

---

たった1行のJavaScriptコードでRedmine UI  
(カスタムフィールドのトラッカー名の表示)を改善する方法

---

株式会社 **Ankoso**ft

2019年10月08日

## 目次

1. 現状の Redmine のカスタムフィールドのトラッカー名の表示 . . .	2
2. 改善した Redmine のカスタムフィールドのトラッカー名の表示 .	3
3. トラッカー名の表示改善の JavaScript コードの説明 . . . . .	4
4. おまけ . . . . .	8
株式会社 Ankosoft のご紹介 . . . . .	10
「ANKO ALM」のご紹介 . . . . .	11

### 注意

本書は Redmine の情報を提供する目的で作成されております。本書の内容を元に運用に影響がでる場合の責任は負いかねますので、ご了承ください。

本書の内容は 2019 年 10 月 08 日現在のものであり、内容は事前予告なく修正される場合があります。

## 1. 現状の Redmine のカスタムフィールドのトラッカー名の表示

下図のように、Redmine のカスタムフィールドにおけるトラッカー名の表示は、チェックボックス+ラベルの形式で表示されるため、トラッカー名が大量にある場合、どのトラッカー名にチェックが入っており、どのトラッカー名にチェックが入っていないのか、把握することが容易ではありませんでした。

特にラベルが途中で改行される場合もあるため、一目で正確なトラッカー名を把握することが難しく、似たようなトラッカー名が続く場合、確認に時間がかかりました。

また、チェックボックスが左、中、右とばらばらにあり、目視での確認に負担がかかりました。

一言で言って、「ごちゃごちゃして分かりづらい」という状況です。

The screenshot shows the Redmine web interface for configuring a custom field. The page title is 'カスタムフィールド >> チケット >> 日付'. The field type is set to '日付' (Date). The 'トラッカー' (Tracker) section is highlighted with a red box, showing a list of trackers with checkboxes for selection. The interface includes a search bar, navigation menu, and various configuration options.

トラッカー

- バグ
- 機能
- サポート
- 受付
- 申込書類の点検
- 申込内容の確認
- 申込内容の入力
- 申込完了の連絡
- 申込登録
- データの取り込み
- ファイリング
- 請求伝票入力
- 請求データ修正
- 未入金データ処理
- 請求伝票送付
- 請求書の確認
- 未入金データ問合せ
- 請求内容の承認
- 請求内容の申請
- 返金処理
- 入金処理
- 集計
- 営業データの取り込み
- 資料作成
- 分析

## 2. 改善したRedmineのカスタムフィールドのトラッカー名の表示

下図のように、チェックボックス+ラベルの形式で表示されているトラッカー名ごとに改行をいれて、チェックがオンになっているトラッカー名を赤字で表示することにより、何のトラッカー名にチェックが入っており、何のトラッカー名にチェックが入っていないのか、一目で把握することが可能になります。チェックボックスが左の同じ位置にあるため、可読性が向上し、追加でチェックをオン、オフにするとときに容易にチェックボックスを選択できます。

The screenshot shows the Redmine interface for configuring a custom field. The field name is '日付' (Date) and its format is '日付'. The 'トラッカー' (Tracker) list is highlighted with a red box, showing the following items with their respective checkboxes and colors:

- バグ
- 機能
- サポート
- 受付
- 申込書類の点検 (Red text)
- 申込内容の確認
- 申込内容の入力
- 申込完了の連絡
- 申込登録
- データの取り込み
- ファイリング
- 請求伝票入力 (Red text)
- 請求データ修正
- 未入金データ処理
- 請求伝票送付
- 請求書の確認
- 未入金データ問合せ
- 請求内容の承認 (Red text)
- 請求内容の申請
- 返金処理

### 3. トラッカー名の表示改善のJavaScriptコードの説明

```
$("#custom_field_tracker_ids").find("input[type='checkbox']").next().after("<br>").end().end().find("input[type='checkbox']:checked").next().css("color","red")
```

上記のJavaScript(JS)コードでトラッカー名の表示改善ができます。たった1行のJSコードで上記のようにUI表示を改善できます。

上記のJSコードは少し長いので下記のように2行に分けて書くことも可能です。

```
$("#custom_field_tracker_ids").find("input[type='checkbox']").next().after("<br>")
$("#custom_field_tracker_ids").find("input[type='checkbox']:checked").next().css("color","red")
```

