing the same email address, new tickets will be created in the first category with that Category Email address.

Auto-select Private	When enabled and the user is given the Private permission (they can post private tickets) the pre-selected visibility in the New Ticket form will be Private. In any other case the default value of the visibility in the New Ticket form will be Public.
Allowed Visibility	<p>Sometimes you want to only allow either Public or Private tickets in a category, not both. This option helps with that.</p> <p>Leaving this setting to the default value (<code>- Select -</code>) both Public and Private tickets can be created in the category. Otherwise only public or only private tickets will be possible in this category, per your setting here.</p> <p>If you set this option to Public the Auto-select Private option above will be ignored.</p>
Notify support staff	<p>If left empty — or the <code>All Support Staff</code> option is selected — all users who are allowed the Support Staff permission in this category will receive email notifications about new tickets and replies submitted in this category. If you'd rather only notify specific members of the support staff please select them from the list.</p> <p>If you are creating a new category you will not see any options other than the default. That's because the permissions for the category are not populated until you save it for the first time. Click on Save in the toolbar and you will see that the options (support staff users) are now available for selection.</p> <p>This option only applies to Akeeba Ticket System Professional.</p>
Notify support staff	<p>This is the opposite to the previous option. If you want all members of the support staff except a few select exceptions to be notified by email you can define here which users will NOT receive e-mail notifications.</p> <p>The same notice about new categories as above applies.</p> <p>This option only applies to Akeeba Ticket System Professional.</p>
Default new post text	The default text to show in the New Ticket area. This is useful for creating a ticket template that applies across all tickets in the category. You may want to instead use Ticket custom fields to create a more accurate and user friendly experience.
Show custom fields	<p>Select when the category custom fields should be displayed.</p> <ul style="list-style-type: none">• Never. The category custom fields will not be displayed by default. The <code>jcfields</code> property will still be populated so that your template overrides can use the category custom fields any way you deem fit. In short, this is an option for advanced site integrators.

- **Category view.** Category custom fields are shown when ATS is displaying the list of top-level categories. They will NOT be shown when displaying the tickets (and subcategories) or a category after clicking a link in the list of top-level categories.
- **Tickets view.** The opposite of the above.
- **Both views.** Show the category custom fields in both Category and Tickets views.

InstantSearch method

By default, every category will use the InstantSearch options defined in the component's Options page. You can, however, have each category have its own InstantSearch settings to further customise the behavior of the New Ticket page for each category.

Not sure what InstantSearch is? Read its documentation.

4. Tickets

This is accessible through the back-end Components, Akeeba Ticket System, Tickets menu item.

Tickets

The screenshot shows the Akeeba Ticket System interface. At the top, there are buttons for '+ New', 'Actions', and 'Delete'. On the right, there are 'Options' and 'Control Panel' buttons. Below these is a search bar with a magnifying glass icon, a 'Filter Options' dropdown, and a 'Clear' button. To the right of the search bar are dropdowns for 'Last replied, descending' and '20'. Below the search bar are several filter dropdowns: 'User', '- Assigned to -', '- Select Category -', '- State -', '- Set Visibility -', '- Select Status -', 'Created after' (with a calendar icon), 'Created before' (with a calendar icon), '- Select Language -', and '- Select Tag -'. The main content is a table of tickets with columns: 'User', 'Created', 'Title', 'Status', 'Public', 'Language', and 'ID'. There are three tickets listed:

<input type="checkbox"/>	User	Created	Title	Status	Public	Language	ID
<input type="checkbox"/>	Ivana Bubeniková ibubenikova5836 ibubenikova5836@gmail.com	2021-09-13 06:15 EEST Last Reply 2021-11-05 13:15 EET, ibubenikova5836	Open Reprehenderit laudantium aperiam eligendi autem natus et temporibus omnis nulla sed iusto excepturi iusto officiis corrupti. Category: Pre-sales requests	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	English	6071
<input type="checkbox"/>	简小红 jxhong6679 jxhong6679@hotmail.com	2021-09-14 02:08 EEST Last Reply 2021-11-05 11:12 EET, jxhong6679	Open Unde et inventore adipisci eligendi pariatur porro vero quod ut ad ut veritatis quis debitis tempora. Category: Data Compliance	<input checked="" type="checkbox"/>	<input type="checkbox"/>	English	6086
<input type="checkbox"/>	Grazia Negri gnegri7136 gnegri7136@bellini.com	2021-09-13 23:29 EEST Last Reply 2021-11-01 11:59 EET, gnegri7136	Closed Et nihil enim qui omnis blanditiis quasi aut. Category: Data Compliance	<input checked="" type="checkbox"/>	<input type="checkbox"/>	English	6096

This is the most important page in Akeeba Ticket System. This is where you can review, manage and reply to your tickets from your site's backend.

Please note that you can still manage tickets from the site's frontend. The backend offers more filtering and convenience options than the frontend.

The fields you see on each ticket are:

User	Who submitted the ticket. You can see their name, username, email address and user profile image (avatar). The avatar has a fallback to Gravatar which is why you may see some profile pictures even though you don't have a profile image field in the user profile.
Created	<p>The date when the ticket was created. The date and time is shown in the timezone you have selected in your backup profile or, if non was selected there, in the Server Timezone defined in your site's Global Configuration. The abbreviation of the timezone is displayed at the end. In the screenshot above, EEST stands for Eastern Europe Standard Time, the timezone of the Republic of Cyprus where our company is located in.</p> <p>Below that you will see the Last Reply line which displays the date, time and username who last replied to the ticket.</p>
Title	<p>The title of the ticket.</p> <p>Before the ticket title you will see a sort label with the state of the ticket: Open, Pending, Closed or any of the custom statuses. Open tickets are those last replied to by the client, awaiting for a Support Staff reply. Pending tickets are those last replied to by a member of the support staff, awaiting a reply from the client.</p> <p>Below the title there's the Category line displaying the category the ticket belongs to. The name of the category is a link which open the category for editing.</p> <p>If a ticket is assigned to a member of the support staff you will see the Assigned To line displaying the full name and username of the user the ticket is assigned to. The ticket assignment is for your internal use; the clients do not see it. It allows your support staff to know who's working on which ticket.</p>
Status	Whether the ticket is published or not. Unpublished tickets will not appear in the frontend of the site for anyone but support staff. Clicking on the icon allows you to toggle between the two ticket states.
Public	Whether the ticket is public or not. Clicking on the icon allows you to toggle between private and public.
Language	This column only appears if your site is using Joomla's "Multi-language" feature. It shows you the language of the category the ticket is created in.
ID	The numeric ticket ID. This is the unique identifier of every ticket and is always shown in the component's front-end.

