



Guru Project Manager

Complete WordPress Plugin Documentation

Project management with project cards, Kanban workflow, milestones, role-aware access, payments, notifications, and frontend shortcodes.

Version 1.0

The Guru Way!

A Guru Plugins Product

<https://plugins.guru-is.com>

Contents

Section	Topic
1	Start Here
2	Requirements, Installation and Activation
3	Navigation and Dashboard
4	Roles, Permissions and Project Access
5	Projects
6	Project Overview and Team Management
7	Tasks
8	Task Notes
9	Kanban Board and Custom Stages
10	Milestones
11	Files
12	Discussions
13	Payments
14	Activity History
15	Notifications and Cron
16	Settings
17	Frontend Shortcodes
18	AJAX Architecture
19	Data Model, Migration and Uninstall
20	Troubleshooting
21	Release and Regression Checklist

22 Getting Started, Modal Compatibility and Accessibility

Scope

This guide is written against the Guru Project Manager v1.0 release-candidate codebase and is intended to be the authoritative operating manual for the plugin.

1. Start Here

Guru Project Manager provides project management inside WordPress for Team Guru and project participants. The plugin connects projects, people, tasks, milestones, Kanban stages, files, discussions, payments, notifications, and activity history in one permission-aware workspace.

- Create a project and assign project members.
- Add tasks, assign owners, connect tasks to milestones, and set due dates.
- Run daily work from the Kanban board or Tasks table.
- Use Milestones for delivery checkpoints and phase-level progress.
- Use Team, Files, Discussions, Payments, and Activity to keep project context in one place.
- Use the frontend shortcode when users should work outside wp-admin.

2. Requirements, Installation and Activation

Requirement	v1.0 requirement / behavior
WordPress	6.2 or newer.
PHP	7.4 or newer.
Email	WordPress must be able to send mail for notification delivery.
WP-Cron	Used for scheduled task and milestone due notifications.
Activation	Creates or upgrades the plugin tables, seeds the current WordPress Administrator as Guru Super Admin, ensures default Kanban stages, and schedules the daily notification event.
Deactivation	Unscheduled the daily notification event and preserves plugin data.

3. Navigation and Dashboard

Area	Who sees it / purpose
Projects	Primary workspace. Lists accessible projects and opens project workspaces.
Team	Guru Admins and Guru Super Admins. Global Guru role management and workload counts.
Payments	Guru Super Admins only. Global payment administration.
Activity	Guru Admins and Guru Super Admins. Recent activity across the plugin.
Settings	Guru Super Admins only. Notifications, due-soon window, and uninstall-data control.
Documentation	A small link below the main heading opens the bundled PDF in a new tab. A Documentation link is also available from the WordPress Plugins screen.

4. Roles, Permissions and Project Access

Guru roles are independent from native WordPress roles, with one important exception: WordPress Administrators are always treated as Guru Super Admins.

Guru role	Core behavior
Super Admin	Full plugin access, sensitive tasks, payments, settings, team management, global activity, and project management.
Admin	Can see all projects, manage projects and Team, and view global Activity. Does not receive Super Admin-only payment/settings/sensitive privileges.
Manager	Can create projects. Can manage a project when assigned to it, or when project rules otherwise allow manager-level access.
Team Member	Default Guru role for users without another Guru role. Can work assigned projects but does not receive manager-level editing by role alone.
Viewer / Client	Can view assigned projects. Cannot use work actions that require <code>can_work_project()</code> .

Permission rules that matter

- Admins and Super Admins can view all projects.
- Non-admin users must be assigned to a project to view it.
- A project Manager or a project member with a Manager Override can manage that assigned project.
- Task editing requires Guru Super Admin, Admin, or Manager role plus project view access.
- Task completion can also be performed by an assigned task user who can work the project.
- Only Guru Super Admins can view sensitive tasks.
- Only Guru Super Admins can manage payments and Settings.
- Only Guru Admins and Super Admins can manage the global Team screen.

