



# Redmicaのチケット起票に ChatGPTを使ってみた

第24回redmine.tokyo勉強会

2023年 6月 3日

大和田 裕

# 自己紹介



名前: 大和田 裕



本名: 山嶋 裕



仕事:

- 日本タイムシェア株式会社(現:TIS)
- 日本IBM株式会社
- 日本チボリシステムズ株式会社(現:日本IBM)
- アットネットホーム株式会社(現:JCOM)
- 独立行政法人 情報処理推進機構 (IPA)
- 株式会社オリジネーション
- 株式会社オープンストリーム

と、ソフトウェア開発に身をおいていました。  
現在は、

# 自己紹介



名前: 大和田 裕



本名: 山嶋 裕



仕事: フリーランスのIT屋 (屋号:POfficeO)

ITアドバイザー/ソフトウェア開発

- 定量的プロジェクトマネジメント
- プロジェクト見える化
- ソフトウェアソリューション



## Redmineとのかかわり

- そもそもは、IPAでの【EPM-X】
- その後は、【EPM-Base】
- redmine.tokyoのスタッフ
- redmineエバンジェリストの会

その他:IPAの連携委員もやっています

# きっかけ

- ChatGPTについて、世間が騒がしいなー。



- redmine.tokyoや他のCommunityでも話題に。



- RedmineとChatGPTの組合せで、何ができるのか？



そうだ、ChatGPTに聞いてみよう！！

# 聞いてみました

Q: RedmineとChatGPTの連携を、5つあげてください。

A: RedmineとChatGPTの連携による具体的な活用例を5つご紹介します。

## 1. チケット作成支援:

ユーザーが新しいチケットを作成する際、ChatGPTを使ってチケットの詳細や要件を収集することができます。ユーザーが簡単な説明やキーワードを提供すると、ChatGPTが追加の質問をしたり、必要な情報を明確化したりすることで、正確かつ完全なチケットが作成されます。

## 2. タスク進捗の自動更新:

チームメンバーがタスクの進捗状況を更新する際に、ChatGPTを活用することで、手動の入力作業を削減できます。ChatGPTに進捗状況の更新を依頼すると、ChatGPTが必要な情報を問い合わせ、Redmineに自動的に進捗の更新を行います。

## 3. 自動アサイン:

新しいチケットが作成された際、ChatGPTを使用して最適な担当者を自動的にアサインすることができます。ChatGPTは、チケットの内容や優先度、チームメンバーのスキルセットなどを考慮して、最適な担当者を推薦することができます。これにより、チケットのアサインプロセスが迅速化され、タスクの効率的な処理が可能となります。

## 4. 自動チケット優先度の設定:

ChatGPTを使用して、新しいチケットの優先度を自動的に設定することができます。ChatGPTはチケットの内容や重要度を判断し、適切な優先度を割り当てます。これにより、チームメンバーが手動で優先度を設定する手間を省き、重要なタスクに対する対応が迅速になります。

## 5. チケット検索と解決策提供:

ユーザーがチケットを検索して解決策を見つける際、ChatGPTを活用することで、効率的なサポートを提供できます。ChatGPTはユーザーの問い合わせやキーワードを解釈し、Redmineのナレッジベースや過去のチケットから関連情報を検索して提供します。これにより、ユーザーが迅速に問題解決や必要な情報の入手ができます。

# ということで、作ってみることにした

## □ 「チケット作成支援」、「自動チケット優先度の設定」及び「自動アサインの一部」。

- 第一段階は、チケットの**新規起票**プログラム作成。(第二段階は修正??)

① ChatGPTに、Redmicaのチケット起票のための項目テンプレートを教える。

② ChatGPTに情報を入力し、テンプレートに従ってチケット項目を答えてもらう。

③ 追加・修正があれば、再度ChatGPTに知らせ、修正された答えをもらう。

④ ChatGPTからの答えを分解し、チケット・データを作成する。

⑤ Redmicaへチケット・データを送り、新規起票する。

# 作ってみました (1/3)

## □ 1回目のトライ

- まずは、①から③を作成し、ChatGPTの動作を確認。  
Python + Flet + OpenAI API



- プログラムは正常に動きましたが、失敗でした。  
①の段階で、入力トークン数が制限を超え、ChatGPTから警告メッセージが返ってきました。

参考:gpt-3.5-turboの制限

- 最大トークン長:4kトークン
- レート制限:1分間に40kトークン、1分間に200リクエスト

# 作ってみました (2/3)

- 2回目のトライ
  - ①から③を作成し、ChatGPTの動作を確認。
  - OpenAI APIをやめ、Seleniumでの画面操作へ。  
Python + Flet + Selenium + (Google Chrome)



