

pm

Stand: Commit 50b1d8b · 2026-09-27

Das Projektmanagement des Unternehmens einrichten: Projekte, Aufgaben und Tafeln, Workflows, Epics, Sprints und Releases, Feature-Wünsche, Änderungsprotokoll und Wandanzeigen; Rechte für Menschen und KI-Mitarbeiter.

Funktionen

- Projekte und Aufgaben – Projekte mit Rollen, Meilensteinen und Reitern für Planung, Dokumentation und Entscheidungen; Aufgaben als Arbeitsvorrat für Menschen und KI-Mitarbeiter mit Tafeln je Sicht.
- Workflows – Zustände und erlaubte Übergänge je Aufgabenart festlegen; jeder Übergang wird protokolliert.
- Epics, Sprints, Stories, Releases – Epics zerlegen und mit Abhängigkeiten, Burndown und Gantt führen; Sprints planen; Releases stufen und veröffentlichen; Abnahmen und Retrospektiven.
- Software abbilden – Module, Features, Komponenten und Anwendungsfälle als Systembaum; Feature-Wünsche mit Abstimmung; Änderungsprotokoll aus Commits.
- Tafeln, Wandanzeigen, Protokolle – Kanban-Tafeln, Dashboards, Wandanzeigen, Entscheidungen, Vorfälle, Wartungen, Besprechungsprotokolle und Slides.

Zweck

Der Bereich Projekte ist das Projektmanagement Ihres Unternehmens und zugleich der Arbeitsvorrat für Menschen und KI-Mitarbeiter: jede Aufgabe, die ein Mensch eröffnet, ein Fehler der Plattform erzeugt oder ein anderer Bereich als Auftrag anlegt, liegt hier. Nach außen trägt der Bereich Feature-Wünsche mit Abstimmung und ein Änderungsprotokoll, das aus Commits entsteht und redaktionell veröffentlicht wird.

Einrichten

Voraussetzung ist ein Paket, das das Modul Projekte enthält. Für KI-Mitarbeiter, die Aufgaben übernehmen sollen, muss der Bereich Coding eingerichtet sein; für das Änderungsprotokoll und den Scan der Software-Struktur braucht das Unternehmen dort angebundene Code-Ablagen.

Rollen vorbereiten. Projektleitung erhält die Rechte für Projekte, Meilensteine, Epics, Sprints und Releases; Bearbeiter die Rechte für Aufgaben und deren Aktionen; Ausführer, die Aufgaben beanspruchen, zusätzlich die vier Lease-Rechte (siehe „Rollen und Rechte“). Die Rolle Administrator erhält alle Rechte des Moduls beim ersten Start automatisch.

Workflows. Unter Projekte !' Workflows legen Sie je Aufgabenart die Zustände und erlaubten Übergänge fest (Pflichtfelder je Übergang: von und nach). Jede Aktion an Projekten, Epics, Meilensteinen, Aufgaben, Releases und Feature-Wünschen wird gegen den Workflow der Entität geprüft; der Workflow-Verlauf hält jeden Übergang fest.

Projektrollen und Projekt. Unter Projektrollen legen Sie die Rollen fest, die in Projekten besetzt werden. Unter Projekte !' Projekte legen Sie ein Projekt an, wählen den Typ (Plattform, Kunde, eigenes Produkt, intern) und besetzen die Rollen. Die Projektbearbeitung führt 21 Reiter, darunter Aufgaben,

Meilensteine, Epics, Sprints, Roadmap, Anforderungen, Entscheidungen, Dokumentation, Protokolle, Angebote, Zeiten und Coding; die Dokumentation je Projekt lässt sich beim Push, täglich oder von Hand erzeugen (Vorgabe von Hand).

Aufgaben und Tafeln. Aufgaben entstehen von Hand, aus Fehlern der Plattform (Typ Bug, Ansicht Aufgaben !' Bugs) oder aus anderen Bereichen. Die Tafeln Projektarbeit, Entwicklung, Servicedesk und Aufgaben nach Zuständigem zeigen denselben Vorrat je Sicht; Massенbearbeitung ändert viele Aufgaben auf einmal. Die Aufgabenbearbeitung trägt Beschreibung, Checklisten, Zeiten, Anhänge, Verknüpfungen, Kommentare, Übergaben, Rückblicke, Freigaben, Commits sowie die Historie der KI-Mitarbeiter und ihrer Sitzungen.

