Chat-bot-team-20

Video presentations

Link for the first live demo: https://drive.google.com/file/d/1r-Z9tOQUw9xpVBAXJCbN5R_DBIHX0QFv/view
Link for the second live demo: https://drive.google.com/file/d/1_DKAy2UgbUeMMk6meNT-RrQ6qOt9WOhY/view
Link for the individual demo: https://drive.google.com/file/d/1KmlmBcBLjYD77Y2lDeO2dYTfjgiib2Q1/view

Github URL: https://github.com/shiro102/Individual_320_chatbot

Contents:

1/ Intro To The Project (which also can be read in the Github's IndividualReport.md)
2/ The Software Development Life Cycles (which also can be read in the Github's IndividualReport.md or SDLC, WBS and Gantt chart.md)
3/ New APIs (which also can be read in the Github's List of new API.md)
4/ Credits

1/Intro to the project

Describe your topic/interest (context of the chatbot, who will use it, etc.)

In this project, we developed an interactive conversational agent that responds to user input. In response to the user, the agent generates sentences as output. There are 2 types of target users. The first type of target users includes anime and manga lovers who would love to talk about them and know more about it. The second type of target users includes anyone who are interested in Japanese culture. There are 2 types of topics as well. The first type of topic of the conversation is about the general information and personal preference of anime. The second type of topic of the conversation is about the general information of Japan such as people, religion, food, samurai and so on.

About Our Project:

Our team chose incremental development as our SDLC model, generated a list of tasks based on each phase of the model, allocated time and manpower to each task based on duration and schedule with a WBS and a Gantt chart as explanation. Then, our team started planning for development. The role of the agent is Japan and anime expert and users are people who are interested in Japanese culture and anime and would love to talk and know more about Japanese culture and anime. The topics of the conversation are the general information and personal preferences of anime and the information of Japan such as people, religion, food, samurai and so on. The GitHub page of our project is this: https://github.com/shiro102/Chat-bot-team-20

How the code works

How to run it:

To compile the code, we run in terminal these 2 lines of codes - “python train.py” and “python chatbot.py”. The first code is to train the model so that the GUI might function properlyl. The second code is to run the app. Further instructions are in the README file in the CODE folder.

Stages of development:

There are five stages of the development for the code: data importing and loading, data preprocessing, data training and testing, model building, and GUI developing.

How the classes are organized:

There are 12 classes used in the code: nltk, json, pickle, numpy, keras, and tkinter, Wikipedia, Stanford Corenlp, Sentimental Analyser, GUI, Home, and Recent.

- Class “nltk” contains a group of libraries which provide statistical processing for English Language and is commonly used for Natural Language Processing. It is used throughout all the developing stages except the model building stage and the GUI development stage.
- There are 4 critical methods within this class: “nltk.stem.wordnetlemmatizer”, “nltk.word_tokenize”, “nltk.pos_tag” (instead of Stanfordnlp’s POS tagging to simplify the implementation) and “nltk.corpus”

  ```
    1. The first method, “nltk.stem.wordnetlemmatizer”, converts a word into
  its lemma form, groups different words to be analyzed as a single item based
  on similar meaning, and then creates a pickle file to store the Python
  objects which we will use while predicting.

    2. The second method, “nltk.word_tokenize”, is used to cleanup and break
  the whole text into small parts, such as words.

    3. The third method, “nltk.pos_tag”, tags every word as “Proper Nouns”,
  “Verb”, “Adjectives” etc. It is used for one of the new features we added -
  POS tagging - and it works in the similar pattern as Stanford Corenlp’s POS
  tagging. We will explain Stanford Corenlp later.

    4. The fourth method, “nltk.corpus”, is used to access “wordnet” which
  helps us to implement a new feature that we added to this code - synonym
  recognition.
  ```

- Class “json” is the data file which predicts the user inputs and gives responses. It is used for importing and loading data, preprocessing data, and getting random responses for the GUI. Json is also used implicitly throughout the program as the fundamental data in chatbot - conversation patterns.
- Class “pickle” is to make the data operations more efficient by removing object hierarchy when dumping our data or when loading our data from the dataset as it converts/treats the data as a single stream. Pickle is used throughout the stages except the data importing stage and the GUI development