The toolbar contains the icons which allow you to manage your tickets (some icons are under the Actions drop-down menu):

New	You can create a new ticket — typically on behalf of a user — by clicking on this button.
Publish	Set the selected tickets' Status to Published. They will be visible in the component's front-end.
Unpublish	Set the selected tickets' Status to Unpublished. They will no longer be visible in the component's front-end.
Make Private	Makes the selected tickets private.

Make Public	Makes the selected tickets public.
Batch	<p>Opens an interface which allows you to make changes across all selected tickets. The following changes are available:</p> <ul style="list-style-type: none">• Assign to. Assign the selected tickets to a specific member of the support staff. Please note that ONLY the users who are granted the Support Staff at the component level (the component's Options page) will appear here. If you want to assign a ticket to users who only have the support staff permission for specific categories you will have to do it by editing each ticket individually.• State. Set the ticket state to Open, Pending, Closed or any of the custom ticket states you have created in the component's Options page.• Move to category. Move the selected tickets to a different category. Please note that this may cause custom fields values to be lost if the exact same custom field (as identified by the custom field ID, NOT its name or label!) is not assigned to the new category you are moving the ticket to.• Add tag. Add a tag to the selected tickets. Use ticket tags to organise tickets, e.g to keep track of tickets concerning a common issue. You can filter the tickets list by tag.
Delete	Completely remove the selected tickets from the database.

Warning

Use with extreme caution! Deleted tickets cannot be restored. There is no "Trash can" feature in Akeeba Ticket System. We strongly recommend unpublishing the ticket instead.

Filters

The filters of the tickets page are mostly self explanatory. There are, however, hidden powers in some of the filters by adding a prefix before your search query e.g. `prefix:my search query`. Only one prefix can be used at a time.

The Search field supports the following prefixes:

- `id`: Search a ticket by its numeric ID.
- `category`: Show all tickets in categories where the category title matches the search query.
- `title`: Show all tickets whose title matches the search query.
- No prefix has the same effect as using the `title`: prefix.

The User field supports the following prefixes:

- `id`: Search a user by their numeric ID.
- `username`: Search for users whose username matches the search query.
- `name`: Search for users whose full name matches the search query.
- `email`: Search for users whose email matches the search query.
- No prefix searches for users whose username, name or email matches the search query.

4.1. Editing a ticket

Click on a ticket's title to view / edit the ticket.

Edit a ticket

The screenshot shows the 'Edit a ticket' interface with the following sections:

- Ticket Information** (selected tab):
 - Basic Information**: ID (6086), Title (*), Alias (*), Category (*), Tags, State (*), Public (*), Status.
 - Ticket Details**: Assigned to, Origin (*), Time, Created, User, Last Reply, Last Replying User.
 - Hosting Environment**: Affected site URL.
 - Site Connection Information**: Ask For Additional Information (No/Yes).

The Ticket Information tab has the basic information about the ticket, as well as any ticket custom fields you may have created. The Basic Information and Ticket Details areas are handled by Akeeba Ticket System itself and shown for all tickets. The areas with the dark headers are custom field groups. If there are fields outside a field group they appear in an area with the generic header title Fields.

The Basic Information is:

- ID** The numeric ID of the ticket. You cannot change that, it's created automatically when submitting the ticket.
- Title** The ticket's title, as displayed to the client and the support staff.
- Alias** The ticket's alias is used to construct the URL to the ticket. By default, it's created automatically from the ticket title. If there is a collision with an existing ticket ATS will append a dash and a number.
- Category** Which category the ticket belongs to. Please keep in mind that changing the category may lose custom fields values if the new category does not support the same custom fields (as identified by their numeric ID, NOT their name or label). It may also make it impossible for the assigned user to reply to the ticket if they do not have the Support Staff permission enabled for the new category.
- Tags** Add or remove ticket tags.
- State** Choose the state of the ticket: Open (awaiting for support staff reply), Pending (awaiting for client reply), Closed (the client cannot reply) or any of the custom ticket states you have created in the component's Options.

The Component

Public Is this ticket public or private? Private tickets are only visible to the support staff and the client who created them.

Status Is this ticket published or unpublished? Unpublished tickets are only visible to support staff.

The Ticket Details are:

Assigned To Choose a user this ticket will be assigned to. The list of users shown here are those who have the Support Staff privilege either globally or for the specific category the ticket belongs in.

Origin Choose how the ticket was filed: web or email.

This is normally handled automatically. You may, however, want to manually use Email if you received a ticket through non-automated means (email to your personal address, instant message, etc) and want to indicate this ticket was not filed directly from the user.

Time The time spent on the ticket in decimal minutes.

Created The date and time the ticket was created.

User The user who owns the ticket.

This is usually the same as the user who submitted the ticket. The only time these differ is when a member of the support staff files a ticket on behalf of the user.

Last Reply The date and time the ticket was last replied to. You cannot change this field.

Last Replying user The user who last replied to the ticket. You cannot change this field.

The Conversation tab at the top of the page shows you the posts of the ticket.

Right below you will find two tabs.

Edit a ticket - Conversation tab

Ticket information
Conversation
Manager Notes

🌐 Wednesday, 15 September 2021 02:08 EEST

Itaque dolor mollitia placeat culpa sunt. Et atque ipsa perspiciatis est velit consequatur. Vero quia occaecati perspiciatis est consequatur ipsam ea molestias.

Sunt quo quia quae consequatur optio perspiciatis. Ut aut provident ex cumque vel impedit. Minus unde quos fugit quasi.

Voluptatum iure est aspernatur sapiente eaque. Quae omnis totam eaque aut sunt. Error aut et aut ipsam ut. Fugiat neque tempore vitae.

Voluptas enim dolorem omnis est odit fuga sed.

Edited by **nicholas** on 2021-11-02 15:07 EET

✎ Edit
🔒 Unpublish
🗑 Delete

简小红
jxhong6679
jxhong6679@hotmail.com

💬 4
🕒 0'

🌐 Tuesday, 14 September 2021 21:06 EEST

Consequuntur libero aspernatur libero repudiandae error. Neque quia accusamus temporibus perferendis. Fugiat in vel esse ut aut. Alias delentit et aut quibusdam natus dolores.

