

著作権処理状態管理システム開発における
既存OSSの調査報告書

目次

- 1 著作権処理状態管理システム開発における既存OSSの調査報告書
- 2 OSS導入によるメリット・デメリット
- 3 機能一覧
- 4 画面イメージサンプル
 - 1 OTRS サンプルイメージ
 - 2 Trac サンプルイメージ
 - 3 BPM Suite サンプルイメージ
 - 4 Redmine サンプルイメージ
- 5 見積機能別マトリクス表

1. 著作権処理状態管理システム開発における既存OSSの調査報告書

1-1. 調査対象OSS

OSS	OS	区分	サーバ	言語	DB	ライセンス	備考
Trac	Win,Linux	PJ管理	Apache	Python	SQLite	BSD	無保証 BSDライセンスのソースコードを複製・改変して作成したオブジェクトコードを、ソースコードを公開せずに頒布可能
Redmine	Win,Linux	PJ管理	Apache	Ruby	MySQL	GPL	無償・オープンソース
Questetra BPM Suite	Win,Linux	BPM	Tomcat	java,flash	H2		有償・カスタマイズ禁止
OTRS	Win,Linux	ヘルプデスク	Apache	Perl	MySQL	GPL3	無償・オープンソース
Sugar Community Edition	Win,Linux	顧客管理	Apache	PHP	MySQL	GPL3	無償・オープンソース
スクラッチ開発	Win,Linux	オリジナル	Apache	PHP	MySQL		

※OSSを利用したシステムでLinuxを選択する

※GPLソフトウェアは自由に改造して公開していい

1. 著作権処理状態管理システム開発における既存OSSの調査報告書

1-2. 補足説明(コメント)

Trac	使用されているDB(SQLite)は小規模の物では使えるが中規模以上のDBとしては向かない。
Trac、Redmine	・チケット発行画面に入力項目(例えば著書名、著者名等)の追加は可能だが、表示する場所までは指定できない。 ・1つの情報を複数の担当者が作業するような業務フローの作りこみは難しい。
Questetra BPM Suite	・カスタマイズは販売元であれば可能と思われる。 ・登場人物とビジネスロジックを組み合わせる処理を作成していく。 ・Tomcatを使用しているため、サーバに搭載するメモリを多く必要とする。
OTRS	インプットを電話またはメールから開始し、クレーム対応等で利用するようなヘルプデスクシステム。
Sugar Community Edition	顧客管理システムに向いているが、状態管理には向かない。
スクラッチ開発	ベースとなるOSやモジュールはOSSを利用する。Webアプリは0から開発を行う。 OSSの解析を行いつつカスタマイズを行うよりも実工数が減る可能性もある。
全て	カスタマイズ以前に使用するOSSを理解しなければ、エンドユーザへのレクチャーも難しい。

1-3. 開発手法案

OSSを使用した開発では、選択したOSSの仕様調査期間、OSSの既存バグによるアップデート等による動作再検証等に工数がかかる可能性が高い。
よって、以下の開発手法を提案する。

OS及びミドルウェアはOSSを利用する
OS LINUX
DB MySQL、PostgreSQL
webサーバ APACHE、PHP

webアプリについてはスクラッチ開発を行う。

2. OSS導入によるメリット・デメリット

2-1. OSS導入によるメリット

- ・導入コストが削減できる。
- ・自由なカスタマイズ開発が可能である。

2-2. OSS導入によるデメリット

- ・ソフトウェアのサポートが基本的でない。
- ・カスタマイズを行うことにより、テスト工数を増やす必要がある。
- ・ソースコードを読んで使いかたを理解しなければならないため、検証期間等が必要となる。
- ・既存OSSをバグ対応等のためにシステムアップデートを行った際、システムが動かなくなる可能性がある。
- ・メインシステムのログイン認証とは別の認証が必要のため、メインシステムのユーザデータとの連携が必要である。
- ・各処理を行ったタイムスタンプを記録するためのフィールドを入力項目に追加する必要がある。
- ・帳票(PDF)出力を行うためのツールを別途用意する必要がある。

