



Redmine – projektmenedzseri szintű teljes körű útmutató

Készítette: ChatGPT

Redmine – Projektmenedzseri szintű teljes körű útmutató

Ez a dokumentum részletesen bemutatja a Redmine projekt- és hibakövető rendszert a kezdeti megismerkedéstől egészen a projektmenedzseri szintű konfigurációig. Az útmutató a legfontosabb funkciókat és a kapcsolódó beállításokat foglalja össze; a hivatkozásoknál szereplő sorszáмок a Redmine dokumentációjából származó bizonyítékokra utalnak.

1. Bevezetés

****Mi az a Redmine?**** A Redmine rugalmas, webes projektmenedzsment alkalmazás, amely Ruby on Rails keretrendszeren épül. Több platformon és több adatbázison is futtatható. A rendszer teljesen nyílt forráskódú és a GNU GPL v2 licenc alatt érhető el.

****Főbb funkciók:****

- Több projekt egyidejű támogatása.
- Rugalmas szerep-alapú jogosultságkezelés.
- Rugalmas hibakövető rendszer.
- Gantt-diagram és naptár.
- Hírek, dokumentumok és fájlok kezelése.
- RSS csatornák és e-mail értesítések.
- Projekt-szintű wiki és fórum.
- Időkövetés.
- Egyéni mezők projektekhez, feladatokhoz és felhasználókhoz.
- Verziókezelő rendszerek integrációja (SVN, Git, Mercurial stb.).
- E-mailben történő feladat létrehozás.
- LDAP hitelesítés és önregisztráció.
- Többnyelvű felület és több adatbázistámogatás.

Ezek a funkciók teszik a Redmine-t olyan eszközzé, amely egyaránt alkalmas informatikai projektek hibakövetésére és általános projektmenedzsmentre.

2. Telepítés és beállítás

2.1 Telepítési lehetőségek

- ****Helyi telepítés:**** A Redmine forráskódja letölthető a hivatalos oldalról, és telepíthető Ruby on Rails környezetben. A telepítés pontos lépéseit a „Installation guide” rész tartalmazza (nem szerepel ebben a dokumentumban).
- ****Felhő szolgáltatások:**** Létezik több Redmine-alapú felhőszolgáltatás (például Redmine Hosting, RedmineUP Cloud), ahol az üzemeltetés és karbantartás nem a felhasználó feladata.

2.2 Első lépések

1. ****Új projekt létrehozása:**** A „Projektek” menüben az „Új projekt” gombra kattintva lehet új projektet definiálni. A projekt létrehozásakor megadható a név, leírás, azonosító és a publikusság, valamint beállítható, hogy alprojektként melyik szülőprojekthez tartozzon.
2. ****Modulok kiválasztása:**** A projekt beállításai között a „Modulok” fülön ki lehet választani,

hogy mely funkciók legyenek aktívak. Modulok kikapcsolása nem törli az adatokat, csak elrejt a funkciót. Az alapmodulok közé tartozik az issue követés, időkövetés, hírek, dokumentumok, fájlok, wiki, repository, fórumok, naptár és Gantt-diagram.

3. ****Nyomkövetők kiválasztása:**** A „Trackerek” fülön állítható, hogy az adott projektben mely issue-típusokat (tracker) használjuk. Az adminisztrátor új trackereket hozhat létre és rendelhet hozzájuk egyedi munkafolyamatot.

4. ****Egyéni mezők kiválasztása:**** Ha vannak globális vagy projekt-specifikus egyéni mezők, a „Custom fields” fülön lehet bejelölni, melyek jelenjenek meg a projekthez tartozó feladatoknál.

5. ****Tagok hozzáadása:**** A „Tagok” lapon adhatók hozzá felhasználók vagy csoportok, és itt rendelhetők hozzájuk szerepek. Egy felhasználó több szereppel is rendelkezhet, amelyek kombinált jogosultságot biztosítanak.

6. ****Verziók:**** A „Verziók” lapon definiálhatók mérföldkövek (verziók), amelyekhez feladatokat rendelhetünk, és a roadmap funkcióval nyomon követhetjük a készültségi állapotukat. A verziókhoz nev, leírás és státusz rendelhető (open/locked/closed).

3. Szerepek és jogosultságok

A Redmine-ben a jogosultságokat szerepkörök (roles) csoportosítják. Egy szerep több jogosultságból áll, és meghatározza, hogy a projekt tagjai milyen műveleteket végezhetnek. Egy felhasználónak projektenként több szerepe is lehet, és a szerepek összesített jogosultságai érvényesülnek.