Nulla et ad non quos autem in eaque. Ea ut perferendis non architecto ab fugiat. Ab vel explicabo optio corporis modi sed. Quod dolores qui est excepturi harum.

Odio omnis eos omnis neque voluptas natus quo. Voluptates soluta quaerat pariatum exercitationem et qui. Rem saepe quo quibusdam molestiae sint dolorem omnis.

✎ Edit
🔒 Unpublish
🗑 Delete

Charlie Admin
charlie
charlie@local.web

💬 0
🕒 0'

Send a reply

👁 **This is a public ticket**
Everybody will be able to see its contents. Do not include usernames, passwords or any other sensitive information.

Edit Insert View Format Table Tools

✖ CMS Content B I U ↺
☰ ☰ ☰ ☰
Paragraph Paragraph sans-serif 16px
🔍 ⋮

P

0 WORDS

👁 Toggle Editor

Time minutes Start Timer

User tags ▼

Attachments

Attachment * +

💬 Send your reply

The Conversation tab contains the ticket thread which both you (the support staff) and the user see. Each post has some or all of the following buttons:

- Publish / Unpublish**

Change the status of each post to published or unpublished. Unpublished posts are not seen in the front-end unless you are a member of the support staff. Members of the support staff and back-end users will see unpublished tickets with a big notice that the post is unpublished.

Edit Opens a post editor page where you can edit the text contained in the post. Very useful when a user accidentally publishes a password in a public ticket.

Delete Completely deletes the post.

Warning

Deleting a post is irreversible. Use with extreme caution. You should unpublish the post instead.

If you also have an attachment to your ticket you will see the following buttons next to it:

Publish / Unpublish Changes the published state of the attachment. Unpublished attachments are only visible to the support staff.

Delete Completely deletes an attachment (both the database record and the file on the disk)

Warning

Deleting an attachment is irreversible. Use with extreme caution. You should unpublish the attachment instead.

Below the conversation you will find the Post a reply area where you can enter a reply to the user and an optional attachment. Use the Post your reply button below it to submit your reply (and send an email to the user, if the respective plugin is enabled).

Edit a ticket - Manager Notes tab

Ticket information Conversation **Manager Notes**

Wednesday, 13 October 2021 14:50 EEST

This is a manager's note.

Hello there!

Edited by nicholas on 2021-11-15 11:04 EET

Edit Delete

Nicholas K. Dionysopoulos
nicholas

Save a manager's note

Edit Insert View Format Table Tools

CMS Content B I U S Paragraph Paragraph sans-serif 16px Q

P 0 WORDS

Toggle Editor

Save the manager's note

The other tab is called Manager Notes. Manager notes are internal communication between the support staff and can also double as private notes. Only support staff can see them. The users cannot see the manager notes.

They work just like the Conversation view with a few differences:

- You can not post attachments to Manager Notes.
- Manager Notes are only seen by the support staff, not the users.
- Other support staff does not receive an email when you post a new Manager Note.

5. Canned Replies

Note

This feature is NOT available on Akeeba Ticket System Core.

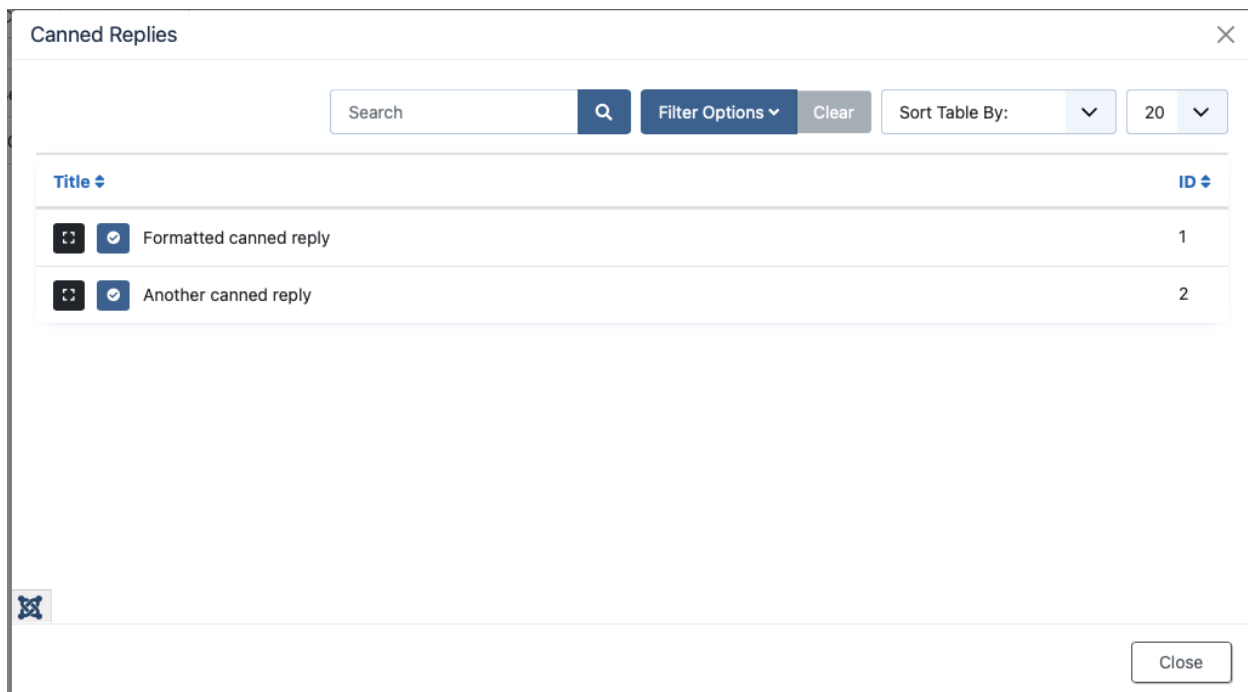
This is accessible through the back-end Components, Akeeba Ticket System, Control Panel menu item and then clicking on the Canned Replies button.

When answering to tickets you will often need to repeat the same reply across different clients. In other cases you may want the contact with the client to follow a formal, prescribed format in most cases. In short, you need some pre-composed replies available to your support staff. This is what the Canned Replied feature in Akeeba Ticket System does for you.

Whenever a member of the support staff is replying to a ticket they will see an editor button called Canned Replies. Clicking on that opens a modal dialog where they can select one of the canned replies you have already defined in the backend of the component.