2-3. 本案件を実装する際の外部データ連携の問題

- ・外部APIの仕様が変更した場合、プログラムの改修を行う必要がある。
- ・外部APIを利用するため、当システムを構築するサーバから外部へ接続可能な環境が必要である。
- ・外部APIから抽出する項目を提示していただく必要がある。

3. 機能一覧

No	機能名称	処理概要	備考	工数(人月)	金額
1	要件定義			2.00	¥1,400,000
1	基本仕様設計	ビジネスプロセスフロー構築	著作権処理状態項目(1.1)の各ステータスとタイムスタンプ(1.3.2)を処理する。	1.00	¥700,000
2	著作処理情報管理	ローカルDB構築	著作処理情報(1.2)の各種情報をDBにて管理する。	0.50	¥350,000
3	著作権処理状態管理	処理状態照合機能	インプット情報の照合機能を設ける(1.4)。	0.50	¥350,000
4		督促メール配信機能	「出版者許諾依頼中」ステータスに督促メールを配信する。 配信するメールのテンプレート(ひな型)を管理する。送信はユーザの使用しているメールソフトで行う。	0.25	¥175,000
5		メタデータ編集機能	データインポートまたは新規登録したメタデータの編集作業を行う。	0.50	¥350,000
6	著作権ポリシーDB参照機能	API接続機能	メタデータにある情報から著作権ポリシーDBへAPIで接続し、データを格納する。	0.50	¥350,000
7	出版者問合せ機能	ローカル出版者DB登録機能	出版者著作権ポリシー(1.2.⑤)関連のデータを管理する。	0.50	¥350,000
8		出版者の照合機能	出版者情報(1.2.⑥)のデータを管理する。	0.50	¥350,000
9		メール送信機能	出版者に対してメールでの問い合わせを行う。送受信はユーザの使用しているメールソフトで行う。 FAXでの問合せは画面から印刷した紙媒体を手作業で送信する。 出版者からの受信メール内容は手動でシステムに登録する。	0.50	¥350,000
10		著者版/出版社版管理	著者版/出版者版情報を管理する。	0.25	¥175,000
11		エンバーゴ期間管理	エンバーゴ期間情報を格納する。	0.25	¥175,000
12		登録可能対象データメール送信	条件を満たしたメタデータ情報をユーザ(当システムの関係者)にメール配信する。 送信はユーザの使用しているメールソフトで行う。	0.25	¥175,000
13		エンバーゴ種類設定機能	メタデータ中のエンバーゴ種類を操作する。	0.25	¥175,000
14	帳票出力機能	公開許諾書作成	公開許諾書をPDFで出力する。	0.25	¥175,000
15	サブシステム連携	教員問合せシステム連携①	教員問合せシステムに対してデータを引き渡す。	0.25	¥175,000
16		教員問合せシステム連携②	督促メールで使用するメタデータの抽出を行う。 送信はユーザの使用しているメールソフトで行う。	0.25	¥175,000
17		リポジトリ投稿システム連携①	本文ファイル名を取り込む。	0.25	¥175,000
18		リポジトリ投稿システム連携②	メタデータと本文ファイルの照合機能を設ける。	0.25	¥175,000
19		データソース抽出	メタデータの取り込みを行う。 データ取り込みの際には重複チェックを行い、登録可否の確認を行う。	0.50	¥350,000
20	データ入出力	データエクスポート①	著作権処理状態のデータをTSVファイルで出力する。 項目は固定で出力する。	0.25	¥175,000
21		データエクスポート②	機関リポジトリのデータインポートに利用可能なSWORD対応のTSVファイルを出力する。	0.25	¥175,000
22		データインポート	データエクスポート①形式でユーザが作成したメタデータを当サブシステムにインポートする。	0.50	¥350,000
23		データアップデート	当サブシステムのユニークIDがあるデータエクスポート①形式のデータはアップデートを行う。	0.50	¥350,000
24	統計機能	判明率	著作権ポリシーの判明率を表示する。	0.25	¥175,000
25		統計値	年度毎の取り扱い件数、許諾件数、処理中件数等の統計値を表示する。	0.25	¥175,000
26	設定作業	権限付与	当サブシステムを利用するユーザに機能毎の権限を付与する。	0.50	¥350,000
27	初期データ取り込み作業	データインポート(ローカルDB)	データソースからTSV形式で作成されたデータを初期データ用としてローカルDBにインポートを行う。	0.25	¥175,000
28		データインポート(ローカル出版者DB)	著作権ポリシーからTSV形式で作成されたデータを初期データ用としてローカル出版者DBにインポートを行う。	0.25	¥175,000
合計				12.50	¥8,750,000