簡単にこのJSコードをRedmineに適応する方法を説明します。

ブラウザでマウス右ボタンをクリックをして「検証」の「Console」に上記のコードを張り付けて、エンターキーを入力することで、上記のコードを実行できます。

下図は「Google Chrome」での図です。他のブラウザも画面や名称は異なりますが、同様の方法でJavaScriptコードを実行できます。

下図の青枠の部分に上記のコードを張り付けます。



たった 1 行の JavaScript コードで Redmine UI (カスタムフィールドのトラッカー名の表示) を改善する方法

[JavaScript コード実行前]

The screenshot shows the Redmine administration interface. On the left, the 'Tracker' section is expanded, showing a list of trackers with checkboxes. Some checkboxes are checked, including '申請書類の点検', '請求伝票入力', and '請求内容の承認'. A red box highlights this list. On the right, the browser's developer console is open, showing a single line of JavaScript code that has been executed. The code is: `$('#custom_field_tracker_ids').find("input[type='checkbox']").next().after("<br>").end().end().find("input[type='checkbox']:checked").next().css("color","red");`

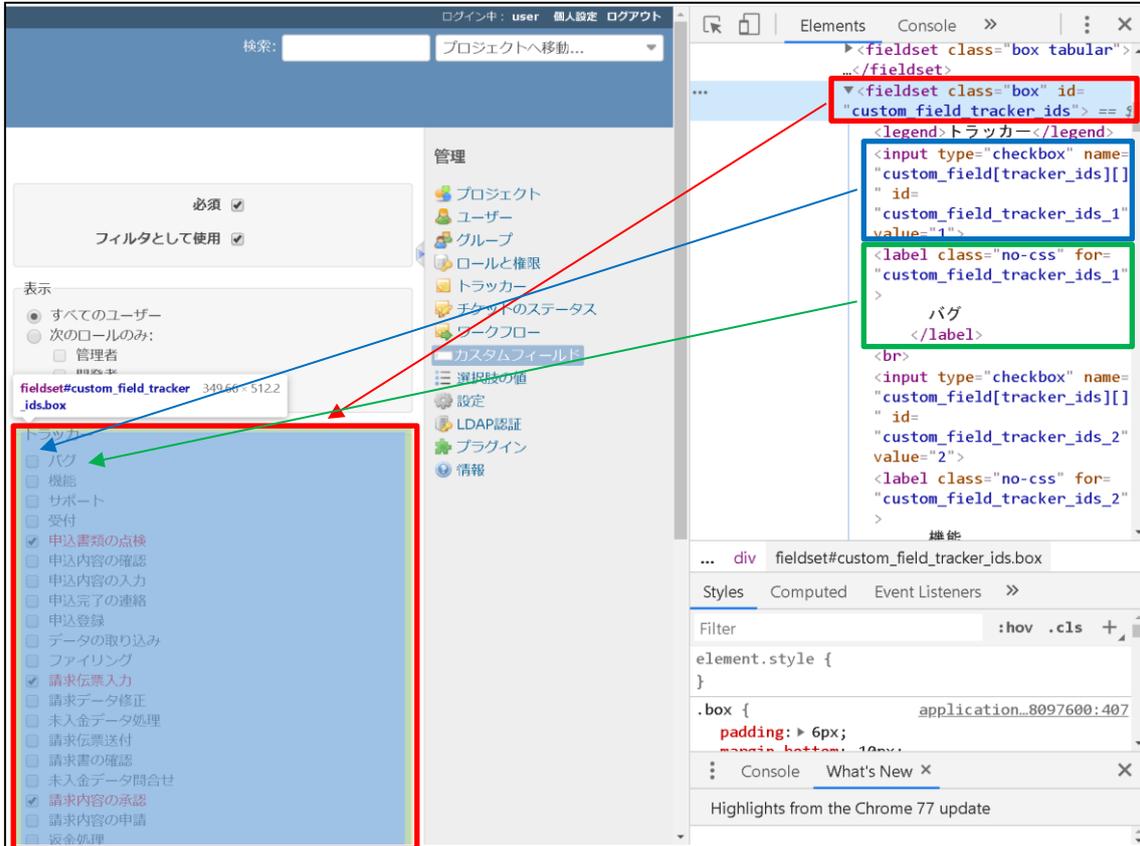
[JavaScript コード実行後]

The screenshot shows the same Redmine administration interface as before. A large blue arrow points from the 'Tracker' list in the previous screenshot to this one. In this screenshot, the checked items in the 'Tracker' list are now colored red. A speech bubble next to the list says 'UI が改善' (UI is improved). The JavaScript console on the right shows the same code as before, but now it has a return value: `(3) [Label.no-css, Label.no-css, Label.no-css, ...]`. The console also shows the execution of `m.fn.init` and the resulting object.