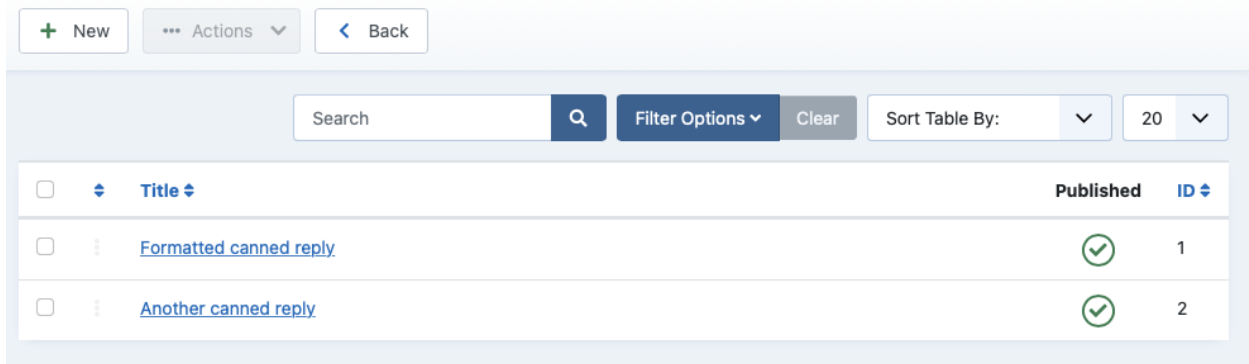
Please note that each editor may display Joomla editor buttons a bit differently — or not at all. The TinyMCE editor which ships with Joomla puts all editor buttons under the drop-down titled CMS Content. Moreover, you must make sure that you have published the Button – Akeeba Ticket System Canned Replies plugin in the `editors-xtd` group and set its Access to Public. Even though the access is Public, the button only appears in Akeeba Ticket System pages and only when the editor is displayed by a user who has the Support Staff permission allowed for the category the ticket belongs to.

The canned replied modal dialog



Each canned reply has two buttons next to its title. The first button that looks like the four corners of a rectangle expands or collapses a preview of the canned reply. This lets you preview a canned reply if you are not sure it's the one you really want. The second button that looks like a checkmark in a circle accept the canned reply and inserts it into the editor.

The back-end Canned Replies page



The backend Canned Replies page is where you will define your canned replies. The canned replies which are published are going to be shown in the modal dialog box.

The order they will be displayed in is the one they are listed on this page when the Sort Table By ordering selector drop-down has the option Ordering Ascending selected. When you select that list ordering option you can reorder the canned replies by dragging the handle (three dots) to the left of a canned reply. You can also edit the ordering as a number by clicking on the title to edit the canned reply.

Speaking of the title, this is something only shown to members of the support staff. It's there to help the support staff locate the canned reply they are looking for. Remember that the modal dialog allows support staff to search for a partial title through its search field.

6. Email Templates

Note

This feature is NOT available on Akeeba Ticket System Core. The free of charge version will not send emails regardless of your options.

Joomla introduced its own Mail Templates feature in Joomla 4.0 released in August 2021. Akeeba Ticket System version 5.0 and later use this core Joomla feature to manage the templates of the e-mails it sends.

You can access Joomla's email templates manager through the System menu item, Templates tab, Mail Templates link. From the Filter Options of that page click on the - Select Extension - drop-down and select Akeeba Ticket System.

In case you need to reset the email templates to their default values, or find that some or all of the Akeeba Ticket System email templates are missing we have included a utility page in Akeeba Ticket System itself. Go to Components, Akeeba Ticket System, Control Panel and click on the Email Templates button.

Email Templates

< Back

Where are the Email Templates?

Email templates for core and third party extensions, like Akeeba Ticket System, are centrally managed by Joomla itself in the System, Mail Templates page. Third party extensions, however, need to implement their own code to install and update the email templates. This is what this page does. If you want to customise the email templates please click the button below to go to Joomla's Mail Templates page.

✉ Mail Templates

Email Template management

✓
Install or Update

Installs new email templates for Akeeba Ticket System and updates existing ones if they've changed. The user-modified email templates for Akeeba Ticket System are left intact.

↺
Reset

Resets the email templates. This removes all of the email templates for Akeeba Ticket System, **including any user-modified email templates**. Then it installs all of the Akeeba Ticket System email templates afresh. Only use this button if you think you've damaged something in your database. Otherwise use the Joomla Email Templates manager to reset each email's subject and body to the default using the buttons provided in Joomla's interface.

The Mail Templates button takes you to Joomla's Mail Templates page and pre-filters it to show only Akeeba Ticket System email templates.

The Install or Update button will only install email templates which are missing. If there are any *unmodified* email templates which have changed since the previous version of ATS that installed them they will be updated too. If you have modified any email template it will NOT be updated.

The Reset button is the nuclear option. It will first remove all Mail Templates assigned to Akeeba Ticket System. It will then install them afresh. Use this if you think you have messed up so badly you don't know what else to do or you believe the database data for the mail templates is messed up beyond any reasonable attempt at recovery. Any customisations you have made will be lost.

6.1. Allowed variables in e-mails

Email templates wouldn't be worth anything if you couldn't include useful information about the ticket, the post and the user in them. You can do so by using "variables". The variables are ALWAYS written in all-uppercase letters, surrounded by curly brackets, without spaces, exactly as defined below.

Not all variables are supported by all email templates. When editing an email template you can see which variables are supported under the Insert Tags sidebar.

The available variables are:

- {SITENAME} The name of your site, as configured in Joomla's Global Configuration page.
- {SITEURL} The URL to your site, as detected the last time you visited the Akeeba Ticket System Control Panel page.

{NAME}	The full name of the person being contacted. For the mail templates which have to do with the mail gateway (receiving emails) this may be the name of the sender as communicated in the email received if a Joomla user cannot be found by the email address they sent an email from.
{URL}	The URL of the ticket on your site
{USER_NAME}	The name of the email recipient.
{ID}	The numeric ticket ID
{TITLE}	The ticket's title
{POSTER_NAME}	The full name of the person who posted the ticket / reply
{POSTER_USERNAME}	The username of the person who posted the ticket / reply
{CATNAME}	The title of the category the ticket belongs in
{TEXT}	The post text
{ATTACHMENT}	A link to the attachment(s) — including the filename and the URL. This does not output anything if there is no attachment

7. Auto-replies

Note

This feature is NOT available on Akeeba Ticket System Core.