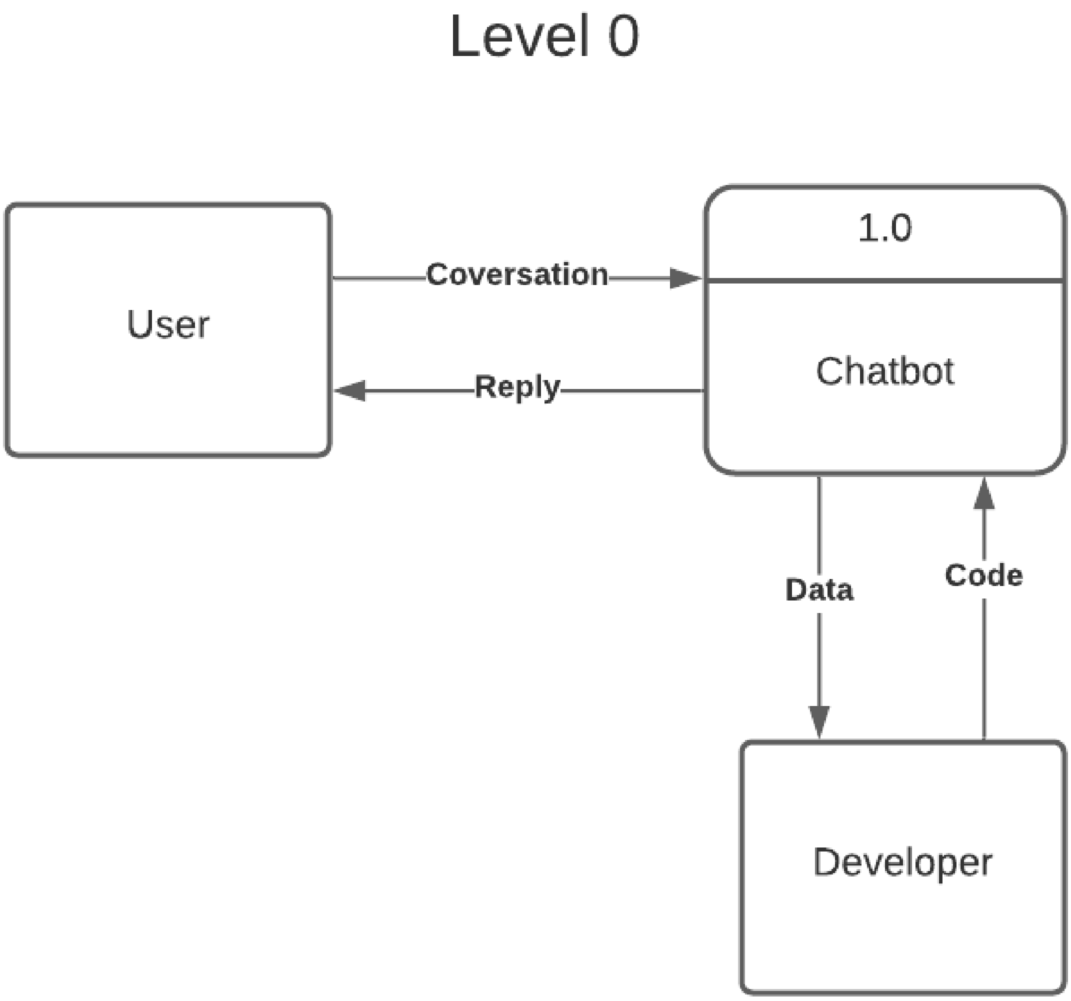
stage. It was also used to save a model for one of the new features that we added to this code - sentiment analysis.

- Class “numpy” is to increase the efficiency of the operation of lists in python. It is used in 2 stages - “creating data for training and testing” and “predicting classes for GUI”.
- Class “keras'' is to build and import the deep neural network model for the trained data. It is used in the stages of building and importing the model to GUI.
- Class “tkinter” is used to develop a graphical user interface by powerful libraries and functions within the class. It is used to develop the final GUI.
- Class “Wikipedia” is used to initiate online searches on Wikipedia in real time. It is an additional feature for our chatbot that functions when the chatbot doesn’t recognize a “Proper Noun” that is found by one of the new features we added - POS tagging.
- Class “Stanford Corenlp” is a service for natural language processing. Instead of creating a wrapper ourselves, we used a wrapper for this class called stanfordcorenlp. The link is put under the reference list at the end of the README file.
- “Sentimental Analyser” is a different python file created to help us with Sentiment Analysis. We create a naïve bayes model to decide between “Negative” and “Positive” Sentences. This model is used to analyze how users react to it.
- “GUI” is a class that is used to initialize everything related to our Graphical User Interface such as images, pages, etc.
- Class “Home” is the class for our home page in the Graphical User Interface and it contains all features present on that page.
- Class “Recent”, is used to record and store recent conversation dialogue in our recent conversation page in the Graphical User Interface. It contains all features present on that page.

