

SlackBot プログラムの仕様書

2019/4/26

浜本 時空

1 はじめに

本資料は 2019 年度新人研修課題にて作成した SlackBot プログラムの仕様についてまとめたものである。本プログラムで使用する Slack[1] とは Web 上で利用できるチームコミュニケーションツールである。本プログラムは以下の 2 つの機能をもつ。

- (1) 指定された文字列を発言する機能
- (2) 指定された条件に合致するゲームに関する商品の検索

なお、本資料において発言とは Slack において文字列を投稿することを指す。また、本資料において発言内容は “” で囲って表す。

2 対象とする利用者

本プログラムは以下のアカウントを所有する利用者を対象としている。

- (1) Slack アカウント
- (2) 楽天 [2] アカウント

楽天アカウントは本プログラムで使用する楽天ブックスゲーム検索 API を使用するために必要である。

3 機能

本プログラムは Slack で “@hamabot” から始まるユーザの発言を受信し、それに対して発言する。発言内容は “@hamabot” に続く文字列により決定される。以下に本プログラムがもつ 2 つの機能を示す。

(機能 1) 指定された文字列を発言する機能

この機能はユーザの “@hamabot (任意の文字列) と言って” という発言に対して、(任意の文字列) を発言する。例えば、“@hamabot おはようと言って” に対しては “おはよう” と発言する。“@hamabot おはようと言ってと言って” のようにユーザの発言の中に “と言って” が複数回続く場合は、発言中で一番初めにある “と言って” より前の文字列を発言内容とする。この場合は “おはよう” と発言する。

(機能 2) 指定された条件に合致するゲームに関する商品の検索

この機能はユーザの “@hamabot (タイトル),(対応機種),(ソート順) で検索” という発言に対して、“で検索” より前の文字列のうちコンマで区切られた (タイトル),(対応機種),(ソート順) で

表 1 表示する商品情報

商品情報名	内容
タイトル	タイトルの名称 (2 文字以上)
対応機種	対応機種の名称
発売日	発売された年月日
価格	単位は円
評価平均	最大 5 点満点
評価件数	評価の件数
楽天ブックス URL	楽天ブックスの商品ページの URL

楽天ブックス内のゲームに関する商品を検索，ソートし，購入が可能な商品のうち上位 3 件の商品情報を表示する．表示する商品情報を表 1 に示す．たとえば，“@hamabot オセロ,PS1, 人気で検索” に対しては，「オセロ」が含まれるタイトルで対応機種が「PS1」のゲームを「人気」というソート順でソートし，上位 3 件の商品を表示する．

楽天ブックスにおいて，商品のソート順には以下の 7 つの種類がある．

- (1) 標準
- (2) 売れている
- (3) 発売日 (古い)
- (4) 発売日 (新しい)
- (5) 価格が安い
- (6) 価格が高い
- (7) レビューの件数が多い
- (8) レビューの評価 (平均) が高い

本プログラムでは (2)～(6) の 5 つのソート順を選ぶことができ，それぞれ人気，古い，新しい，安い，および高い，と入力することでソート順を指定できる．これらの文字列が入力された場合は，楽天ブックスでの「標準」というソート順でソートされる．

上記の (機能 1) と (機能 2) のどちらにも当てはまるユーザの発言を受信したときは，(機能 1) が優先される．一方，(機能 1) と (機能 2) のどちらにも当てはまらないユーザの発言を受信したときは，以下のメッセージを発言する．

表 2 動作環境

項目	内容
OS	Ubuntu 18.04.2 LTS(Bionic Beaver)
メモリ	512MB
CPU	Intel(R) Xeon(R) CPU E5-2670 v2 @ 2.50GHz
Ruby	2.5.3p105
Gem	bundler 1.16.1
	mustermann 1.0.2
	rack 2.0.4
	rack-protection 2.0.1
	sinatra 2.0.1
	tilt 2.0.8

こんにちは！私は hamabot です．私には 2 つの機能が備わっています．

機能 (1)

「 」と言って」と入力すると「 」と発言します．

機能 (2)

楽天ブックスで販売しているゲームに関する商品を検索し，上位 3 件を表示します．

< 検索フォーマット => @hamabot [タイトル],[機種],[ソート] で検索 >

| [タイトル]...2 文字以上で入力してください．

| [機種]...Switch, DS, PS3 など，機種名をアルファベットと数字で入力してください．

| [ソート]... 人気, 古い, 新しい, 安い, 高い のうちから一つ選ぶことができます．(それ以外の場合は標準のソートを行います．)

なお，販売していない商品は検索の対象外となっております．

4 動作環境

本プログラムは Heroku[3] 上で動作させることを想定している．Heroku とは，PaaS (Platform as a Service) と呼ばれる形態のサービスで，アプリケーションを実行するためのプラットフォームである．

5 動作確認済み環境

本プログラムは表 2 の環境で動作確認済みである．

6 環境構築

6.1 概要

本プログラムの動作のために必要な環境構築の項目を以下に示す．

- (1) Slack の Incoming Webhook の設定
- (2) Heroku アカountの作成と設定
- (3) Slack の Outgoing Webhook の設定
- (4) 楽天ブックスゲーム検索 API のアプリ ID の取得