Manager Overrides

A project member can receive manager authority for one project without changing that user's global Guru role. The plugin protects against removing the last manager-level user from a project.

5. Projects

Projects are the top-level work containers. Each project can connect members, tasks, stages, milestones, files, discussions, payments, and activity.

Project capability	Behavior
Create	Available to Guru Super Admins, Admins, and Managers.
Edit	Available to users who can manage the project.
Statuses	Active, On Hold, Completed.
Project cards	Show project summary information and provide the entry point to the project workspace.
Status views	Projects can be filtered by Active, On Hold, and Completed.
Sorting	The UI supports updated, newest, oldest, A-Z, Z-A, and due-date sorting.

Add / Edit Project modal

Field / control	Purpose
Project title	Primary project name.
Description	Project summary/details.
Status	Active, On Hold, or Completed.
Start date	Optional project start date.
Due date	Optional project due date.
Project Members	AJAX WordPress user lookup. Selected users are shown as removable chips.
Manager Overrides for This Project	AJAX WordPress user lookup for project-level manager authority. Selecting an override also ensures the user is a project member.

Project notifications and history

- Project creation is logged and can trigger the Project created notification.
- Project status changes are logged and can trigger Project status changed notifications.
- Project due-date changes can trigger Project due date changed notifications.
- Completing a project can trigger the Project completed notification.
- Member add/remove and manager override changes are logged.

6. Project Overview and Team Management

The Project Overview is the high-level project landing page. The Team section inside the overview is the primary place to manage membership after a project has been created.

Team action	Behavior
View members	Displays current project users with identity and project-manager state.
Add Member	AJAX WordPress user search adds a project member without a page reload.
Remove Member	AJAX removal. The plugin also removes that user's task assignments inside the project.
Manager Override	AJAX add/remove for project-level manager authority.
Last-manager protection	A manager-level project member cannot be removed if that would leave the project without another manager-level user.

Global vs project role

Global Project Manager roles carry into every assigned project. Manager Overrides apply only to the specific project where the override is enabled.

Global Team screen

Column	Meaning
User	WordPress display name, email, and avatar.
Guru Role	Global Guru role selector.
Projects	Number of project memberships.
Complete Projects	Number of assigned projects with completed status.
Manager Overrides	Count of project-specific manager overrides.
Open Tasks	Assigned tasks whose workflow status is not completed.

7. Tasks

Tasks are the primary units of work. They can be created from project task controls and viewed from both the Tasks table and Kanban board.

Task field	Behavior
Task Name	Required task title.
Description	Rich-text description using the native WordPress editor when editing.
Stage	Current project-specific Kanban stage.
Priority	Low, Normal, High, Urgent.
Milestone	Optional milestone relationship.
Due Date	Optional task due date.
Assignees	AJAX lookup restricted to current project members. Selected users appear as removable chips.
Sensitive	Super Admin-only sensitive-task checkbox. Sensitive tasks are only visible to Guru Super Admins.

Task View and Edit

- Task names open a View Task modal.
- View mode shows description, stage/status, priority, milestone, due date, assignees, notes, and action permissions.
- Authorized editors can switch into Edit Task.
- Existing descriptions remain in view mode until Edit Description is selected.
- Task assignee changes are synchronized on save and can trigger reassignment notifications.

Complete, Reopen and Delete

Task state	Actions
Open task	Complete is available to users allowed to complete the task. Delete is shown to project managers.
Completed task	Reopen replaces Complete. Delete is not available while completed.
Delete behavior	Requires project-management authority, removes task assignees, task notes, and task notification rows, then refreshes linked milestone progress.
Reopen destination	Reopen moves the task to the project's first open stage.
Complete destination	Complete moves the task to the project's configured completed stage.

8. Task Notes

Task Notes preserve discussion and decisions at the task level without requiring full task-edit permission.