Data Flow Diagrams (DFD)

- Level 0:

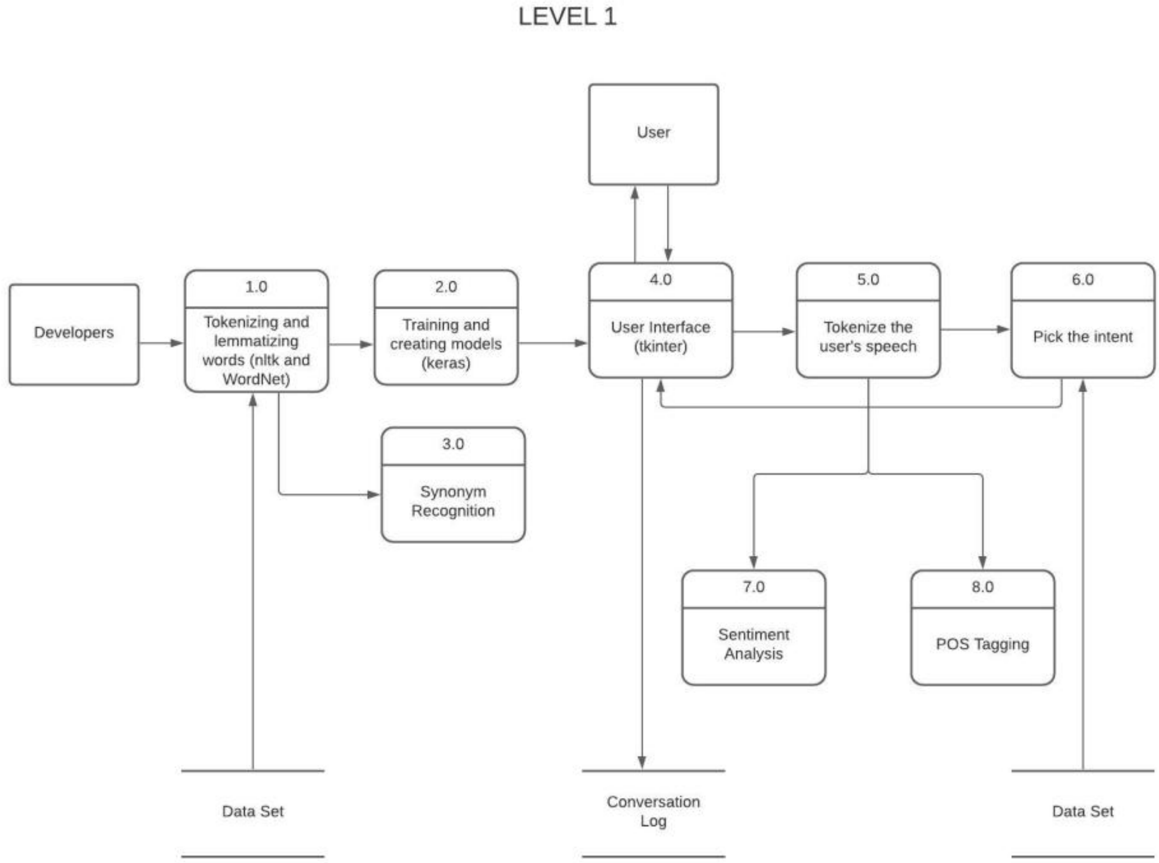
  - Image:



  - Explain: This is our level 0 DFD, as you can see we have two entities namely, the user that is using the chatbot and the developers, which would be our entire team. The way the developers interact with the chatbot is by implementing new features fixing any bugs etc.