たった1行の JavaScript コードで Redmine UI (カスタムフィールドのトラッカー名の表示) を改善する方法

[html の分析]

該当の html は下記のような構造になっています。



```

<fieldset class="box" id="custom_field_tracker_ids">
  <legend>トラッカー</legend>
  <input type="checkbox" name="custom_field[tracker_ids][]" id="custom_field_tracker_ids_4" value="4">
  <label class="no-css" for="custom_field_tracker_ids_4">バグ</label>
  <input type="checkbox" name="custom_field[tracker_ids][]" id="custom_field_tracker_ids_5" value="5">
  <label class="no-css" for="custom_field_tracker_ids_5">機能</label>

```

赤字で示した部分が「トラッカー」の部分の大枠を構成し、その中に青字で示したチェックボックスと緑字で示したラベルが1つのセットになり、トラッカー名分繰り返り出現します。

該当の html と JS コードは文字色で区分したような関係になっています。

```

$("#custom_field_tracker_ids").find("input[type='checkbox']").next().after("<br>").end().end()
.find("input[type='checkbox']:checked").next().css("color", "red")

```

「赤字の要素の配下の各青字の要素の次の要素（つまり緑字の各要素）の後にオレンジの色の命令を実行します（1つ目は「<br>」（改行）を挿入する、2つ目は文字色を赤字にする」というコードになります）。

「end()」は「一つ前の要素セットに戻る」処理をしています。2度出てくるので2つ前の要素に戻っていません。これにより赤字の要素は青字の2つのfindの起点になります。

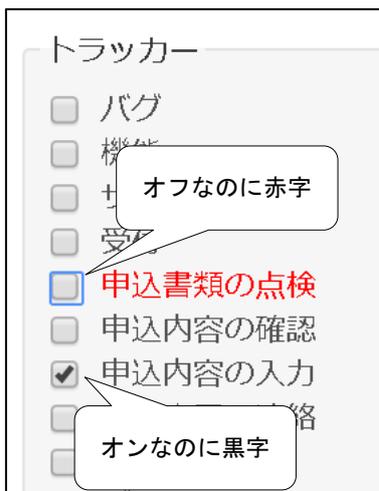
「検証」の「Console」でJSコードを実行するのは一時的な実行しかできずブラウザを更新するとJSコードが消えてしまうため、恒久的に該当のJSコードを実行するためには、「View Customize plugin」（<https://github.com/onozaty/redmine-view-customize>）をRedmineにインストールして、上記のJSコードを登録してください。技術サポートが必要な方は弊社（[sales@ankosoft.co.jp](mailto:sales@ankosoft.co.jp)）までご連絡ください。

## 4. おまけ

### [おまけ1]

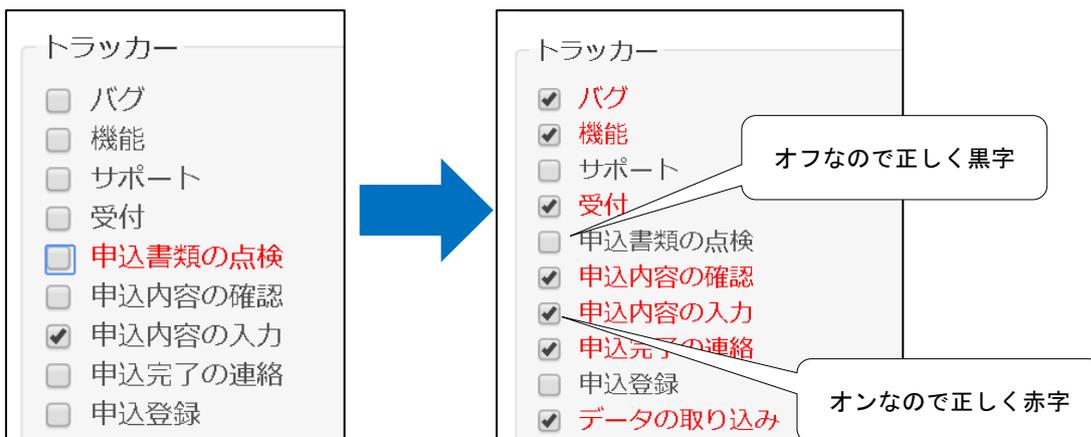
上記の処理は、上記の処理をしたその時の状態を元に処理をするため、上記の処理後にチェックボックスをオン、オフにしてもラベルの色は変わりません。