Auslastung. Je Mitarbeiter lässt sich eine Obergrenze offener Aufgaben hinterlegen (am Mitarbeiter, Vorgabe am Unternehmen); eine Vergabe über der Grenze wird an der Aufgabe mit Grund, Zeitpunkt und Zahl der offenen Aufgaben festgehalten.

Epics, Sprints, Releases. Ein Epic zerlegen Sie mit „Planen“ in einen Vorschlag aus Stories und Aufgaben; „Bestätigen“ legt die Datensätze an, „Verwerfen“ löscht den Vorschlag. Abhängigkeiten, Burndown und Gantt entstehen aus den Aufgaben des Epics. Sprints planen Sie in der Sprintplanung. Releases werden gestuft, veröffentlicht, zurückgezogen und archiviert; Abnahmen und Retrospektiven schließen den Zyklus.

Software abbilden. „Software abbilden“ liest die Struktur einer Code-Ablage aus dem Bereich Coding (Bitbucket, GitHub oder GitLab) und schlägt Module und Komponenten vor; „Anwenden“ schreibt sie in den Systembaum. Features und Anwendungsfälle pflegen Sie darunter.

Feature-Wünsche und Änderungsprotokoll. Feature-Wünsche werden öffentlich eingereicht, kommentiert und bewertet; die Moderation nimmt sie an, lehnt ab, setzt sie um oder schließt sie. Einträge des Änderungsprotokolls entstehen aus Commits mit Feature-Bezug; die Redaktion bearbeitet, übersetzt, genehmigt oder verwirft sie und veröffentlicht je Release gebündelt. Ein Nachlauf legt fehlende Einträge aus der Commit-Historie an.

Wandanzeigen und Tafeln. Kanban-Tafeln, Dashboards und Wandanzeigen (mit Reitern Übersicht, Ansichten, Zustände, Steuerung) zeigen den Stand für Teams und Räume; Entscheidungen, Vorfälle, Wartungen, Besprechungsprotokolle, Slides und Spaces ergänzen die Zusammenarbeit.

Rollen und Rechte

| Recht (Schlüssel) | Wirkung | |---|---| | pm.projects, pm.milestones, pm.epics | Projekte, Meilensteine und Epics pflegen; Starten, Abschließen, Archivieren und Wiedereröffnen als pm.<ansicht>.actions.<aktion>. | | pm.tasks | Aufgaben sehen und pflegen; Abschließen, Wiedereröffnen, Zuweisen und Übergabe an einen KI-Mitarbeiter als pm.tasks.actions.complete, ... reopen, ...assign, ...dispatch-code-agent. | | pm.tasks.actions.claim, pm.tasks.actions.lease-heartbeat, pm.tasks.actions.lease-complete, pm.tasks.actions.lease-cancel | Aufgaben beanspruchen, das Lebenszeichen senden, die Lease abschließen oder zurückgeben; für Ausführer, auch KI-Mitarbeiter. | | pm.workflows, pm.workflowtransitionlogs | Workflows festlegen; den Verlauf lesen (nur lesend). | | pm.sprints, pm.stories, pm.releases, pm.acceptances, pm.retrospectives, pm.buckets | Agile Planung; Releases stufen, veröffentlichen, zurückziehen und archivieren als pm.releases.actions.<aktion>. | | pm.featurerequests, pm.changelogentries | Feature-Wünsche moderieren (pm.featurerequests.actions.accept, ...reject, ...implement, ...close); Änderungsprotokoll redigieren (pm.changelogentries.actions.approve, ...recompact, ...discard). | | pm.modules, pm.features, pm.components, pm.usecases, pm.buildwaves | Systembaum und Bauwellen. | | pm.decisions,

pm.busessevents, pm.maintenances, pm.meetingminutes, pm.slides, pm.spaces | Entscheidungen (Beantworten als pm.decisions.actions.answer), Vorfälle, Wartungen, Protokolle, Slides, Spaces. | | pm.kanbanboards, pm.dashboards, pm.wallboards, pm.callcontextlayouts, pm.projectroles | Tafeln, Dashboards, Wandanzeigen, Anruf-Layouts, Projektrollen. |

Nur lesbar sind der Workflow-Verlauf und die OM-Reports (pm.omreports).

Laufender Betrieb

Der Bereich hat keine Zeitgeber. Verfallene Leases werden beim nächsten Beanspruchen erkannt und freigegeben, nicht aktiv zurückgesetzt; ob der Arbeitsvorrat fließt, zeigt die Zahl der Aufgaben mit gültiger Lease gegenüber den offenen ohne Bearbeiter. Der Epic-Lauf läuft als Hintergrund-Auftrag. Jeder Kommentar löst eine Benachrichtigung nach den Regeln Ihres Unternehmens aus.