- Project participants who can view the task may add notes, including Viewer / Client users when project access permits.
- Sensitive-task notes inherit sensitive-task visibility.
- Notes show the author's WordPress avatar, display name, timestamp, and content.
- Adding a note is AJAX-driven.
- The main Tasks table shows a live Notes count.
- Task-note activity is logged.
- Task note added notifications can be enabled in Settings.

When to use Notes

Use Task Notes for updates, decisions, or context that belongs to one task. Use project Discussions when the conversation affects the wider project.

9. Kanban Board and Custom Stages

Every project gets its own stage configuration. New projects begin with Backlog, To Do, In Progress, Review, and Completed.

Board capability	Behavior
Task drag and drop	Authorized users can move task cards between stages.
Completed stage	Exactly one stage is designated as the completed stage.
Completion semantics	The completed state is tied to the stage flag, not the literal stage name. Renaming Completed to Delivered still preserves completion behavior.
Stage IDs	Tasks are attached to stage IDs so renaming does not break task placement.

9.1 Manage Stages

Manage Stages is a staged editor. Nothing changes in the database until Save Stages succeeds.

Action in modal	Before Save Stages
Rename	Changes the draft stage name only.
Drag to reorder	Changes draft order only.
Add Stage	Adds a draft stage row only.
Remove	Marks an existing stage for removal. New unsaved stages can be removed from the draft.
Completed stage	Changes the draft completed-stage choice only.
Task destination	For a removed stage with tasks, choose which remaining stage receives those tasks.
Close behavior	Result
Save Stages	Sends the entire stage configuration in one AJAX request. The server validates it, applies inserts/renames/order/removals/completed-stage changes, then returns fresh board HTML.
Cancel / X / backdrop / Escape	If changes are dirty, the user is asked whether to discard unsaved changes. Discarding restores the saved configuration and writes nothing.
After Save	The main Kanban board is replaced immediately with freshly rendered server HTML. New, renamed, moved, and removed stages appear without a page reload.

Removal safety

At least one stage must remain. If a removed stage contains tasks, those tasks must have a valid destination stage. Exactly one active stage must be designated as completed.

Stage activity

- Stage added is recorded in Activity when applicable.
- Stage renamed is recorded in Activity when applicable.
- Stages reordered is recorded in Activity when applicable.
- Stage removed is recorded in Activity when applicable.
- Completed stage changed is recorded in Activity when applicable.
- Final stage configuration saved is recorded in Activity when applicable.

10. Milestones

Milestones are project checkpoints that can group tasks and report completion progress.

Milestone field / action	Behavior
Milestone Name	Required title.
Status	Active or Completed.
Due Date	Optional milestone due date.
Tasks	Dropdown contains only currently available tasks. A task already assigned to another milestone is excluded.
Selected task list	Choosing a task adds it beneath the dropdown with an X control.
New milestone	Selected tasks are staged in the modal until Save Milestone creates the milestone.
Existing milestone	Adding or removing task relationships in Edit Milestone happens immediately through AJAX.

View / Edit Milestone

- Milestone titles open a View Milestone modal.
- View mode shows status, due date, progress, and assigned tasks.
- Authorized project managers can switch to Edit Milestone.
- Editing supports title, status, due date, and task relationships.

Complete, Reopen and Delete

State	Behavior
Open	Complete and Delete are available to authorized project managers.
Completed	Reopen replaces Complete. A completed milestone must be reopened before deletion.
Delete	Detaches linked tasks by clearing their milestone relationship. Tasks themselves are never deleted.
Automatic progress	Milestone progress is recalculated from linked task completion state.
Notifications	Milestone completion, due soon, overdue, and date-change notifications are supported.

11. Files

- Users who can work the project can upload files.
- Uploads use WordPress Media Library handling.
- The plugin stores a project file record that points to the WordPress attachment.
- Uploaded files show uploader and upload date.
- File upload activity is logged.
- New file uploaded notifications can be sent to other project members.