下図の例では「申込書類の点検」はオフになったのに赤字のままであり、「申込内容の入力」はオンになったのに黒字のままです。



これでは実運用上、利便性が半減してしまうので、チェックボックスのオン、オフの変更があるたびに赤字、黒字を変更させる必要があります。

JSコードは少し長くなりますが、下記のコードを追加で実行することにより動的な変更に対してもラベルの色を変更することが可能になります。



たった1行のJavaScriptコードでRedmine UI（カスタムフィールドのトラッカー名の表示）を改善する方法

```
//チェックボックスが変化したら発動
$("#custom_field_tracker_ids").find("input[type='checkbox']").change(function() {
  if ($(this).prop('checked')){
    $(this).next().css("color","red");//チェックが入れば文字色を赤にする
  } else {
    $(this).next().css("color","")
    //チェックがなければ文字色の指定を無くす
    //デフォルトの色である黒になる
  }
});
```

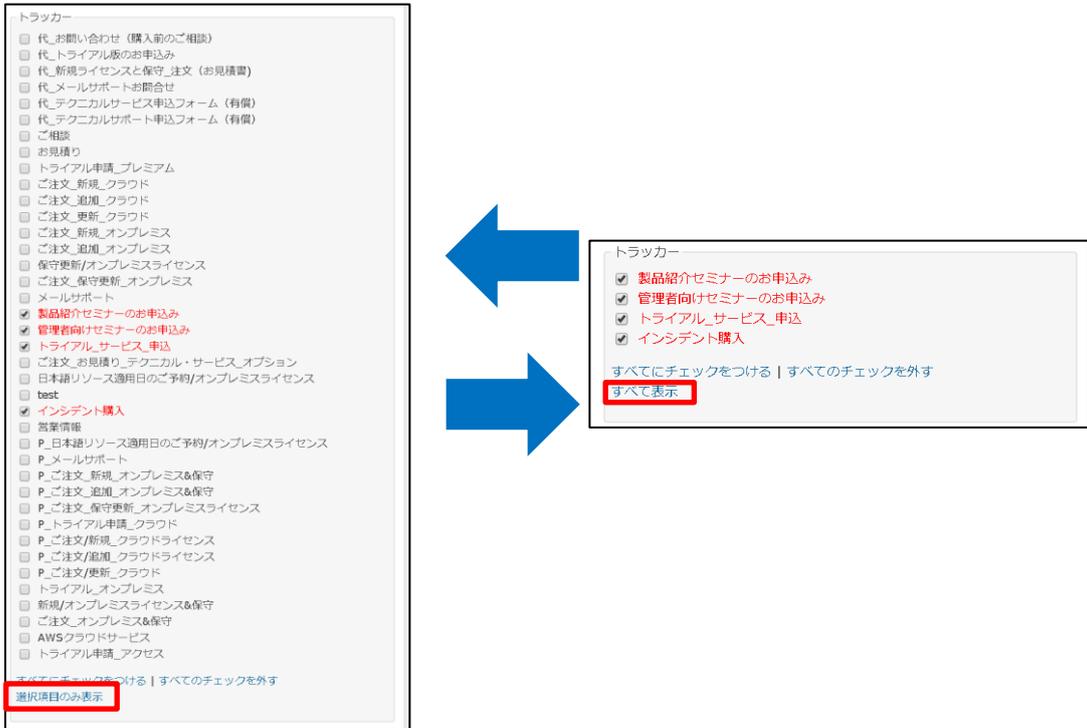
上記のように、オレンジ色の部分でチェックボックスの変動があると発動するように設定し、ifの部分でチェックボックスのオン、オフに応じてフォントの色を変える処理をしています。

[おまけ2]

応用するとチェックが入っているトラッカーだけを表示する機能、再度全て表示する機能を追加することができます。すでにチェックが入っているトラッカーだけが表示されるので、関連するトラッカーの確認や登録されているトラッカーの解除などを行う時に作業効率が向上します。

ご関心がある方は、弊社 (sales@ankosoft.co.jp) までご連絡ください。

[チェックが入っている項目のみ表示]



## 株式会社 Ankosoft のご紹介

### オープンソースソリューション専門企業 ANKOSOFT

Ankosoft はオープンソース開発を基盤にした会社で、プロジェクト管理と品質管理分野で世界的に有名な Redmine, Jenkins, SonarQube 等に対する技術力を保有しております。