You will often want to send an automatic reply to your users without human interaction. Some practical examples (non-inclusive list!):

- Thanking a user for filing a ticket and letting them know that a human agent will reply to them shortly, or inform them about any holidays / vacation time which may adversely effect the response time.
- Send a notification to a user after a certain amount of time since your support staff last replied to them (an automatic follow up post)
- Remind the user to attach a file if they forgot to do so; practical cases are when log files / screenshots are needed for software support or if you are running something like a digital print agency which requires the user to upload their artwork.
- Send an automatic reply to most frequently asked questions, based on keywords in the title or body of the ticket.
- Send an warning to users who have sent too many posts in a ticket, informing them that every time they post on their ticket they end up at the bottom of the reply queue.

This feature consists of three parts:

1. The automatic reply rules. This is what you define in this page. It determines the conditions for sending an automatic reply and the content of the reply.
2. The automatic reply bot usernames. This is defined in the Akeeba Ticket System - Automatic Replies plugin's Options.
3. A CLI script which must be manually scheduled by you, using CRON, to run the rules and send the replies. Please refer to the CRON scripts documentation for more information.

Warning

If you do not create a CRON script for this feature no automatic reply will ever be sent!

Each auto-reply rule has the following settings:

Title	The title of this auto reply rule. This is only visible to you.
Categories	Which ticket categories this rule applies to. Select the first option ("- Select a category -") to apply the rule to all categories.
Run based on manager reply	Apply this auto-reply rule depending on whether a human member of the support staff has already posted a reply to this ticket: <ul style="list-style-type: none">• If manager has not replied. Only send the auto-reply if a human member of the support staff has not posted yet to this ticket.• If manager has replied. Only send the auto-reply if a human member of the support staff has already posted to this ticket.• Always. It doesn't matter if a human member of the support staff has already posted to this ticket or not, send the auto-reply anyway.
Number of posts	The minimum number of total posts in the ticket before this rule is triggered. Leave empty or set to 0 to send the auto-reply irrespective of the number of total posts on the ticket.
Minimum time (minutes)	How many minutes must elapse after the ticket was first filed before sending this auto-reply. Set to 0 to ignore this option.
Run based on attachments	Apply this auto-reply rule depending on the existence of an attachment in any of the posts in this ticket: <ul style="list-style-type: none">• No attachments. Send this auto-reply only if there are no attachments in any of the posts in this ticket.• One or more attachments. Send this auto-reply only if there are attachments in any of the posts in this ticket.• Always. Send this auto-reply without caring about the existence of attachments.
Status	Set to Unpublished to disable the rule without deleting it.
Keywords in ticket title	Enter a list of keywords or phrases which must be present in the ticket title for this auto-reply to be sent. Type one keyword or phrase per line.
Keywords in ticket text	Enter a list of keywords or phrases which must be present in the ticket body (the posts of the user) for this auto-reply to be sent. Type one keyword or phrase per line.

Please note that there are some important considerations regarding the operation of this feature that you should be aware of:

- Only Open tickets are taken into account. If the ticket is in any other status it will be ignored.
- Only one auto-reply will ever be sent per ticket.
- The ordering of the rules matters. The first rule to match the ticket will send an auto-reply. All other rules will be ignored for that ticket.

- If a user is a member of the support staff in Category A and posts to Category B where he is not a member of the support staff the "Manager reply" option will consider him as a manager anyway. This means that auto-replies will never be sent to tickets owned by managers of ANY category on your site if the "Manager reply" option is set to "If not replied". Granted, this is a very rare case, but you should be aware that this is by design and not a bug.

8. The menu item types

Akeeba Ticket System only defines a handful of menu item types you can use to create new menu items in your Joomla site:

Assigned Tickets Displays the tickets that are assigned to the currently logged in support staff member.

Warning

This is supposed to be used on menu items which are only visible by registered, logged in users who are members of the support staff (they are set up as managers to the component or as support staff to one or more categories). Do not assign this to the Public or Guest viewing access level or to a viewing access level accessible by regular users. If you do your visitors will get a 403 Forbidden error message. This is by design.

Categories This is the main view that you want to include in a menu item on your site. It displays all the available support ticket categories and allows the user to navigate into them, create new tickets and so on.

Important

You **MUST** always have a menu item with this menu type, even in a hidden menu (a menu which is not published) per language. If you don't the SEF URL router shipped with Akeeba Ticket System will not work correctly and you will end up with strange URLs and/or routing issues (displaying the wrong category or ticket).

Category This menu item type allows you to create a menu item which displays the tickets inside a specific category and allows users to post new tickets only to that category.

My Tickets This is a convenient way to allow your users to view a list of all the tickets they have filed on your site. It will display all tickets submitted by the currently logged in user.

Warning

This is supposed to be used on menu items which are only visible by registered, logged in users. Do not assign this to the Public or Guest viewing access level. If you do your visitors will get a 403 Forbidden error message. This is by design. A non-logged-in user can't be shown "his" or "her" tickets because they cannot own any tickets!

Latest Open Shows all the open tickets, private and public, across all categories sorted by last reply date descending (oldest on top). This is only available to the support staff, i.e. users with the Access Administration Interface privilege or the Support Staff privilege.

Chapter 5. The plugins

Note

Some of the plugins are not available in Akeeba Ticket System Core.

Some features of Akeeba Ticket System require an integration with Joomla. As is usual, this is done through plugins.

We also use plugins to implement utility tasks which will can be set to run on a schedule.

The plugins can be found in the Joomla! plugins manager (System, Manage area, Plugins link). The plugins are placed in different groups, depending on the kind of functionality they perform. This chapter is organised by plugin group.

1. The “ats” plugins

Unlike plug-ins in the other groups, the ATS plugins are only loaded by and used in Akeeba Ticket System to implement task automation features.

1.1. Automatically close old tickets

Note

This plugin is NOT available in Akeeba Ticket System Core.

Displayed in the Joomla! Plugin Manager as Akeeba Ticket System - Automatically close old tickets

This plugin allows you to automatically close tickets with a status other than Open or Closed which have not received a reply from either the client or the support staff for a specific amount of time. It makes the *autoclose* CRON command available.