- Level 1:

- Image:



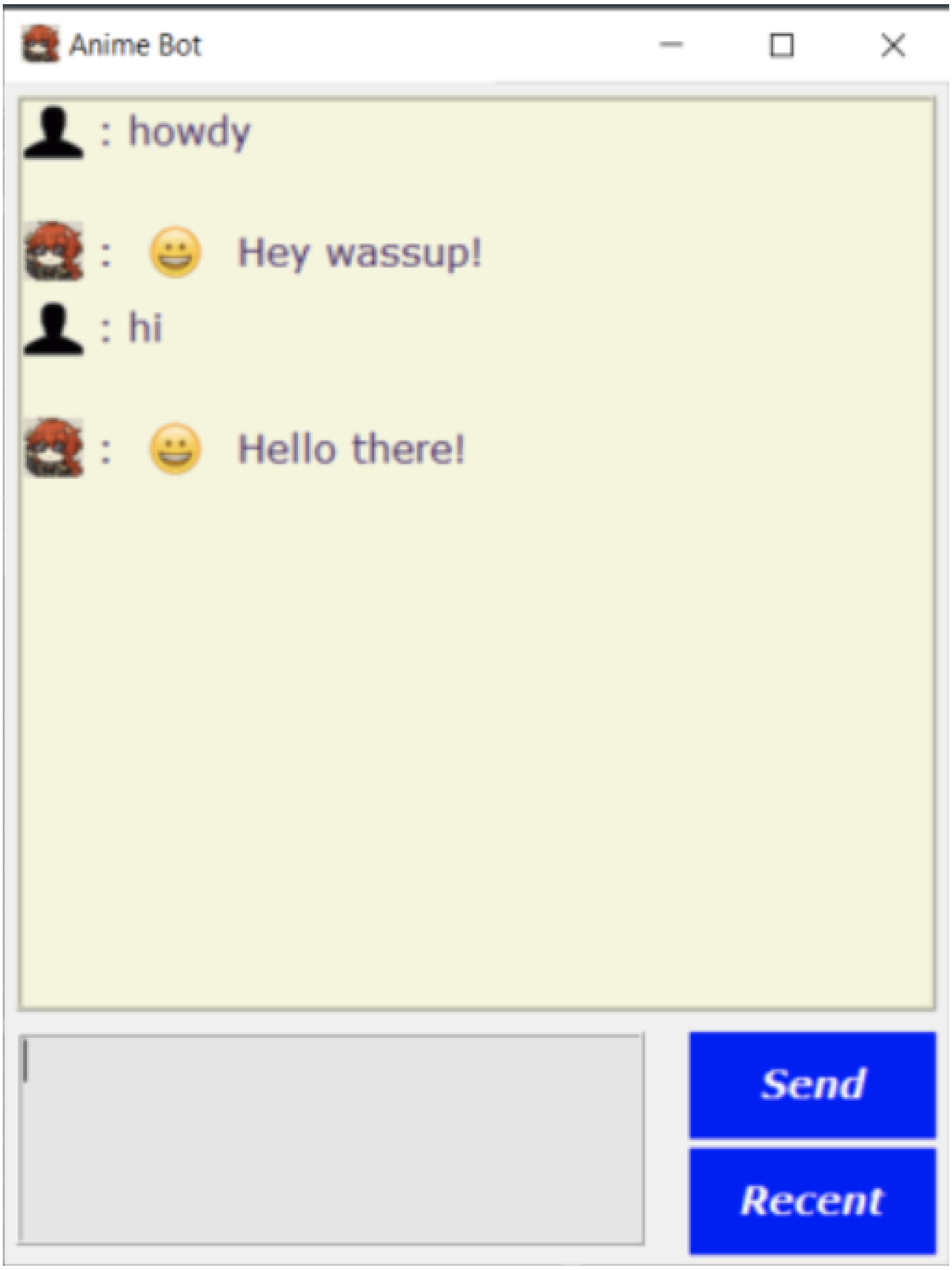
- Explain: This right here is our level 1 DFD. Like the level 0 DFD we still have our developer and user as our entities. We have our synonym recognition, POS tagging and sentiment analysis as our processes. Our synonym recognition process works on the intents that is already in the dataset. Whereas the POS and sentimental works when only when the user has typed something on the UI and the bot prepares its response by picking required response from the dataset which is symbolized by the datastore at the bottom. We have another datastore that stores the conversation log named “conversation log”, the option to store this conversation comes from the UI.