3.1 Szereptulajdonságok

A szerepek számos beállítással rendelkeznek:

- ****Név:**** a szerep megjelenített neve.
- ****Feladatok hozzárendelése:**** meghatározható, hogy ehhez a szerephez tartozó tagokhoz lehet-e feladatot rendelni.
- ****Feladatok láthatósága:**** beállítható, hogy a felhasználó az összes feladatot, a nem privát feladatokat (alapértelmezés), vagy csak a saját / hozzá rendelt feladatokat lássa.
- ****Időnaplók láthatósága:**** meghatározható, hogy a felhasználó minden időnaplót, vagy csak a sajátját lássa.
- ****Felhasználók láthatósága:**** megadható, hogy a felhasználó minden aktív felhasználót láthasson, vagy csak a látható projektek tagjait.

A szerephez tartozó jogosultságok a műveletek hosszú listájából választhatók ki. Ezek kiterjednek a projektek létrehozására, tagok kezelésére, modulok ki-/bekapcsolására, feladatok kezelésére, fórumokra, dokumentumokra, időkövetésre és sok más területre.

4. Nyomkövetők és munkafolyamatok

4.1 Trackerek

A „tracker” egy feladat-típus (pl. hiba, funkció, támogatás), amely meghatározza az issue-k űrlapját és munkafolyamatát. Minden trackerhez megadható a név, az alapértelmezett státusz, hogy megjelenjen-e a roadmapben, egyedi munkafolyamat, valamint hogy mely alapértelmezett és egyéni mezők jelenjenek meg. Új tracker létrehozásakor érdemes egy meglévő tracker munkafolyamatát lemásolni, majd testre szabni.

4.2 Státuszok és munkafolyamatok

A munkafolyamat határozza meg, hogy egy feladat mely státuszról mely státuszokba válthat át, szerepkörön és trackerenként. A munkafolyamat beállító képernyőjén szerep és tracker alapján választható ki az engedélyezett átmenetek mátrixa; ehhez a felhasználónak „Edit issues” jogosultságra van szüksége. A mező engedélyeknél azt is beállíthatjuk, hogy egy adott mező adott státuszban olvasható, kötelező vagy szerkeszthető legyen.

5. Feladat- és hibakövetés

5.1 A feladat fogalma

A Redmine-ben az „issue” (feladat) a rendszer alapegysége. Minden feladat egy projektbe tartozik, felelőse (owner) van, verzióhoz vagy mérföldkőhöz rendelhető és kapcsolatok hozhatók létre közöttük.

5.2 Kapcsolatok és alfeladatok

A feladatok között különböző kapcsolatokat lehet rögzíteni. A „Kapcsolódó” kapcsolattípus csupán linket ad a két feladat között; a „duplikálja/duplikált” kapcsolatok esetén egy feladat lezárása a másikat is lezárhatja; a „blokkolja/blokkolt” kapcsolat blokkolja a zárást; a „megelőzi/követi” kapcsolat sorrendet definiál, és a kezdő/időpontok egymáshoz igazodnak; a „lemásolva” kapcsolatok pedig a feladat klónozását jelzik. A feladatok alfeladatokra bonthatók, így hierarchikus struktúra építhető ki (a részletek a felületen keresztül érhetők el).

5.3 Watcherek (figyelők)

Mind a feladatok, mind a wiki oldalak esetén lehetőség van figyelőket (watchers) felvenni. A figyelők listája megjelenik a feladatnál, és az érintett felhasználók e-mail értesítést kapnak a változásokról. Projektadminisztrátorok másokat is hozzáadhatnak a figyelők listájához.

5.4 Feladatok létrehozása és frissítése

Új feladat létrehozásakor ki kell választani a trackert (pl. hiba, feladat), megadni a tárgyat, leírást, státuszt, prioritást, kategóriát, felelőst, határidőt, becsült időt, valamint tetszőleges egyéni mezőket. A státuszok listáját és a lehetséges átmeneteket a munkafolyamat szabályozza. A feladat frissítése során naplózható a munkaidő (lásd időkövetés), módosítható a státusz, hozzászólások fűzhetők hozzá, és csatolmányok tölthetők fel. A frissítéshez megfelelő szerepkör-jogosultság szükséges.

6. Időkövetés

A Redmine időkövető modulja lehetővé teszi, hogy nyomon kövessük, mennyi munkát fordítottunk egy konkrét feladatra vagy projektre. Minden időbejegyzéshez tevékenység (activity) rendelhető, és megjegyzéssel egészíthető ki. A becslések és a ténylegesen ráfordított órák rögzítésével a projektmenedzserek pontosabban tervezhetnek, és a rendszer felhasználónként képes számlázható órákat nyilvántartani.

6.1 Modul engedélyezése

Az időkövetés modul projekt-szinten kapcsolható be vagy ki a „Modulok” fülön.