4. 画面イメージサンプル(『OSS: Trac』のサンプルイメージ)

チケット登録 - SampleProje...

localhost/trac/SampleProject/newticket

vio

チケット登録

admin としてログイン中 ログアウト ヘルプ/ガイド Trac について 個人設定 more

Trac Lightning

SampleProject
My example project

Search

チケットの新規作成

属性

概要:

報告者:

詳細:

B I A

 wysiwyg textarea WikiFormatting が使用可能

分類: タスク

マイルストーン:

バージョン:

関係者:

終了予定日:

担当者:

優先度: 通常

コンポーネント: その他

キーワード:

開始予定日:

進捗率(%):

☐ このチケットにファイルを添付します

プレビュー

チケットの新規作成

※ 詳しい使い方は [TracTickets](#) を参照してください。

trac

Powered by Trac 0.12.ja1. By Edgewall Software.
MacOS theme - maintained by Olemis Lang.

Powered by
Trac Lightning
Version 3.0.6

Wiki

タイムライン

ロードマップ

リポトリブラウザ

チケットを見る

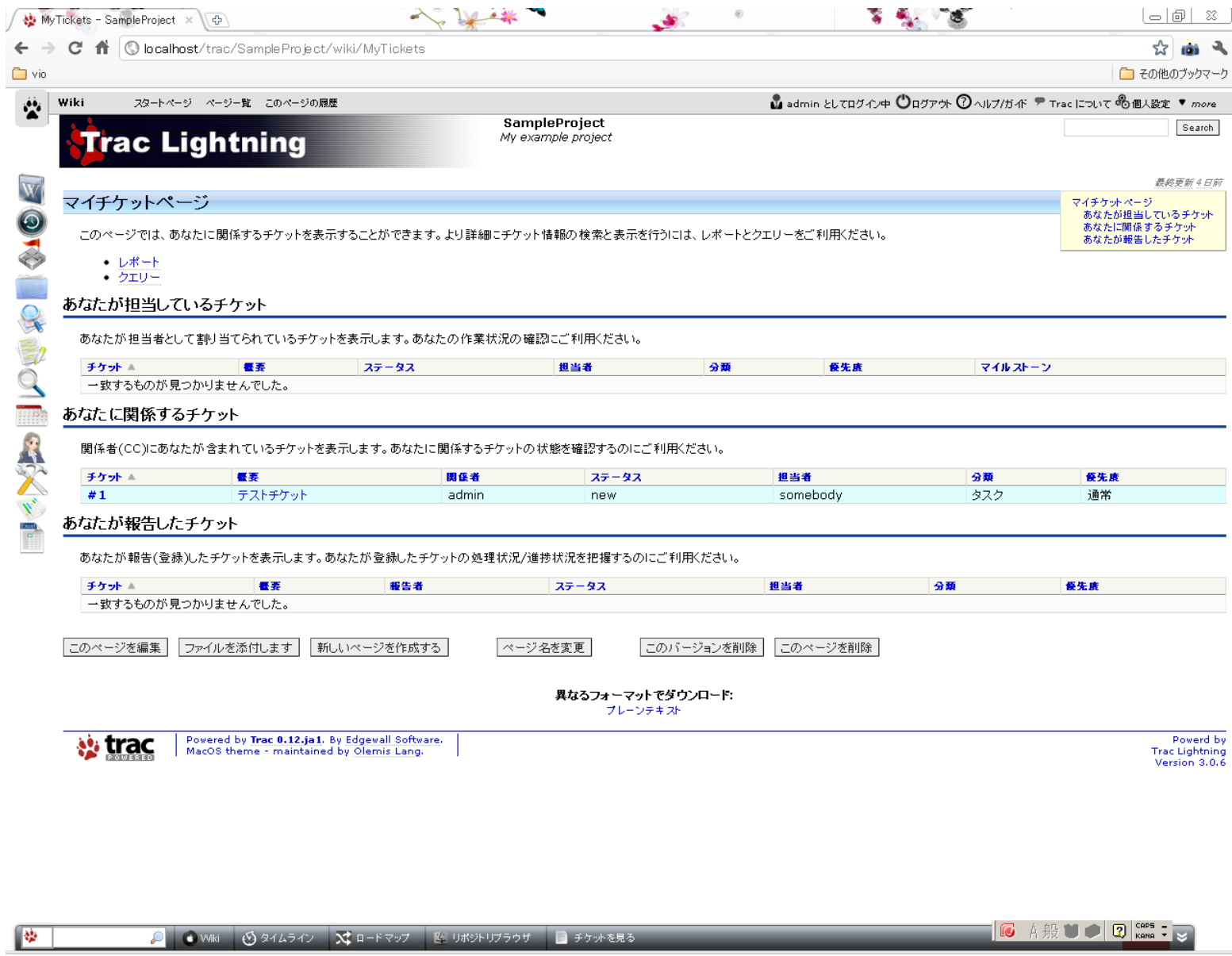
A 般

CAPS KANA

more

登録画面

4. 画面イメージサンプル(『OSS: Trac』のサンプルイメージ)



MyTickets - SampleProject

localhost/trac/SampleProject/wiki/MyTickets

Wiki スタートページ ページ一覧 このページの履歴

admin としてログイン中 ログアウト ヘルプ/ガイド Trac について 個人設定 more

Trac Lightning

SampleProject
My example project

マイチケットページ

このページでは、あなたに關係するチケットを表示することができます。より詳細にチケット情報の検索と表示を行うには、レポートとクエリーをご利用ください。

- レポート
- クエリー

あなたが担当しているチケット

あなたが担当者として割り当てられているチケットを表示します。あなたの作業状況の確認にご利用ください。

チケット	優先度	ステータス	担当者	分類	優先度	マイルストーン
一致するものが見つかりませんでした。						

あなたに關係するチケット

関係者(CC)にあなたが含まれているチケットを表示します。あなたに關係するチケットの状態を確認するのにご利用ください。

チケット	優先度	関係者	ステータス	担当者	分類	優先度