Uninstall note

Even when destructive plugin-data uninstall is enabled, WordPress Media Library attachments are not deleted. Only the Guru Project Manager file-record table is removed.

12. Discussions

Discussions provide project-level conversation threads and replies.

Action	Behavior
New topic	Requires project work permission and a topic title plus message.
Reply	Requires project work permission and a valid parent topic.
Formatting	Discussion content is sanitized with WordPress allowed HTML.
Activity	New topics and replies are logged.
Notification	A new top-level discussion can notify project managers. Replies do not trigger the same new-discussion notification.

13. Payments

Payment administration is intentionally restricted to Guru Super Admins. Normal users can only view payment records tied to their own WordPress user ID.

Payment field	Behavior
Project	Required project.
Project User	Required WordPress user.
Milestone	Optional relationship, validated to the selected project.
Amount	Non-negative decimal amount.
Paid	Yes / No.
Paid Date	Set automatically when marked paid and cleared when changed back to unpaid.
Note	Optional payment note.

Payment operations

- Add Payment is AJAX-driven.
- Paid/unpaid toggles are AJAX-driven.
- Delete Payment is AJAX-driven and Super Admin-only.
- Payment activity is marked sensitive.
- Payment added and payment paid email notifications are supported.
- The frontend [guru_project_payments] shortcode shows a permission-aware payment portal.

14. Activity History

Activity provides an audit trail of important actions. Global Activity and the full Project Activity tab now use server-side AJAX pagination at 100 activities per page. Every activity entry displays its date and time using the WordPress site date/time settings.

Search and date drill-down

- ? Search activity text and actor name/email. Global Activity also searches project titles.
- ? Date presets: All Activity, Today, Last 7 Days, Last 30 Days, and This Month.
- ? Custom From Date and To Date selectors use inclusive whole-day filtering.
- ? Search, filters, and Previous/Next pagination update through AJAX with no page reload.
- ? Result counts show the current range, such as Showing 101-200 of 687 activities.
- ? Sensitive activity is filtered before totals and pagination are calculated.

15. Notifications and Cron

Email notifications are sent through WordPress wp_mail() using deduplication records in the plugin notifications table. The daily due-notification process runs on the gpurprma_daily_notifications cron hook.

Notification	Primary trigger / recipient pattern
Project created	Project creation.
Project member added	User added to a project.
Project completed	Project completion.
Task assigned	Task assignment.
Task completed	Task completion.
Milestone completed	Milestone completion.
New discussion	New top-level project discussion.
Payment added	Payment record created.
Payment marked paid	Payment set to Paid: Yes.
Task due soon	Daily cron, assigned users, within Due Soon Window.
Task overdue	Daily cron, assigned users, due date before today.
Task status changed	Task changes stage/status.
Task note added	Task note created.
Task reassigned	Assignee list changes.
Milestone due soon	Daily cron, project managers.
Milestone overdue	Daily cron, project managers.
Project member removed	Project member removal.
Project status changed	Project status changed.
Task due date changed	Task due date changed.
Milestone date changed	Milestone due date changed.
New file uploaded	Project file upload.
Project due date changed	Project due date changed.

16. Settings

Settings are available only to Guru Super Admins and save through AJAX without a page reload.

Setting	Behavior
Due Soon Window	0 to 30 days. Used by both task and milestone due-soon checks.
Notification checkboxes	Enable or disable each supported notification event.
Delete all Guru Project Manager data on uninstall	Default unchecked. Enables destructive plugin-data cleanup only when explicitly saved as checked.

Destructive uninstall warning

When enabled, uninstall permanently removes Guru Project Manager custom tables, plugin options, Guru role user metadata, and the scheduled notification hook. It does not delete WordPress users or Media Library attachments.

Getting Started

The Settings screen also includes a compact Getting Started panel with copy buttons for [guru_project_manager] and [guru_project_payments], a four-step setup path, and a link to this documentation PDF.