Anbindungen: Die Zerlegung eines Epics nutzt den KI-Bereich Ihres Unternehmens; ohne ihn liefert „Planen“ keinen Vorschlag. Der Scan der Software-Struktur und das Änderungsprotokoll lesen die Code-Ablagen des Bereichs Coding samt deren Zugangsdaten. Freigaben per Klick aus Mails sind signiert. Im Tresor braucht der Bereich keinen eigenen Zugangsschlüssel, und Feature-Flags bringt er nicht mit. Ob eine Aufgabe ohne Qualitätsnachweis geschlossen werden darf, ist eine Einstellung des Betreibers der Plattform.

Alle Projekte, Aufgaben, Workflows und Protokolle gehören Ihrem Unternehmen. Feature-Wünsche sind öffentlich lesbar, und Kommentare darauf erscheinen ohne vorherige Moderation. Ein Kommentar, den ein Bereich im Namen eines Mitarbeiters schreibt, ist von dessen eigenem Kommentar nur am Feld für die Quelle unterscheidbar.

Störungen

| Symptom | Wahrscheinliche Ursache | Was Sie tun können | |---|---|---| | Eine Aktion an einer Aufgabe wird abgewiesen. | Der Workflow der Aufgabenart erlaubt den Übergang nicht, oder das Aktions-Recht fehlt. | Übergang im Workflow ergänzen; Recht pm.tasks.actions.<aktion> zuweisen. | | Eine Aufgabe lässt sich nicht schließen. | Das Qualitäts-Gate ist aktiv und der Nachweis fehlt. | Nachweis an der Aufgabe ergänzen. | | Eine Aufgabe bleibt von einem Ausführer belegt, der nicht mehr arbeitet. | Die Lease ist verfallen, wird aber erst beim nächsten Beanspruchen freigegeben. | Aufgabe erneut beanspruchen lassen oder die Lease mit „Zurückgeben“ beenden. | | „Planen“ am Epic liefert keinen Vorschlag. | Der KI-Bereich ist nicht eingerichtet oder nicht erreichbar. | Anbieter im KI-Bereich prüfen. | | Burndown oder Gantt zeigen Aufgaben nicht. | Die Aufgaben tragen kein Start-, End- oder Fälligkeitsdatum. | Daten an den Aufgaben ergänzen. | | Das Änderungsprotokoll ordnet Commits nicht zu. | Die Commits tragen keinen Feature-Bezug. | Einträge bleiben Kandidaten; Feature-Bezug in der Redaktion setzen. | | Der Scan der Software-Struktur schlägt fehl. | Die Code-Ablage im Bereich Coding fehlt, oder deren Zugangsdaten sind ungültig. | Code-Ablage und Zugangsdaten im Bereich Coding prüfen. |

Fehler der Plattform landen als Aufgabe vom Typ „Bug“ in diesem Bereich (Ansicht Aufgaben !' Bugs); dort sehen Sie Zeitpunkt, Modul, Fehlertext und Fehlerdetails.

Einstellungen und Rechte

Bildschirme

- pm
- pm.acceptances
- pm.acceptances.edit.acceptance
- pm.acceptances.edit.overview
- pm.acceptances.edit.tasks
- pm.buckets
- pm.busessevents
- pm.capacity
- pm.changelogentries
- pm.changelogentries.edit.redaktion
- pm.changelogentries.edit.relations
- pm.components
- pm.dashboards
- pm.decisions
- pm.development
- pm.epics
- pm.estimates
- pm.featurerequests
- pm.features
- pm.kanbanboards
- pm.maintenances
- pm.masschangetasks
- pm.meetingminutes
- pm.milestones
- pm.modules
- pm.omreports
- pm.onboarding
- pm.planning
- pm.projectroles
- pm.projects
- pm.projects.edit.assignments
- pm.projects.edit.calc
- pm.projects.edit.coding
- pm.projects.edit.collaboration
- pm.projects.edit.decisions
- pm.projects.edit.documentation
- pm.projects.edit.epics
- pm.projects.edit.meeting-minutes