6.2 Idő naplózása

- ****Feladatnál:**** A feladat nézetében a „Log time” gombra kattintva rögzíthetünk ráfordított időt. Ugyanez elérhető a feladat frissítésének képernyőjén is.
- ****Projekt szinten:**** Ha a „Log time” párbeszédben nem választunk ki feladatot, akkor a projekt egészéhez naplózhatunk órákat.

6.3 Időformátumok

A becsült és a ráfordított idő több formátumban megadható: külön óra és perc (pl. `1h`, `30m`, `1h30`), illetve decimális formátum (pl. `1.5`, `1.5`), amely 1,5 órát jelent. A decimális időformátumnál a rendszer a tizedes értéket percként értelmezi (pl. 0,10 = 6 perc).

6.4 Időnaplózás commit üzenetből

Verziókezelő integráció esetén a commit üzenetben feltüntetett issue hivatkozás és időmennyiség alapján a rendszer automatikusan naplózhat időt. Ehhez engedélyezni kell az időnaplózást a repository beállításoknál; a commit üzenetben a beállított kulcsszót (`refs` vagy más) követően kell megadni az issue azonosítót és az időt (pl. `Fixed something, refs #435 @1h30`).

7. Gantt-diagram és Naptár

7.1 Gantt-diagram

A Gantt-diagram azokat a feladatokat jeleníti meg, amelyekhez kezdődátum és határidő tartozik, vagy olyan verzióhoz vannak rendelve, amelynek van határidője. A diagramban vizuálisan látható a projekt ütemezése és a feladatok egymáshoz viszonyított időtartama.

7.2 Naptár

A naptár havi nézetben jeleníti meg a projektet. Minden feladat megjelenik, amelyhez legalább kezdődátum tartozik, és jelöli a határidőt is. A naptár a határidővel rendelkező verziókat is listázza.

****Modul engedélyezés:**** A naptár modul projekt-szinten be- vagy kikapcsolható a modulok listájában. A hozzáférést szerepkörök szerint lehet engedélyezni vagy tiltani. A naptárban a feladatok listájához hasonlóan szűrők állíthatók be.

8. Egyéni mezők

Az egyéni mezők (custom fields) lehetőséget adnak arra, hogy további adatokat rögzítsünk feladatokhoz, időbejegyzésekhez, projektekhez, verziókhoz, felhasználókhoz és más entitásokhoz.

8.1 Mezőtípusok

Az egyéni mezők típusa lehet: logikai (checkbox), dátum, lebegőpontos szám, egész szám, hivatkozás (URL), lista (legördülő), hosszú szöveg (többsoros, támogatja a formázást), rövid szöveg, felhasználó-referencia vagy verzió-referencia.

8.2 Érvényesítés és értékek

Meghatározható minimális és maximális hosszkorlát, illetve reguláris kifejezés az érték validálására. A lista típusú mezőknél több érték kiválasztása is engedélyezhető, megadhatók a lehetséges választékok és alapértelmezett érték.

8.3 Értékek linkelése URL-hez

Egyes mezőtípusoknál (logikai, dátum, lista, egész szám, szöveg) beállítható, hogy a mező értéke URL-ként működjön. A link sablonjában helyettesítő változók használhatók, például ``%value%`` (a mező értéke), ``%id%`` (az érintett objektum azonosítója), ``%project_id%`` és ``%project_identifier%``.

9. Wiki, dokumentumok és fórumok

9.1 Wiki oldalak

Projektenként saját wiki érhető el. Új wiki oldal létrehozásához egyszerűen hozzunk létre egy linket a ``[[ÚjWikiOldal]]`` formátumban; a linkre kattintva megnyílik az új oldal szerkesztője. Lehetőség van szülő/gyermek viszony kialakítására, így a „Cím szerinti index” nézetben a hierarchia alapján rendezett lista jelenik meg.

Az egyes wiki oldalakon található eszköztáron a következő műveletek érhetőek el:

- **Szerkesztés:** az oldal tartalmának frissítése; a szülőoldal módosítható.
- **Figyelés:** a „Watch” gombra kattintva feliratkozunk az oldal értesítéseire.
- **Átnevezés/Áthelyezés:** az oldal átnevezése vagy más projektbe történő áthelyezése; a linkek átirányítása választható.
- **Zárolás:** az oldal zárolása, amelyet csak megfelelő jogosultsággal lehet feloldani.
- **Törlés:** az oldal törlése – a rendszer megerősítést kér.
- **Előzmények:** megtekinthető az oldal korábbi verzióinak listája; bármely verzió visszaállítható.