#1	テストチケット	admin	new	somebody	タスク	通常

あなたが報告したチケット

あなたが報告(登録)したチケットを表示します。あなたが登録したチケットの処理状況/進捗状況を把握するのにご利用ください。

チケット	優先度	報告者	ステータス	担当者	分類	優先度
一致するものが見つかりませんでした。						

このページを編集 ファイルを添付します 新しいページを作成する ページ名を変更 このバージョンを削除 このページを削除

異なるフォーマットでダウンロード:
プレテンキスト

Powered by Trac 0.12.1. By Edgewall Software.
MacOS theme - maintained by Olemis Lang.

Powered by
Trac Lightning
Version 3.0.6

ステータス確認画面

4. 画面イメージサンプル(『OSS: Trac』のサンプルイメージ)

The screenshot displays the Trac Lightning web interface in a browser window. The address bar shows the URL: `localhost/trac/SampleProject/ticket/1#comment:2`. The page title is "チケット #1 (accepted タスク)". The main content area shows a ticket summary for "テストチケット" (Test Ticket) with fields for reporter (テスト), assignee (admin), priority (通常), and status (accepted). Below the summary is a section for "添付ファイル" (Attachments) and a "更新履歴" (History) section showing two comments. The first comment, from admin, adds the assignee "admin". The second comment, also from admin, changes the assignee from "somebody" to "admin" and the status from "new" to "accepted". At the bottom, there is a "コメントの追加" (Add Comment) section with a text area and a "保存" (Save) button. The interface includes a sidebar with various icons and a top navigation bar with links like "チケットを見る", "admin", and "ヘルプ/ガイド".