The idea behind this plugin is that some clients will see the reply from the support staff which addresses their issue but will neither reply nor close their ticket. Over time your support ticket system will show too many tickets as pending a reply from the client which makes it really hard for you to understand how many support cases are really open, making it potentially difficult to schedule your support staff. This plugin automatically closes stale tickets which are most likely resolved or for which the client has otherwise lost interest.

The plugin can send an automatic post on the ticket when closing it depending on the options below.

Maximum inactive ticket age	Eligible tickets older than this many days (but newer than the "Silent close period") will be set to Closed status, posting a notification that the ticket is closed because it's inactive. If you don't want to use this feature set it to a really long time e.g. 10950 (that's 30 years in days).
-----------------------------	--

Silent close period	Eligible tickets older than this many days will be set to Closed status, <i>without</i> posting a notification. This must be a bigger number than the previous setting. Otherwise all tickets will be closed silently. If you don't want to use this feature set it to a really long time e.g. 10950 (that's 30 years in days).
---------------------	---

1.2. Automatic replies

Note

This plugin is NOT available in Akeeba Ticket System Core.

Displayed in the Joomla! Plugin Manager as Akeeba Ticket System - Automatic replies

This plugin allows you to send automatic replies to tickets, either instantly (when they are filed) or on a schedule. It makes the *autoreply* CRON command available when set up to do so.

You can set up the automatic replies and the rules under which they are sent in the Auto-replies page of the component.

Run on ticket creation	Should automatic replies be processed immediately after a new ticket is created? Be careful with this option. If an Auto-reply rule results in a reply being sent it will appear to be sent at the same time as the new ticket is submitted. This may confuse clients and they might even ignore the automatic reply, confusing it with the new ticket confirmation they received at the same time.
Make available as a CRON task	When enabled, it makes the <i>autoreply</i> CRON command available to the CLI CRON script and the CRON URL of ATS. This command processes all Auto-Reply rules for all currently Open tickets and sends the corresponding emails.
Bot usernames	When you are using the auto-reply feature in Akeeba Ticket System you need one or more Joomla users to appear as the senders of the reply text. This is where you specify their usernames, one per line.

Important

The users must already exist in your Joomla! site. If you have not created them already, go to Users, User Manager in the back-end of your site and create one or more users. These users DO NOT have to have Support Staff permissions in any ATS category. In fact, you never need to log in with these users or even have valid email addresses for them.

As an example, you can make a new user with the full name Automatic Reply, username *auto_reply_bot* and email address *bot@example.com*.

1.3. Delete Manager Notes on ticket close / unpublish

Note

This plugin is NOT available in Akeeba Ticket System Core.

Displayed in the Joomla! Plugin Manager as Akeeba Ticket System - Delete Manager Notes on ticket close / unpublish

This plugin allows you to delete the manager notes from tickets when they get closed or unpublished. It makes the *deletenotes* CRON command available when set up to do so.

This might be a requirement for EU GDPR compliance. If you collect personally identifiable or confidential information in Manager Notes -- such as social security numbers, login credentials etc -- you have to delete them when you no longer need them to fulfil the request of your client. Obviously, when a ticket is closed or unpublished we can reasonably assume that the request of the client has been fulfilled therefore there is no legal basis for keeping this information in the ticket system. This plugin can automatically delete that information either immediately at ticket close / unpublish or after the fact for all closed / unpublished tickets.

The options for this plugin are:

Run on ticket close / unpublish	Automatically deletes the Manager Notes as soon as a ticket's status is set to Closed or when the ticket itself is unpublished.
---------------------------------	---

Make available as a CRON task When enabled, it makes the `deletenotes` CRON command available to the CLI CRON script and the CRON URL of ATS. This command will delete the Manager Notes from all tickets which are already closed or unpublished, in batches of 100 tickets.

1.4. Fetch email

Note

This plugin is NOT available in Akeeba Ticket System Core.

Displayed in the Joomla! Plugin Manager as Akeeba Ticket System - Fetch Email

This plugin is responsible for *receiving* email and creating ticket replies or new tickets. This feature is documented in its own documentation section.

1.5. Remove Obsolete Attachments

Note

This plugin is NOT available in Akeeba Ticket System Core.

Displayed in the Joomla! Plugin Manager as Akeeba Ticket System - Remove Obsolete Attachments

This plugin allows you to automatically delete attachments a certain amount of time after they are sent. It makes the `removeattachments` CRON command available when set up to do so.

Only closed When enabled only attachments which are old enough and belong to already closed tickets will be removed. Otherwise all attachments, even from open tickets, will be removed if they are old enough.

Maximum attachment age (days) Attachments older than this many days will be deleted.

2. The “actionlog” plugins

2.1. Action Log plugin

Displayed in the Joomla! Plugin Manager as Action Log - Akeeba Ticket System Log User Actions.

This plugin integrates with Joomla's User Actions Log, recording significant events performed by users in the Akeeba Ticket System installation on your site.

3. The “console” plugins

3.1. Akeeba Ticket System

Note

This plugin is NOT available in Akeeba Ticket System Core.

Displayed in the Joomla! Plugin Manager as Console - Akeeba Ticket System.

This plugin implements the Akeeba Ticket System – Joomla CLI Application integration. This is required for the `cli/joomla.php` script to process all the commands whose name starts with `ats:` described in the Automation and CRON tasks documentation pages. If this plugin is unpublished CLI CRON jobs for Akeeba Ticket System tasks will fail.

4. The “content” plugins

4.1. Akeeba Ticket System Custom Fields Show–on Behavior

Displayed in the Joomla! Plugin Manager as Content - Akeeba Ticket System Custom Fields Show–on Behavior.

This plugin allows you to have conditional custom fields in tickets. If this plugin is unpublished conditional fields will stop working.

5. The “editors-xtd” plugins

5.1. Akeeba Ticket System Canned Replies

Note

This plugin is NOT available in Akeeba Ticket System Core.

Displayed in the Joomla! Plugin Manager as Button - Akeeba Ticket System Canned Replies.

This plugin displays the Canned Replies button in the editor used for ticket replies. You must have it published at all times for support staff to be able to use canned replies.

6. The “finder” plugins

6.1. Akeeba Ticket System

Displayed in the Joomla! Plugin Manager as Smart Search - Akeeba Ticket System.

This plugin integrates Akeeba Ticket System with Joomla!'s Smart Search feature (a.k.a. "Finder"). In short, it allows Smart Search to index and search public tickets.