A List of 5 features that can be shared to others as API

- POS Tagging: includes the ability to searching on wikipedia and give responses based on that.
- Python file "SentimentalAnalyzer": a file that implements sentiment analysis
- Our chat bot: an application for others to view and modify.
- The method "remove_noise": removes all unnecessary words from a sentence
- Our Graphical User Interface

Features:

- Synonym recognition
    - Function: It identifies synonyms within sentences and give corresponding answers. It allows users to make inputs more diversified and give correct answers at the same time
    - Snippet:



: howdy

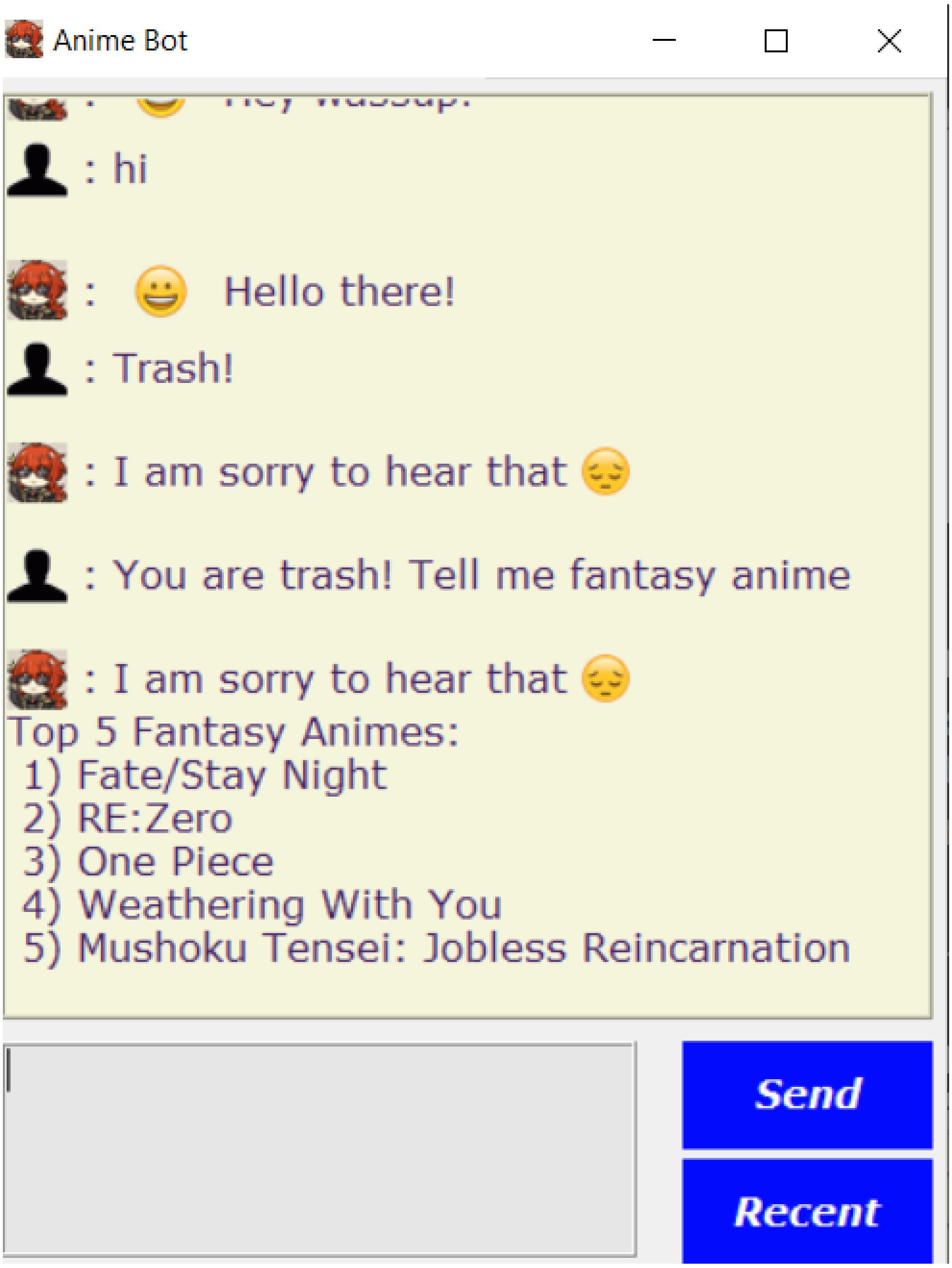
: 😀 Hey wassup!

: hi

: 😀 Hello there!

- Sentiment analysis
    - Function: It recognizes user input that contains positive, negative, or neutral emotions and give corresponding answers without us having to code. It makes dialogue turns more lively and realistic.

  - Snippet:



: 😀 Hey wassup!

: hi

: 😀 Hello there!

: Trash!