チケットを見る

admin としてログイン中 ログアウト ヘルプ/ガイド Trac について 個人設定 more

Trac Lightning

SampleProject
My example project

チケット #1 (accepted タスク) 更新: 1

テストチケット 登録: 3分前 最終更新: 0秒前

報告者: テスト 担当者: admin

優先度: 通常 マイルストーン:

コンポーネント: その他 バージョン:

キーワード: admin

開始予定日:

終了予定日:

進捗率(%):

説明

複製 削除

▼ 添付ファイル

ファイルを添付します

▼ 更新履歴

92秒前 admin comment:1

- 関係者 admin を追加

返信 編集 削除

1秒前 admin comment:2

- 担当者 somebody から admin に変更
- ステータス new から accepted に変更

返信 編集 削除

コメントの追加 参照 ↑

B I A wysiwyg textarea WikiFormatting が使用可能

データ編集画面

4. 画面イメージサンプル(『OSS: BPM Suite』のサンプルイメージ)

The screenshot displays the Questetra BPM Suite web application. The browser address bar shows the URL: 192.1.1.138:18080/userweb/PE/Workitem/newList. The application has a green header bar with navigation tabs: マイページ, 全ユーザの処理記録, and プロセスモデルの管理. A user profile for 'miz さん' is visible on the left sidebar.

The main content area shows a process instance for 'テスト' (Test) with ID 'p7'. The status is '実行中' (Executing). The start date is 2011-01-20 17:10. Below this, a table shows the current task: '出版社の公開条件確認' (Check publication conditions) with status '処理中' (Processing) and assigned to 'miz (全社)'.

The process data section shows a task named 'このデータを再利用してプロセスを開始' (Start process using this data). Below this, a flowchart diagram illustrates the process flow:

```

graph LR
    Start((開始)) --> Task1[出版社の公開条件確認]
    Task1 -- 承認 --> Task2[公開の可否]
    Task1 -- 不承認 --> Task3[出版社連絡先調査]
    Task2 --> End1((終了1))
    Task3 --> End2((終了2))
  
```

The flowchart shows two parallel paths from the initial task. The top path involves checking publication conditions and then determining if publication is possible, leading to '終了1' (End 1). The bottom path involves investigating the publisher's contact information, leading to '終了2' (End 2).

At the bottom right, there are icons for document management and a language selector set to 'CAPS KANA'.

4. 画面イメージサンプル(『OSS: Redmine』のサンプルイメージ)

著作権処理状態管理システム

ホーム マイページ プロジェクト 管理 ヘルプ

ログイン中: miz 個人設定 ログアウト

検索: 著作権処理状態管理システム

概要 活動 チケット 新しいチケット ガントチャート カレンダー 文書 ファイル 設定

新しいチケット

トラッカー * 未着手

題名 *

親チケット

説明

B I U S C H1 H2 H3 表 表 pre 添付

テキストの書式: ヘルプ

ステータス * 新規

優先度 * 通常

担当者

本の名まえ *

開始日 2011-01-25

期日

予定工数 時間

書いたひと *

添付ファイル ファイルを選択 選択されていません 任意のコメント

別のファイルを追加 (最大サイズ: 5 MB)