- プログラムは正常に動きましたが、失敗でした。  
ChatGPTが  
**「私はロボットではありません」**  
画面となり、自動でも手動でも抜け出せません。



# 作ってみました (3/3)

## □ 3回目のトライ

- ①から③を作成し、ChatGPTの動作を確認。
- OpenAI APIをやめ、Seleniumでの画面操作へ。  
Python + Flet + Selenium + (Google Chrome)
- ChatGPTへのログインだけ手動に。



できたー！

## □ 完成

- ④、⑤のRedmicaチケット起票機能を組込む。  
Python + Flet + Selenium + (Google Chrome)  
+ Python-Redmine

**目標達成!**

かな？

# 開発環境

---

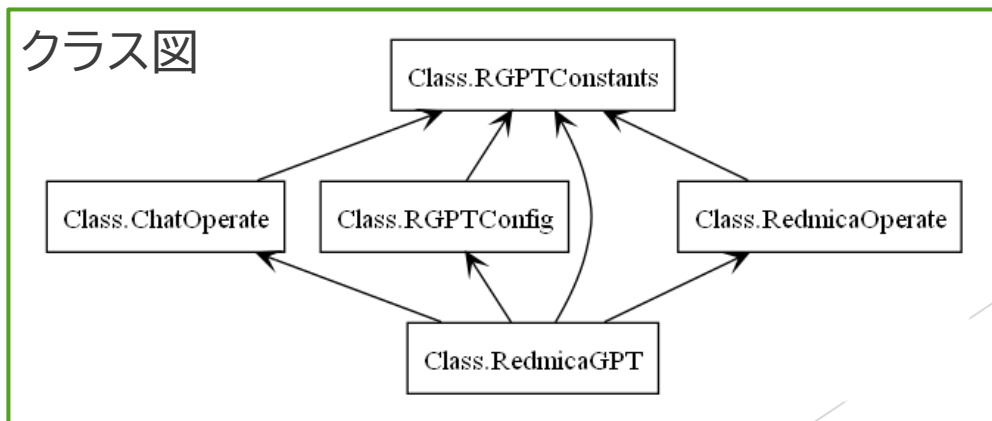
- OS: Windows 11
- Web Browser: Google Chrome
  - ポート指定: port=9222
  - URL指定: <https://chat.openai.com/auth/login>
- Redmca 1.2 (Stand Alone)
- 言語: Python 3.11
  - Flet, Selenium, Inscriptis, Python-Redmine
- IDE: PyCharm Community Edition

# プログラム構成

## プロジェクト・フォルダ

RGPTConstants.py	定数定義
RedmicaGPT.py	メイン・クラス、プログラム&画面制御
RGPTConfig.py	設定ファイル読込、実行環境の設定
ChatOperate.py	ChatGPT操作クラス
RedmicaOperate.py	Redmica操作クラス
config.ini	実行環境設定ファイル
Init_chat.template	ChatGPTに設定する項目テンプレート・ファイル
slack_map.csv	SlackIDとRedmicaIDとのマッピング・ファイル

## クラス図



実際に使ってみます

# 作ってみての感想

- OpenAI APIは制限があって使いづらい。
  - gpt-3.5-turbo
    - 最大トークン長: 4kトークン
    - レート制限: 1分間に40kトークン、1分間に200リクエスト
    - 入出力トークン同一価格で、1kトークンあたり\$0.002
  - gpt-4 (8kトークン版)、gpt-4-32k (32kトークン版)
    - 入力トークン: \$0.03 / 1kトークン
    - 出力トークン: \$0.06 / 1kトークン
  
- ChatGPTと画面連携するには、いろいろ課題が。
  - セキュリティの課題。
  - 応答が極端に遅くなる時がある。時間帯によっては接続できない。
  - 項目テンプレートの文言は、まだまだ改善の余地が。
  - 同じ質問でも、答えがまちまち。余計な言葉が入ったりする。
  - 同じ質問でも、返ってくるHTMLのtag構成が異なる。

ChatGPT Plusではどうだろう？

ご清聴ありがとうございました