6.2 Slack の Incoming Webhook の設定

Slack が提供している Incoming Webhook の設定をおこなう．

- (1) Slack にログインし，画面左上にある自分の名前をクリックし，「Slack のカスタマイズ」を選択する．
- (2) 画面左側の「アカウント」欄にある「App 管理」を選択する．
- (3) 画面左側の「管理」欄にある「カスタムインテグレーション」を選択する．
- (4) 「認証済みカスタムインテグレーション」から「Incoming Webhook」を選択する．
- (5) 「設定を追加」をクリックし，送信するチャンネルを選択し追加する．
- (6) Webhook URL から URL をコピーして設定を保存する．ここでコピーした URL は 6.3 節の手順 (7) で使用する．

6.3 Heroku アカountの作成と設定

- (1) 以下の URL から Heroku にアクセスし，「Sign up」から新しいアカウントを登録する．

`https://www.heroku.com/`

- (2) 登録したアカウントでログインし，「Getting Started on Heroku」の使用する言語として Ruby を選択する．
- (3) 「I'm ready to start」をクリックし，「Download Heroku CLI for...」から CLI をダウンロードする．
- (4) 以下のコマンドを実行し，Heroku にログインする．

```
$ heroku login
```

- (5) 作成したプログラムが存在するディレクトリに移動する．以下に，アプリケーションを~/myapp 以下で作成した場合のコマンド例を示す．

```
$ cd ~/myapp
```

- (6) 以下のコマンドを実行し，Heroku 上にアプリケーションを生成する．

```
$ heroku create <myapp_name>
```

上記のコマンドのうち，<myapp_name>は任意のアプリケーション名を設定する．ここで設定したアプリケーション名は，6.4 節の手順 (9) にて必要になるため，控えておく．

- (7) 以下のコマンドを実行し，Heroku の環境変数に Incoming Webhook の URL を追加する．

```
$heroku config:set INCOMING_WEBHOOK_URL="<incoming_webhook_url>"
```

上記のコマンドのうち，<incoming_webhook_url>は 6.2 節の手順 (6) にて入手する URL である．

- (8) settings.yml に Incoming Webhook の URL を記入する．記入例を以下に示す．

```
incoming_webhook_url: <incoming_webhook_url>
```

6.4 Slack の Outgoing Webhook の設定

Slack が提供している Outgoing Webhook の設定をおこなう．

- (1) Slack にログインし，画面左上にある自分の名前をクリックし，「Slack のカスタマイズ」を選択する．
- (2) 画面左側の「アカウント」欄にある「App 管理」を選択する．
- (3) 画面左側の「管理」欄にある「カスタムインテグレーション」を選択する．
- (4) 「認証済みカスタムインテグレーション」から「Outgoing Webhook」を選択する．
- (5) 「設定を追加」をクリックし「Outgoing Webhook インテグレーション」の追加をクリックする．
- (6) 以下の 3 つの項目を設定する．
- (7) 「チャンネル」にて，発言を監視するチャンネルを選択する．
- (8) 「引き金となる言葉」に，Webhook が動作する契機となる単語を設定する．
- (9) 「URL」に，Webhook が動作した際に POST を行う URL を設定する．今回は Heroku 上で動作させるため，以下の URL を設定する．

```
https://<myapp_name>.herokuapp.com/slack
```

上記の URL のうち，<myapp_name>は 6.3 節の手順 (6) で設定したアプリケーション名である．

6.5 楽天ブックスゲーム検索 API のアプリ ID の取得

今回使用した API は楽天ブックスゲーム検索 API である．本 API を使用するための，アプリ ID の取得方法について以下に示す．

- (1) 以下の URL にアクセスし，「アプリ ID 発行」をクリックする．

```
https://webservice.rakuten.co.jp/
```

- (2) ログインし、アプリ新規作成フォームのアプリ名とアプリ URL 欄を記入し、新規アプリを作成する。
- (3) 「アプリ ID/デベロッパー ID」欄のアプリ ID をコピーし、以下のコマンドを実行して Heroku の環境変数にアプリ ID を追加する。

```
$heroku config:set INCOMING_WEBHOOK_URL="<アプリ ID>"
```

上記のうち、<アプリ ID>がコピーしたアプリ ID である。

- (4) settings.yml にアプリ ID を記入する。記入例を以下に示す。

```
applicationId: <アプリ ID>
```

7 使用方法

本プログラムは Heroku 上で動作するため、Heroku へデプロイすることで実行できる。Heroku には以下のコマンドを実行してデプロイできる。

```
$ git push heroku master
```

8 エラー処理

- (1) (機能 2) について、検索したが購入できる商品が存在しなかった場合には以下のメッセージを発言する。

購入できる商品はありませんでした。

9 保障しない動作

本プログラムの保証しない動作を以下に示す。

- (1) 楽天ブックスゲーム検索 API のサービス停止時における本プログラムの実行
- (2) 4 章に示した動作環境以外の環境での本プログラムの実行
- (3) Slack の Outgoing Webhook 以外からの POST リクエストをブロックする動作

10 おわりに

本資料では 2019 年度新人研修課題にて作成した SlackBot プログラムの仕様についてまとめた。

参考文献

- [1] Slack: その仕事、Slack で。 , Slack (オンライン) , 入手先 <https://slack.com/intl/ja-jp/> (参照 2019-04-25)
- [2] 楽天株式会社 : 楽天ウェブサービス - 楽天市場 , 楽天株式会社 (オンライン) , 入手先 <https://webservice.rakuten.co.jp/> (参照 2019-04-25)
- [3] Heroku: Heroku: クラウド・アプリケーション・プラットフォーム , Heroku (オンライン) , 入手先 <https://jp.heroku.com/home> (参照 2019-04-25)