オープンソースの不足した部分を開発して、ソリューションとしてパッケージ販売をしており、技術サポート、教育などオープンソースの普及に努めております。

本社 〒111-0041 東京都台東区元浅草 2-6-4 上野コアビル 3F

五反田事務所 〒141-0031 東京都品川区西五反田 8-8-16 五反田高砂ビル 204

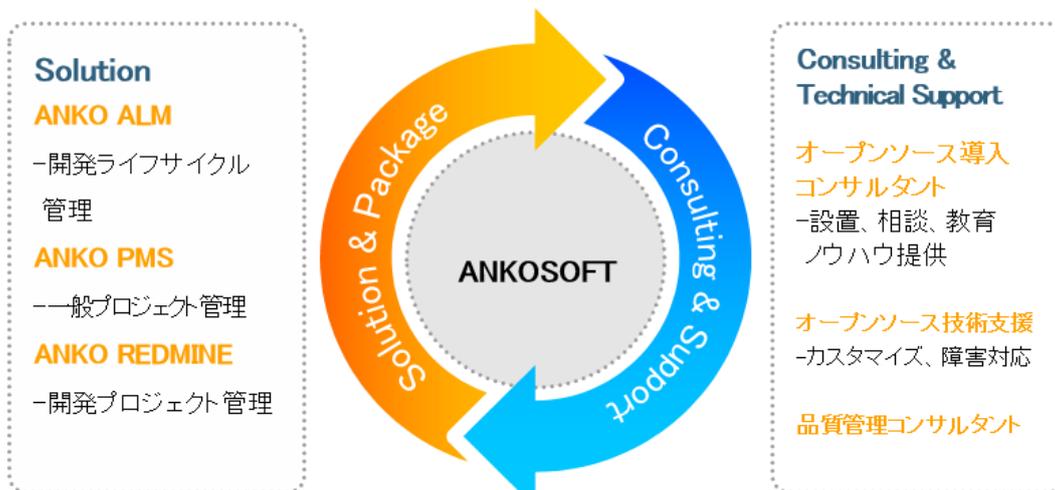
TEL : 050-5326-6639

EMAIL : sales@ankosoft.co.jp

URL : <http://ankosoft.co.jp/>

### 事業領域

Ankosoft の事業領域は、ソリューション開発および販売、オープンソースコンサルタント、技術支援、教育、品質管理コンサルタントなどを主な事業として行っております。



## 「ANKO ALM」のご紹介

オープンソースのプロジェクト管理システムである「REDMINE」を基盤にして、REDMINE に足りない機能を加えて開発した「ALM」システムです。REDMINE に足りなかった管理機能、利便性、モニタリング機能、ツール間の連動やプラグインのバグなどを改善・開発しました。特に、開発ライフサイクルにおけるツール間の連動は欠かせないものであるため、オープンソース間で連動できるよう開発された製品です。

「ALM」とは、アプリケーションソフトウェアの開発・運用において、全体のライフサイクルを総合的に管理することによって、ソフトウェアの品質や開発生産性、変化への対応力などを向上させるためのもの、もしくはそれらを実現させるためのソフトウェアのプラットフォームツールを指します。

## 「ANKO ALM」の特長

### ツール間の連動

開発、ビルド、テストなど、開発サイクルに必要なツールが互いに連動して自動化されている機能は、開発者に快適な環境を提供し、開発スピードをアップさせます。

### ソース品質の見える化

ANKO ALMの使用によって、開発プログラムソースのスケール感だけでなく、テストカバレッジ情報を自動的に分析して、把握することができ、今まで把握が難しかったプログラムの中まで確認できるようになり、より一層堅固で安定した製品開発をサポートします。

### オープンソースで構成

ANKO ALMはオープンソースで構成されており、世界中に厚いユーザー層がいて、最新技術を取り入れています。さらに、持続的にアップデートされていて、オープンソース間の互換性も備えています。また、多様なプラグインを提供しているため、必要な機能の追加や開発が可能です。

### 低費用で構築可能

オープンソースで構成されたANKO ALMはリーズナブルな費用で構築が可能です。商用ソリューションの場合、数千万円の製品が多く、一部機能のみ備えている製品でも数百万円に達するため、導入が簡単ではありません。一方、ANKO ALMは低費用で優れた機能を提供するので、一石二鳥の効果を期待できます。