Book Bot Project

Group 7

October 27, 2017

Abstract

This file is a project report about Group 7's Book Bot Project for CmpE451 class. Information in this file is up to Milestone 1.

1 Project Description

Book Bot Project is an assistant that can help users with everything related to books. It is a bot, and it works via Telegram. Working via Telegram makes the bot more accessible; users don't need to install anything extra and it is cross platform.

The base and most important functionality is to communicate with natural language. The bot can carry on a conversation with daily English.

Users can ask the bot about books. They can search books with different keywords, genres or authors, they can filter them in any way they want and they can sort them. They also can get information about specific books, see its current popularity via ratings or read the comments about the book.

Users can also give the bot feedback about the books they have read. They can comment on a book or rate a book. This will help the bot learn more about what the user likes and help it work more user oriented

The bot can also recommend the user some books depending on their previous ratings and taste of books.

1.1 Project Requirements

We have defined our project requirements as follows and our customer has agreed and approved the requirements. Note that the following is a summary of the detailed requirements page. You can see the full list from project requirements in our wiki page.

1. Functional Requirements
   (a) System Requirements
      i. Telegram will be the medium for the bot.
      ii. Goodreads or Google Books will be our source of book related information.
      iii. Wit.ai will be used for NLP purposes to understand natural language.
      iv. There will be regular users, admins and moderators.
      v. There will be a conversation tree which is going to be used to determine the conversation direction.
   (b) User Requirements
      i. We will have regular users as Telegram Users.
         A. Users will be able to get information; search for books, filter them, sort them and read comments about them
         B. Users will be able to give feedback about books; comment on them and rate them.

        C. Users will be able to get book recommendations.

    ii. We will have admin users.

        A. Admins will be able to modify the conversation tree

        B. Admins will be able to see and flag comments

        C. Admins will be able to see and block users

    iii. We will have moderator users.

        A. Moderators will be able to see and flag comments

        B. Moderators will be able to see and block users

2. Non Functional Requirements

    (a) Speed: We are creating a bot which users can chat with so the response time cannot be more than 5 seconds.

    (b) Size: We want our bot to be portable an not bulky, we will use the Telegram as our medium hence the user won't need extra space for the bot.

    (c) Usage: It should be easy to use, there shouldn't be any training time.

    (d) Requirements: Bot needs internet connection to work.

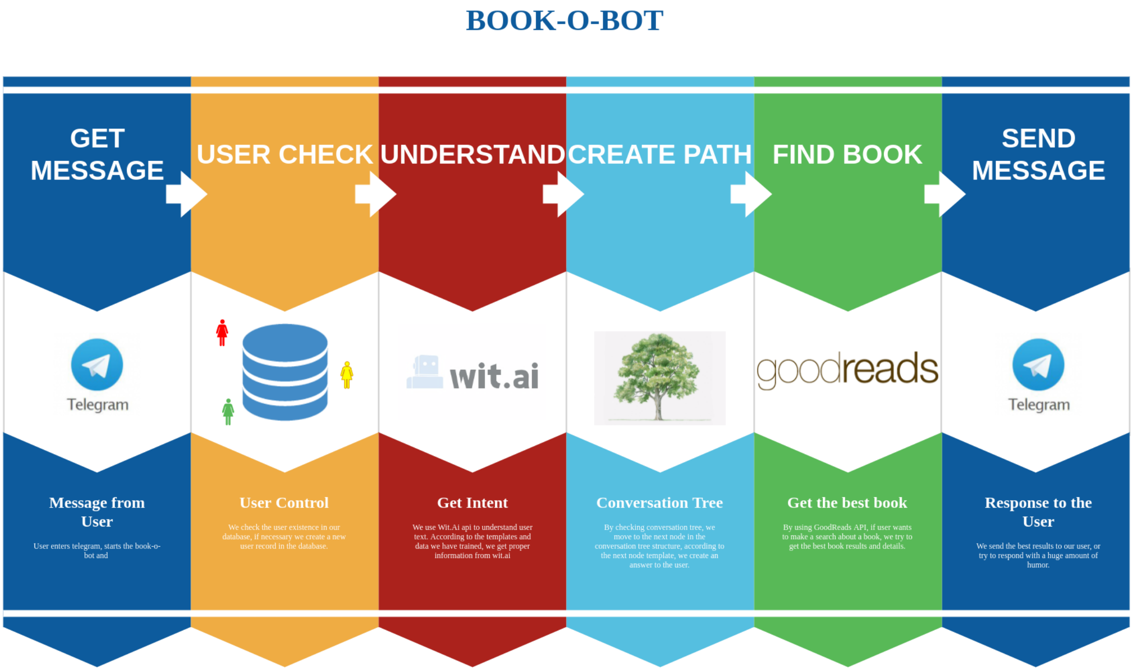
    (e) Failures: If bot fails, it shouldn't take more than ten minutes to restart the system and get it working again.

