

# Checkliste: Onboarding Automation Plugin

Diese Checkliste hilft dir, das Redmine Automation Plugin erfolgreich zu implementieren und zu nutzen. Sie ist in verschiedene Phasen unterteilt.

## Phase 1: Vorbereitung & Installation

- Redmine Version prüfen:** Stelle sicher, dass deine Redmine-Installation die vom Plugin benötigte Mindestversion unterstützt.
- Plugin herunterladen:** Lade die neueste Version des Redmine Automation Plugins von der offiziellen Quelle herunter.
- Plugin installieren:** Installiere das Plugin gemäß der offiziellen Installationsanleitung. Achte auf alle Abhängigkeiten.
- Plugin aktivieren:** Aktiviere das Plugin in der Redmine-Administration.
- Zugriffsrechte prüfen:** Stelle sicher, dass die Benutzer die notwendigen Berechtigungen haben, um das Plugin zu nutzen.

## Phase 2: Konfiguration & Test

- Erste Konfiguration:** Navigiere zur Plugin-Konfigurationsseite in der Redmine-Administration und konfiguriere die grundlegenden Einstellungen.
- Erstellung einer einfachen Automation Regel:** Erstelle eine einfache Automation-Regel, z.B. einen bestimmten Ticket TAG setzen bei der Erstellung eines neuen Issues.
- Test der Regel:** Aktiviere die Automation Test-Regel und führe sie aus. Überprüfe, ob sie wie erwartet funktioniert. Schau Dir hierzu auch das Redmine Anwendungslog (/log) an, ob es Fehlermeldungen gibt.
- Fehlerbehebung:** Falls Fehler auftreten, prüfe die Nachricht im Redmine Log und versuch den Fehler zu lösen. Nutze die Plugin-Dokumentation oder wende dich an den Support wenn Dir dies nicht gelingt.
- Test verschiedener Szenarien:** Teste verschiedene Bedingungen und Aktionen mit deiner Automation-Regel. Funktioniert sie wie geplant?
- Überprüfung der Trigger:** Stelle sicher, dass die Trigger korrekt verwendet wurden, um die automatisierte Aktion auszulösen.
- Überprüfung der Bedingungen:** Stelle sicher, dass die festgelegten Bedingungen zutreffen, um die automatisierte Aktion für die richtigen Entitäten auszulösen.
- Überprüfung der Aktionen:** Überprüfe, ob die Aktionen korrekt ausgeführt werden.
- Log-Dateien überprüfen:** Überprüfe regelmäßig die Log-Dateien des Plugins auf Fehler oder Warnungen.

## Phase 3: Implementierung & Feinabstimmung

- Definition von Use Cases:** Identifiziere die wichtigsten Anwendungsfälle, für die du das Plugin einsetzen möchtest.
- Ausarbeitung von Automation Regeln:** Überlege Dir detaillierte Automation-Regeln für jeden Use Case.
- Testing & Anpassung:** Teste die ausgearbeiteten Regeln gründlich und passe sie bei Bedarf an.
- Dokumentation:** Dokumentiere die ausgearbeiteten Regeln, wenn notwendig.
- Schulung anderer Benutzer:** Schule die relevanten Benutzer in der Verwendung des Plugins, wenn ihr euch die Arbeit teilt.
- Überwachung und Wartung:** Überwache die Performance der Automation Regeln und führe regelmäßige Wartungsarbeiten durch.
- Monitoring von Fehlern:** Richte für potenzielle Fehler eigene Abfragen für das Reporting Log ein.

# Checkliste: Onboarding Automation Plugin

## Phase 4: Fortlaufende Verbesserung



**Feedback sammeln:** Sammle Feedback von den Anwendern über die Effektivität der Automation Regeln.



**Optimierung der Regeln:** Optimierte die Regeln basierend auf dem Feedback und der Performance-Überwachung.



**Neue Use Cases identifizieren:** Identifiziere weitere Anwendungsfälle für den Einsatz des Plugin.

### Zusätzliche Hinweise:

- Erstelle Backups deiner Redmine-Datenbank vor der Installation und Konfiguration des Plugins.
- Lies die Plugin-Dokumentation sorgfältig durch.
- Nutze die verfügbaren Ressourcen, wie z.B. Dokumentation, Blogbeiträge oder den Support, um Fragen zu klären.
- Sei Vorsichtig im Umgang mit Regeln, die eine Löschaktion auslösen. Prüfe diese sorgfältig.
- Prüfe die Aktualität deiner Bestandsregeln regelmäßig. Neue Tracker, Projekte, Benutzerdefinierte Felder etc. erfordern gegebenenfalls eine Anpassung der Regel-Bedingungen.

**Diese Checkliste bietet einen umfassenden Überblick. Die spezifischen Schritte können je nach deinen Anforderungen und der Komplexität deiner Automatisierungsprozesse variieren.**