チケットのウォッチャー ☐ r miz

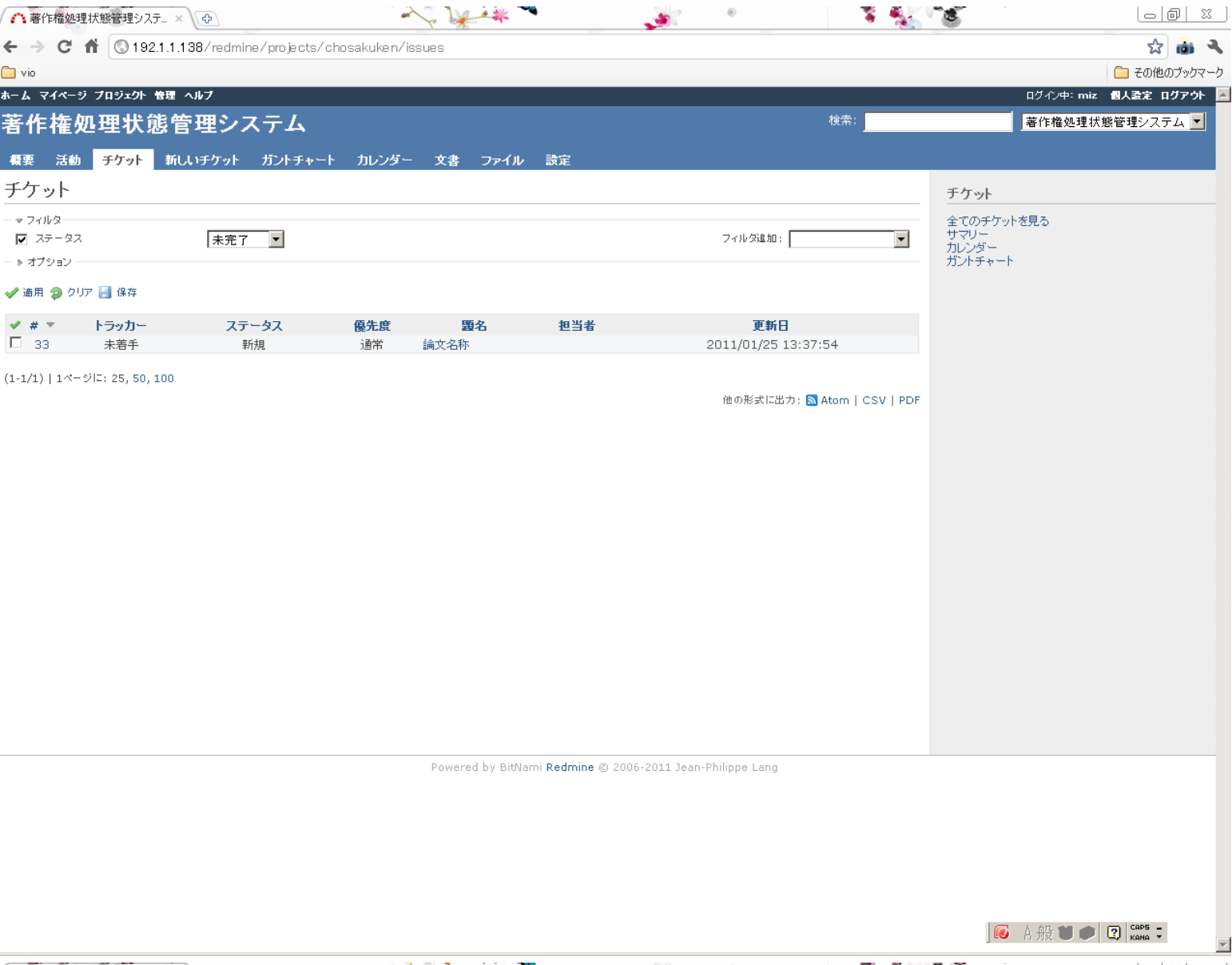
作成 連続作成 プレビュー

Powered by BitNami Redmine © 2006-2011 Jean-Philippe Lang

一般 CAPS KANA

登録画面

4. 画面イメージサンプル(『OSS: Redmine』のサンプルイメージ)



ステータス確認画面

4. 画面イメージサンプル(『OSS: Redmine』のサンプルイメージ)

データ編集画面

著作権処理状態管理システム

192.1.1.138/redmine/issues/33

vio

他のブックマーク

ホームマイページプロジェクト管理ヘルプ

ログイン中: miz 個人設定ログアウト

著作権処理状態管理システム

検索:

更新時間を記録ウォッチをやめる複製コピー移動削除

未着手 #33

論文名称

r miz が4日前に追加. 1分前に更新.

ステータス: 新規

優先度: 通常

担当者: -

カテゴリ: -

対象バージョン: -

本のなまえ: 論文名称

開始日: 2011/01/21

期日: -

進捗 %: 0%

作業時間の記録: -

書いたひと: 著者

子チケット追加

関連するチケット追加

履歴

r miz が4日前に更新 #1

• トラッカーを未着手から公開条件確認に変更

r miz が2分前に更新 #2

• 題名をほげから論文名称に変更

r miz が1分前に更新 #3

• トラッカーを公開条件確認から未着手に変更

• 本のなまえを論文名称にセット

• 書いたひとを著者にセット

更新

プロパティの変更 (続き)

トラッカー * 未着手

題名 * 論文名称

親チケット

説明 B I U S C H1 H2 H3 三 三 三 三 pre 三 三 三 三

テキストの書式: ヘルプ

全てのチケットを見るサマリーカレンダーガントチャート

チケットのウォッチャー (1) 追加

r miz

CAPS KANA