    (f) Target devices: We are targeting all platforms in which Telegram works.

    (g) User Satisfaction: Bot should keep track of the success of its recommendations by tracking the ratings of the recommended books.

1.2 Project Design

See Figure 1



BOOK-O-BOT

GET MESSAGE

USER CHECK

UNDERSTAND

CREATE PATH

FIND BOOK

SEND MESSAGE

Message from User

User enters telegram, starts the book-o-bot and

User Control

We check the user existence in our database, if necessary we create a new user record in the database.

Get Intent

We use Wit.Ai api to understand user text. According to the templates and data we have trained, we get proper information from wit.ai

Conversation Tree

By checking conversation tree, we move to the next node in the conversation tree structure, according to the next node template, we create an answer to the user.

Get the best book

By using GoodReads API, if user wants to make a search about a book, we try to get the best book results and details.

Response to the User

We send the best results to our user, or try to respond with a huge amount of humor.

Figure 1: Project Design

Row Name	Row Name
Column Name	column value
Column Name	column value

Table 1: An project plan table.

Row Name	Row Name
Column Name	column value
Column Name	column value

Table 2: An deliverable status table.

1.3 Project plan

TODO: write project plan. It is going to be in tabular form. See Table 1

2 Project Milestones

2.1 Milestone 1

TODO: Milestone 1

2.2 Milestone 2

TODO: Milestone 2

2.3 Final Milestone

TODO: Final Milestone

3 Project status

3.1 Deliverable List

1. One
   - (a) One point one: Explanation
   - (b) One point to: What is this deliverable
2. Two.

3.2 Deliverable Status

TODO: write deliverable status. It is going to be in tabular form. See Table 2

3.3 Deliverable Evaluation

TODO: Evaluate

4 Coding Work

You can find each team member's contribution to the code in the Table 3

Name	Coding Work
Salih Sevgican	GoodReads API implementation book API class for ease of use Main Page of the project (Front End) AWS has been configured for the deployment Updated requirements.txt GoodReads compatibility with TelegramBot tested on telegramapi
Irmak Kavasoglu	Initial environment setup for Django Setting up database model for Telegram User Setting up database model for Templates Setting up database model for Conversation Node Integrating Mptt tool for conversation tree Registering models to admin page and customizing their look
Add your name	add your contribution

Table 3: Coding work table.

5 Evaluation of tools and managing the project

5.1 Django

5.2 Mptt

5.3 Telegram

5.4 Wit

5.5 GoodReads

GoodReads API is very useful to make a search about a book by given any kind of keyword. Keyword can be related to author, genre or title, doesn't matter, this api makes a succesfull search every time. We've used a wrapper library for GoodReads API which made easy to write methods for search types. Library is a litlle bit complicated despite it is written to ease our job. Unfortunately not every book is being hold as a same type in goodreads database, some of queries returns arrays with additional information of book such as order number of a book in the book series, and some of queries returns just the name of book. This is an issue we need to handle very well in order to make a great impression with the responds to the bot user.

5.6 AWS

6 Summary

TODO fill