Please note that private tickets are never indexed. If you make a public ticket private it will be immediately removed from Smart Search's index.

Note

No, we cannot have private tickets indexed and their results shown only to the ticket owner. This is a limitation of Smart Search, not Akeeba Ticket System. In order to do that we would have to create a separate Joomla! user group and a viewing access level for each and every Joomla! user. That would render your site so slow as to be completely unusable.

7. The “user” plugins

7.1. Akeeba Ticket System

Note

This plugin is NOT available in Akeeba Ticket System Core.

Displayed in the Joomla! Plugin Manager as User - Akeeba Ticket System.

This plugin does two things, either of which can be disabled:

- Allows users to enter a *signature*. The signature is displayed with all their posts, at the bottom of each and every post.
- Allows users to enter additional email addresses for the ticket system. Any e-mails sent by ATS will be copied to these email addresses. Furthermore, tickets and replies can be sent from these email addresses. If an existing user of the site is using one of the provided e-mails then the user with the main email address will take precedence when creating a new ticket or sending a reply.

Note

All users of your site can set up their signature, even if they do not have access to the ticket system. Doing otherwise would require doing N+1 queries to the Joomla permissions tables, where N is the number of Akeeba Ticket System categories. Several of our users have dozens of categories. Therefore these queries would make the profile edit page very slow, frustrating users.

Chapter 6. The modules

Note

Some of the modules are not available in Akeeba Ticket System Core.

Modules can be used to display useful snippets of information in the front- or backend of the site. Akeeba Ticket System includes a number of site and administrator modules to help you display some of its information to your users without them having to visit the component's pages.

1. Administrator modules

Administrator modules can be used in Joomla's dashboard pages to create information-rich overviews for backend users.

1.1. Ticket Graphs

Displayed as: Akeeba Ticket System – Ticket Graphs

This module displays a graph with the number of tickets and posts submitted over the last 30 calendar days in Akeeba Ticket System. It is the same as the graph you see in Akeeba Ticket System's Control Panel page.

The graph data is retrieved asynchronously from your server via AJAX to prevent blocking the display of the dashboard it is placed in.

This module has no options besides the default Joomla module options.

1.2. Ticket Stats

Displayed as: Akeeba Ticket System – Ticket Stats

This module displays a compact table with statistics about tickets in the Open, Pending and Closed states and across Public tickets, Private Tickets and their aggregate.

Note

The Pending ticket type includes tickets in the Pending state *as well as* every custom state. This is intentional. With up to 100 custom ticket states the table could get unwieldy.

Clicking on the headers of the table takes you to the backend Tickets page of Akeeba Ticket System with the type of ticket already preselected in the filters *alongside any other filters you may have already applied on that page*.

This module has no options besides the default Joomla module options.

2. Site modules

Site modules can be used in the public frontend of your site to present information to your site's visitors.

2.1. Tickets info

Displayed as: Akeeba Ticket System - Tickets info

This module allows you to present a quick shortcut to the user's ticket by status: Open, Pending (including all custom states) and Closed — or only some or none of the above. You can further show quick buttons to the My Tickets page and the New Ticket page. Finally, you can add custom HTML to be presented at the top of the module, between the list of tickets and the buttons, and at the bottom of the module.

This module has the following options besides the default Joomla module options.

Text before	HTML to include at the top of the module.
Text before buttons	HTML to include between the list of number of tickets by status and the My Tickets / New Ticket buttons.
Text after	HTML to include at the bottom of the module.
Show open	Display the number of open tickets submitted by the current user.
Show pending	Display the number of pending tickets submitted by the current user. This includes all tickets with a custom status. This is intentional; with up to 100 custom statuses listing each one separately could result in an unwieldy module.
Show closed	Display the number of closed tickets submitted by the current user.
Show My Tickets	Display a link to the My Tickets page. Please make sure you have created a menu item to Akeeba Ticket System's 'My Tickets' view if the URL in the module seems to be broken or takes you to the wrong page.
Show New Ticket button	Display a New Ticket button. When this option is enabled the New Ticket button is always displayed, without checking if the user has permissions to submit tickets in any category. This is intentional. Checking the permissions on each and every ATS category is very slow. With just 10 categories we measured an average of 0.3 seconds page load performance impact, twice as much as a brand new Joomla installation took to load on the same server. If the button leads to the wrong page or shows a Not Found error please remember to create a menu item to Akeeba Ticket System's 'New Ticket' view before turning on this option.

Chapter 7. Automation and CRON scripts

Note

This feature is not available in Akeeba Ticket System Core.

Akeeba Ticket System can benefit from the periodic execution of optional, administrative tasks. If you want to allow your clients to create new tickets or reply to existing ones over email you also need to set up a scheduled task for that too.

There are two ways Akeeba Ticket System can help you automate tasks:

- **CRON jobs with the Joomla CLI application.** If your server gives you access to real CRON jobs, executing command line scripts periodically, this is the preferred method. Automated tasks execute outside of your web server and are not bound to the same time limitations as loading pages on your site. Moreover, since the web server and most of Joomla is not loaded they tend to be more memory efficient and faster than any other method.
- **URLs for CRON jobs.** There's a special, password-protected URL you can use to execute maintenance tasks automatically. Since they run outside the context of a visitor accessing your site they do not slow your visitors down. However, since they are still accessed through your site, they are bound by the same time limits as any other page load of your site. Furthermore, third party plugins loading on your site may cause unintended effects, increase the memory consumption and slow down the execution of ATS CRON tasks executed this way. This is only recommended when your host does not allow you to create real CRON jobs. You can use the URLs with pseudo-CRON (access a URL at specified intervals) either provided by your host or a third party service such as WebCRON.

In all of the above cases the CRON commands executing are the same. They are provided by plugins in the **ats** group. A cheatsheet with the available CRON commands and links to the respective plugins' documentation can be found at the end of this chapter.

Why not use the scheduled tasks feature in Joomla 4.1 and later?

This feature WILL be supported in a future release.

At the time we finished working on Akeeba Ticket System 5 for Joomla 4 in November 2021 this Joomla feature wasn't ready yet. In fact, it was still being worked on for inclusion in the development build for Joomla 4.1 which is to be released in February 2022.

Once Joomla 4.1 reaches a Release Candidate state we will work on integrating Akeeba Ticket System's scheduled tasks with Joomla's scheduled tasks feature. Considering that we also have our internal release cycle we expect this feature to be included in a public release within 1–2 months of Joomla 4.1's release. The benefit for us and you is that having a common scheduler makes things far more consistent and easier to manage.