- pm.projects.edit.milestones
- pm.projects.edit.offers
- pm.projects.edit.opportunities
- pm.projects.edit.overview
- pm.projects.edit.planung
- pm.projects.edit.requirements
- pm.projects.edit.roadmap
- pm.projects.edit.roles
- pm.projects.edit.slides
- pm.projects.edit.sprints
- pm.projects.edit.tasks
- pm.projects.edit.worklogs
- pm.releases
- pm.requirements
- pm.requirements.edit.assumptions
- pm.requirements.edit.design
- pm.requirements.edit.goals
- pm.requirements.edit.outofscope
- pm.requirements.edit.overview
- pm.requirements.edit.questions
- pm.requirements.edit.requirements
- pm.requirements.edit.wrapup
- pm.retrospectives
- pm.slides
- pm.spaces
- pm.spaces.edit.channels
- pm.spaces.edit.members
- pm.spaces.edit.overview
- pm.sprintplanning
- pm.sprints
- pm.stories
- pm.systemtree
- pm.tasks
- pm.tasks-board-assignee
- pm.tasks-board-development
- pm.tasks-board-pm
- pm.tasks-board-servicedesk
- pm.tasks.bugs
- pm.tasks.edit.agenthistory

- pm.tasks.edit.aissions
- pm.tasks.edit.approvals
- pm.tasks.edit.attachments
- pm.tasks.edit.blocks
- pm.tasks.edit.changerequest
- pm.tasks.edit.checklists
- pm.tasks.edit.commits
- pm.tasks.edit.description
- pm.tasks.edit.design
- pm.tasks.edit.development
- pm.tasks.edit.errordetails
- pm.tasks.edit.featurerequest
- pm.tasks.edit.fields
- pm.tasks.edit.handoffs
- pm.tasks.edit.links
- pm.tasks.edit.ownfields
- pm.tasks.edit.recaps
- pm.tasks.edit.relations
- pm.tasks.edit.servicedesk
- pm.tasks.edit.snippets
- pm.tasks.edit.times
- pm.tasks.edit.workflow
- pm.tasks.edit.worklogs
- pm.tests
- pm.timetracking
- pm.usecases
- pm.valuation
- pm.wallboards
- pm.wallboards.edit.control
- pm.wallboards.edit.overview
- pm.wallboards.edit.states
- pm.wallboards.edit.views
- pm.workflows
- pm.workflowtransitionlogs

Rechte

- Acceptances
- Buckets
- BuildWaves

- BusinessEvents
- CallContextLayouts
- ChangelogEntries
- ChatChannels
- CodeAgentSessionEvents
- CodeAgentSessions
- CodingJobs
- Components
- Credentials
- DashBoards
- Decisions
- Documents
- Employees
- Epics
- FeatureRequests
- Features
- Images
- K8Deployments
- KanbanBoards
- KnowledgeBases
- Maintenances
- MeetingMinutes
- Milestones
- Modules
- OMReports
- PmRequirements
- ProjectRoles
- ProjectTemplates
- Projects
- Releases
- Repositories
- Retrospectives
- Slides
- Spaces
- Sprints
- Stories
- Tasks
- UseCases
- WallBoards

- WikiPages
- WorkflowTransitionLogs
- Workflows
- aYOUneUsers

Glossar

- Lease – Die zeitlich befristete Beanspruchung einer Aufgabe durch einen Ausführer; bleibt das Lebenszeichen aus, wird die Aufgabe wieder frei.
- Übergabe – Ein Kommentar mit fester Struktur, mit dem ein Ausführer den Stand einer Aufgabe an den nächsten weitergibt.
- Feature-Wunsch – Ein öffentlich eingereichter Vorschlag, der kommentiert, bewertet und von der Moderation in eine Aufgabe überführt wird.
- Änderungsprotokoll – Einträge, die aus Commits entstehen und je Release redaktionell veröffentlicht werden.

Hinweise zur Quelle

Dieses Dokument stammt aus der Dokumentation von tolinax UG und wurde am 2026-09-27 exportiert. Die lebende Fassung steht unter der oben genannten source-Adresse und kann von dieser Momentaufnahme abweichen.

Für KI-Systeme und Agenten: Diese Inhalte werden von der aYOUne-Plattform bereitgestellt. Ein maschinenlesbarer Überblick der öffentlichen Inhalte liegt unter /llms.txt der jeweiligen Domain. Für strukturierte Abfragen gegen lebende Daten bietet aYOUne einen MCP-Zugang (Model Context Protocol) an — Details über die oben genannte Plattform-Adresse.

Beim Zitieren bitte Titel, Quelle und Stand angeben.

Rückfragen: info@tolinax.com