17. Frontend Shortcodes

Shortcode	Behavior
	Full logged-in project workflow. Initial state can read guruprma_project, guruprma_tab, guruprma_status, and guruprma_sort query values.
[guru_project_payments]	Logged-in payment portal. Guru Super Admins see all payment users; normal users see their own payment records only.

Frontend AJAX lifecycle

- Frontend navigation requests guruprma_frontend_view.
- The server renders the requested project or project-list HTML.
- The frontend replaces the shortcode container HTML.
- The plugin rebinds dynamic project, Kanban, stage, upload, payment, team, user lookup, and milestone controls after replacement.
- Task editor instances are cleaned up before frontend HTML replacement.
- Modal behavior is designed to work inside the shortcode container as well as wp-admin.

Modal and editor compatibility

When the shortcode is placed inside another modal or popup, the task editor keeps WordPress link and Media Library overlays inside the active task dialog so stacking and focus remain reliable. Task editor instances are removed before AJAX content replacement.

18. AJAX Architecture

All 30 public AJAX handlers in v1.0 verify the `guruprma_ajax` nonce with `check_ajax_referer()` before processing request data. The centralized `verify_ajax()` method then enforces the logged-in requirement, while each handler applies action-specific capability and object checks.

Area	AJAX actions
Projects	Save project, search users, add/remove project member, toggle manager override, frontend view.
Kanban	Save full stage configuration, add/update/delete/reorder legacy stage endpoints, set completed stage, move task.
Tasks	Get task, save task, add task note, move task, complete/reopen task, delete task.
Milestones	Get milestone, set milestone task, save milestone, complete/reopen milestone, delete milestone.
Team	Save global Guru user role.
Payments	Save, toggle paid state, delete.
Collaboration	Add discussion, upload file.
Settings	Save settings.

Server-side security

- Nonce verification is centralized, but capability checks remain specific to the requested object/action.
- Project IDs, user IDs, task IDs, milestone IDs, and stage IDs are validated server-side.
- Sensitive-item access is rechecked on the server.
- AJAX is used for data mutations and frontend navigation wherever appropriate in the plugin.

19. Data Model, Migration and Uninstall

Guru Project Manager stores operational data in dedicated WordPress-prefixed custom tables.

Table group	Stored data
Projects / project_users	Projects, membership, manager overrides.
Stages / tasks / task_users / task_notes	Kanban stages, tasks, assignees, notes.
Milestones	Project milestones and milestone state.
Payments	Payment records.
Discussions	Topics and replies.
Files	References to WordPress Media Library attachments.
Activity	Audit history.
Notifications	Notification deduplication/sent records.

Migration behavior

- The plugin tracks a database version and runs upgrades when needed.
- The current schema includes migration logic for reserved sensitive columns and project-specific Kanban stages.
- Existing fixed workflow statuses are mapped into per-project stage records and stage IDs during the Kanban migration.
- Fresh installs receive the default five-stage workflow.

Uninstall behavior

Setting state	Result
Unchecked, default	Uninstall returns without deleting Guru Project Manager data.
Checked	Uninstall clears the cron hook, drops the plugin allowlisted custom tables, removes <code>_guruprma_role</code> user metadata, and deletes plugin options.
Always preserved	WordPress user accounts and Media Library attachments.

20. Troubleshooting

Symptom	First checks
User cannot see a project	Confirm login, Guru role, project membership, and project status/filter.
User cannot manage a project	Confirm Guru role or Manager Override and project membership.
Sensitive task missing	Only Guru Super Admins can view sensitive tasks.
Task assignee search returns nobody	Confirm users are already project members and search at least two characters.
Milestone task dropdown is empty	Tasks already assigned to another milestone are intentionally excluded.
Manage Stages changes disappear	Changes are draft-only until Save Stages is clicked.
Manage Stages will not save	Confirm at least one stage remains, exactly one completed stage is selected, all active stages have names, and removed stages with tasks have a valid destination.
Frontend control stops responding	Refresh the shortcode view and confirm no browser JavaScript error is blocking dynamic rebinding.
Email not received	Check WordPress mail delivery, notification setting, user email address, and spam filtering.
Due notifications late	WP-Cron is traffic-driven by default. Confirm cron is running reliably.
Settings not persisting	Confirm the logged-in user is Guru Super Admin and the AJAX request is not being blocked.