9.2 Dokumentumok, fájlok és hírek

A projekt „Dokumentumok” modulja lehetővé teszi különböző típusú dokumentumok feltöltését és kategorizálását. A „Fájlok” modulban a projekt fájlok tölthetők fel, verziózhatók és letölthetők. A „Hírek” modul segítségével a projekt aktualitásairól lehet bejegyzéseket közzétenni. Ezek a modulok az alaplmodulok között szerepelnek.

9.3 Fórumok (Boards)

Ha a projektben engedélyezve van a „Boards” modul, a tagok témákat hozhatnak létre, hozzászólhatnak, szerkeszthetik vagy törölhetik a bejegyzéseket a szerepüknek megfelelő jogosultságok szerint.

10. Repository és verziókezelő integráció

Ha a projektben engedélyezve van a „Repository” modul, a „Repository” fül alatt böngészhetjük a projekt verziókezelő táráát. Megnyithatjuk a mappákat, a fájlokat, megtekinthetjük az egyes revíziókhoz tartozó commit üzeneteket, és tetszőleges revíziót tölthetünk be a jobb felső mezőbe beírt azonosítóval. A commit üzenetekben hivatkozhatunk issue azonosítókra és naplózhatunk ráfordított időt (lásd 6. fejezet).

11. Bővítmények

A Redmine funkcionalitása pluginekkal bővíthető. A bővítmény telepítésének lépései:

1. Másoljuk a bővítmény könyvtárát a `plugins` mappába.
2. Ha a bővítmény adatbázis migrációt igényel, futtassuk a ``bundle exec rake redmine:plugins:migrate RAILS_ENV=production`` parancsot (előtte készítsünk biztonsági mentést).
3. Indítsuk újra a Redmine-t.

Ezután a bővítmény megjelenik az „Administration → Plugins” menüben, ahol engedélyezhető és konfigurálható. A bővítmények listáját a hivatalos Plugin Directory tartalmazza. Bővítmények például: Agile plugin (Kanban táblák, Sprint-kezelés), CRM plugin, Helpdesk plugin stb.

12. Projektmenedzseri legjobb gyakorlatok

Ez a fejezet a Redmine hatékony használatához ad iránymutatást projektmenedzseri szinten.

- ****Projektstruktúra:**** Hozzunk létre szülő- és alprojekteket a nagyobb programok áttekinthetőségéért, használjuk a verziókat a mérföldkövekhez.
- ****Szerepek:**** Definiáljunk világos szerepköröket (pl. projektmenedzser, fejlesztő, tesztelő). A szerepek megfelelő beállításával szabályozhatjuk a feladatok láthatóságát és szerkeszthetőségét.
- ****Munkafolyamat:**** Tervezéskor határozzuk meg a feladatok státuszait (pl. Új, Folyamatban, Tesztelés, Kész), majd hozzunk létre trackereket és állítsuk be a munkafolyamatot szerepkörönként.
- ****Időkövetés:**** Kötelezzük a csapatot az időnaplózásra; a decimális formátumot használva egyszerűbben összegezhetők a ráfordított órák. Vizsgáljuk a „Spent time report” oldalakat a projekt állapotának elemzéséhez.
- ****Dokumentáció:**** Használjuk a wiki modult a követelmények, döntések és jegyzetek dokumentálására; alkönyvtárakban rendszerezhetjük a fejezeteket. A figyelés funkcióval a résztvevők értesülnek a változásokról.
- ****Egyéni mezők:**** Definiáljunk projekt-specifikus mezőket (például üzleti prioritás, becsült költség), és szabjuk testre a listákon látható oszlopokat. Használjuk a reguláris kifejezéseket az értékek validálására.
- ****Kommunikáció:**** Az email értesítések és a figyelők segítenek abban, hogy a csapat naprakész legyen a módosításokról. A fórummodulban strukturáltan gyűjthetők a megbeszélések.
- ****Integráció:**** Kapcsoljuk össze a Redmine-t a verziókezelő rendszerrel; így a commit üzenetek automatikusan frissítik a feladatokat és naplózzák az időt. További integrációk (CI/CD, chat) bővítményekkel érhetők el.

13. Összegzés

A Redmine egy sokoldalú, nyílt forráskódú projektmenedzsment és hibakövető eszköz, amely támogatja a több projektet, rugalmas jogosultságkezelést, időkövetést, egyéni mezőket és számos kiegészítő modult. A rendszer testreszabható trackereken és munkafolyamatokon alapul, amelyek lehetővé teszik, hogy a projektmenedzserek a saját folyamataikhoz igazítsák a feladatok életciklusát. A moduláris felépítésnek köszönhetően a Redmine funkciói bővítményekkel tovább bővíthetők. Ha a fenti iránymutatásokat követjük, hatékonyan használhatjuk a Redmine-t komplex projektek tervezésére, végrehajtására és nyomon követésére.