1. CLI script for CRON jobs

Note

This feature is only available in Akeeba Ticket System Professional.

This is the recommended method for scheduling Akeeba Ticket System CRON tasks on your site.

The big picture

Setting up a CLI script for a CRON job is generally a fairly simple task. You need to know the path to your PHP CLI executable and the path to your site.

The command line script you need to execute to perform CRON tasks is stored under your site's `cli` directory and is called `joomla.php`. This is the Joomla CLI Application, an official script shipped with Joomla itself (it's already present on your site; we do not install it ourselves).

Therefore you need to enter a command line similar to the following to your server's CRON job manager:

```
/usr/local/bin/php /home/USER/webroot/cli/joomla.php ats:cron CRONCommandName
```

where */usr/local/bin/php* is the path to your PHP CLI executable and */home/USER/webroot* is the absolute path to your web site's root. You can get this information from your host.

The `CRONCommandName` is the name of the CRON command. Please consult the CRON Commands section of this documentation for a list of available CRON command names.

Here are some **very important notes**:

- You **MUST** publish the Console - Akeeba Ticket System plugin and set its Access to Public. This plugin lets the Joomla CLI Application know about the `ats:cron` command and how to run it.
- We cannot know and document the path to your PHP CLI executable. Please ask your host about it. Also note that sometimes hosts will give you the path to the PHP CGI binary. This will **NOT** work properly with the Joomla CLI Application script. CGI (Common Gateway Interface) and CLI (Command Line Interface) are a world apart. The former is designed to run scripts which will be served as web pages. The latter is designed to run command line scripts, usually for use with CRON.
- We cannot know the path to your site and document the exact path to Joomla's CLI Application script. However, you can always go to Akeeba Ticket System's control panel page and click the Automation tab. It will give you the exact path to the CLI script. It can do that because it runs on the same server which will be executing the CRON task, therefore it can figure out the correct path for you.

Setting up a CRON job in cPanel

Go to your cPanel main page and choose the CRON Jobs icon from the Advanced pane. In the Add New CRON Job box on the page which loads, enter the following information:

Common Settings Choose the frequency of your backup, for example once per day.

Command Enter your CRON command as explained above. For example:

```
/usr/local/bin/php /home/USER/webroot/cli/joomla.php ats:cron CRONCommandName
```

Finally, click the Add New Cron Job button to activate the CRON job.

2. URL for CRON jobs

This is an alternative method for scheduling Akeeba Ticket System CRON tasks on your site. Use it only if the CLI script method cannot be used on your site.

Initial configuration

Akeeba Ticket System protects the URL for CRON jobs with a password. This prevents unauthorized users from triggering CRON jobs on your site too frequently, something which could have unpredictable results or even bring your site down.

First go to Akeeba Ticket System in your site's administrator backend. Click on the Options button to view the Options page. Select the Automation tab.

Find the Secret Key option and enter a secret. Please only use lowercase letters a-z without accents or diacritics, numbers 0-9, dashes and underscores. Make this long, ideally over 24 characters. You can create a truly random secret key in the Random.org site [<https://www.random.org/passwords/?num=1&len=24&format=html&rnd=new>].

This will be your SecretKey noted below.

Running CRON tasks through the URL

You can execute the CRON tasks with the following URL:

```
https://www.example.com/index.php?option=com_ats&view=cron&key=SecretKey&command=CRONCommandName
```

where <https://www.example.com> is the URL to your site, SecretKey is your secret key as explained above and CRONCommandName is the name of the CRON command you want to execute. Please consult the CRON Commands section of this documentation for a list of available CRON command names.

Results

Akeeba Ticket System communicates the status of the CRON task execution through HTTP status code headers.

200 OK	The CRON task executed successfully
500 Internal Server Error	The CRON task failed to execute or Joomla crashed before the task executed
501	The command you specified was not found. Did you forget to publish the respective plugin?
503	You have not configured a Secret Key yet.
403	You have entered the wrong Secret Key. If your Secret Key contains any characters other than the ones we described above you need to URL encode it first.

3. CRON commands

CRON commands for Akeeba Ticket System are made available by plugins in the `ats` group. At the time of this writing the following CRON commands are available:

mailfetch	You need to publish the Akeeba Ticket System - Fetch Email plugin for this command to work. Downloads and processes email to create new tickets and ticket replies. This command replaces the obsolete <code>ats-mail-fetch.php</code> CLI script which was used in Akeeba Ticket System 3.1.0 and earlier.
autoclose	You need to publish the Akeeba Ticket System - Automatically close old tickets plugin for this command to work.

Automatically closes tickets open without activity for longer than a specified amount of time.

This command replaces the obsolete `ats-autoclose-tickets.php` CLI script which was used in Akeeba Ticket System 3.1.0 and earlier.

autoreply

You need to publish the Akeeba Ticket System - Automatic Replies plugin and enabled its `Make available` as a `CRON` task option for this command to work.

Automatically replies to tickets based on the Auto-Replies rules set up in the Akeeba Ticket System component.

This command replaces the obsolete `ats-auto-reply.php` CLI script which was used in Akeeba Ticket System 3.1.0 and earlier.

deletenotes

You need to publish the Akeeba Ticket System - Delete Manager Notes on ticket close / unpublish plugin and enabled its `Make available` as a `CRON` task option for this command to work.

Removes the Manager Notes from all closed and unpublished tickets. This may be required by GDPR and similar privacy protection laws if you are storing personally identifiable or sensitive information (e.g. service connection information, social security numbers, ...) in Manager Notes.

removeattachments

You need to publish the Akeeba Ticket System - Remove Obsolete Attachments plugin for this command to work.

Automatically deletes attachments a certain amount of time after they were first submitted.

This command replaces the obsolete `ats-remove-attachments.php` CLI script which was used in Akeeba Ticket System 3.1.0 and earlier.

Part II. Appendices

Table of Contents

A. GNU Free Documentation License 129

Appendix A. GNU Free Documentation License

Copyright (C) 2000, 2001, 2002 Free Software Foundation, Inc. 51 Franklin St , Fifth Floor, Boston, MA 02110-1301 USA . Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors

or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-

network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation

is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/> [<http://www.gnu.org/copyleft/>].

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (C) YEAR YOUR NAME.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software

Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.