21. Release and Regression Checklist

- Plugin header Version, GURUPRMA_VERSION, readme Stable tag, and documentation version all equal 1.0.
- Run WordPress Plugin Check against the exact final ZIP.
- Fresh install: activate and confirm all custom tables are created.
- Upgrade test: confirm existing projects/tasks migrate into project-specific Kanban stages.
- Create, edit, complete, reopen, and delete a task.
- Add/remove task assignees through AJAX.
- Create a milestone, add available tasks, edit it, add/remove tasks through AJAX, complete, reopen, and delete it.
- Create a stage draft, rename/reorder/add/remove stages, close without saving and confirm no change.
- Repeat Manage Stages and click Save Stages, then confirm the main board updates immediately.
- Add/remove project members and Manager Overrides from Project Overview > Team.
- Test global Guru role changes from Team.
- Upload a file and post a discussion/reply.
- Add, toggle, and delete payments as Guru Super Admin.
- Test both frontend shortcodes as Super Admin, Manager, Team Member, and Viewer / Client.
- Test task/milestone notifications and WP-Cron due notifications.
- Uninstall with delete-data unchecked and verify data remains.
- Reinstall, enable delete-data, uninstall, and verify only Guru Project Manager-owned data is removed.
- Run a final em dash scan on all user-facing plugin text.
- If a future release adds a plugin-integrity SHA256 file, regenerate it after every final file change.

- Test modal keyboard focus, Escape behavior, link editing, and Media Library use inside a parent popup/modal.
Guru Way

Simple. Lean. Powerful. Keep the workflow obvious, permission-aware, AJAX-driven where appropriate, and free of unnecessary page refreshes.

Guru Plugins

Browse all plugins: <https://plugins.guru-is.com>

Support the project: <https://plugins.guru-is.com/#support>

22. Getting Started, Modal Compatibility and Accessibility

The v1.0 Settings screen includes a compact Getting Started panel for Guru Super Admins. It keeps onboarding inside the plugin without adding a wizard or another configuration system.

Getting Started

- Copy [guru_project_manager] for the full permission-aware frontend project workspace.
- Copy [guru_project_payments] for the permission-aware frontend payment portal.
- Create a project, add WordPress users to the project team, assign work, then place the shortcode where the team will use it.
- Open the bundled documentation PDF directly from Settings or the plugin navigation.

Nested Modal and Editor Behavior

The frontend shortcode can be rendered inside a compatible site modal or popup. Guru Project Manager keeps its own dialogs modal-aware and dynamically initializes the native WordPress task editor only when needed.

- WordPress link and Media Library overlays are temporarily docked inside the active task modal so parent modal stacking and focus traps do not hide or block them.
- Escape does not close the Guru task modal while a WordPress editor overlay is open.
- Task editor instances are removed before AJAX view replacement to prevent stale TinyMCE instances and duplicate editor state.

Dialog Accessibility

- Guru dialogs expose dialog semantics with aria-modal and descriptive labels.
- Opening a dialog moves focus into it and closing returns focus to the control that opened it when that control still exists.
- Tab and Shift+Tab stay within the active Guru dialog, while native WordPress link and media overlays keep control of their own keyboard interaction.
- The topmost Guru dialog handles Escape, preventing multiple open dialogs from being dismissed at once.

WordPress.org Release Check

Before submitting v1.0, run WordPress Plugin Check against the exact final ZIP and perform a manual smoke test of frontend navigation, modal editing, link insertion, Media Library use, permissions, AJAX actions, notifications, and uninstall behavior.