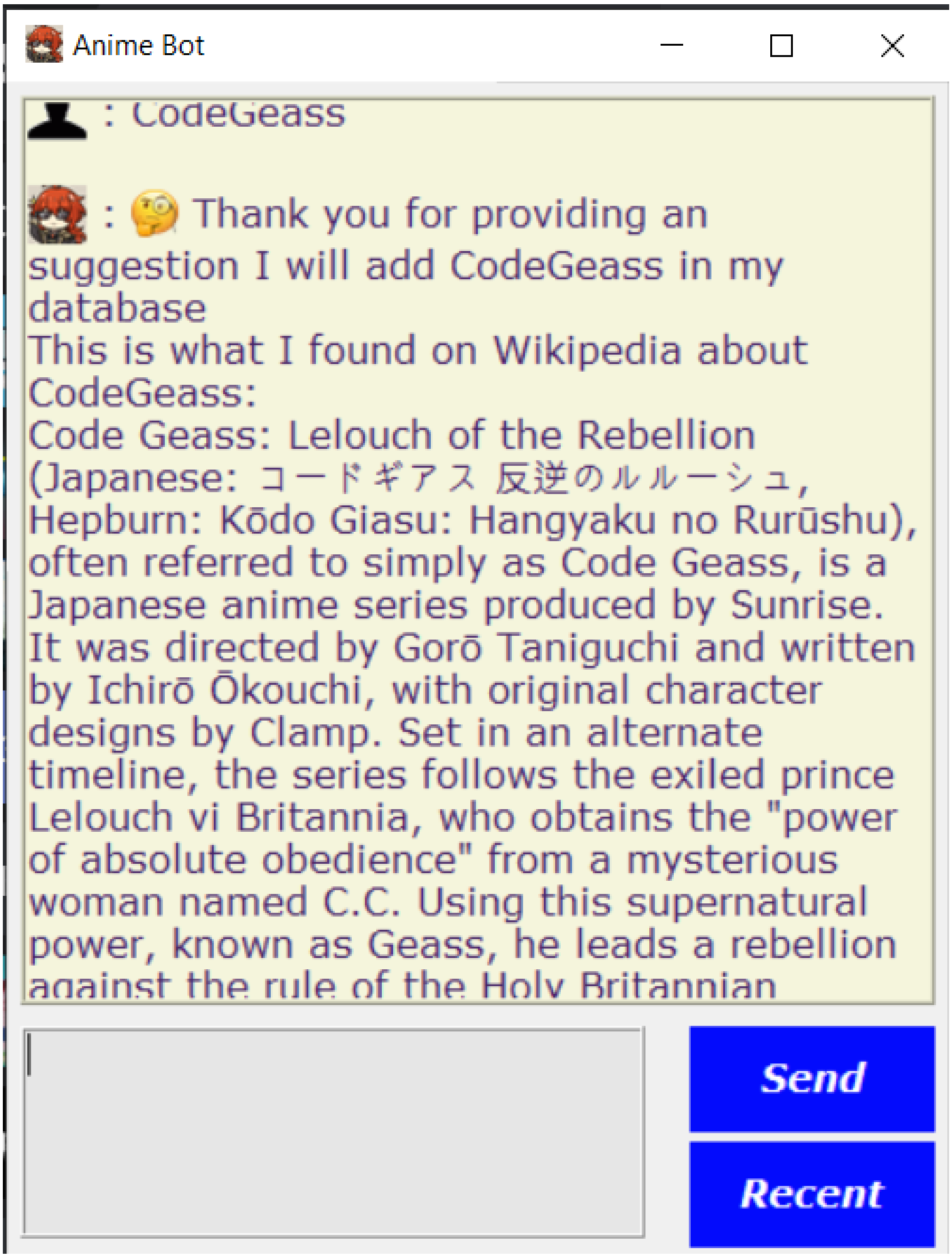
: I am sorry to hear that 😔

: You are trash! Tell me fantasy anime

: I am sorry to hear that 😔
Top 5 Fantasy Animes:
 1) Fate/Stay Night
 2) RE:Zero
 3) One Piece
 4) Weathering With You
 5) Mushoku Tensei: Jobless Reincarnation

- POS tagging
  - Function: It gets the information about Proper Nouns that our bot doesn't know and searching them real time on wikipedia. With this feature, our chatbot is able to answer topics that are outside of our designed topic and the users might get more satisfaction throughout conversations.

  - Snippet:

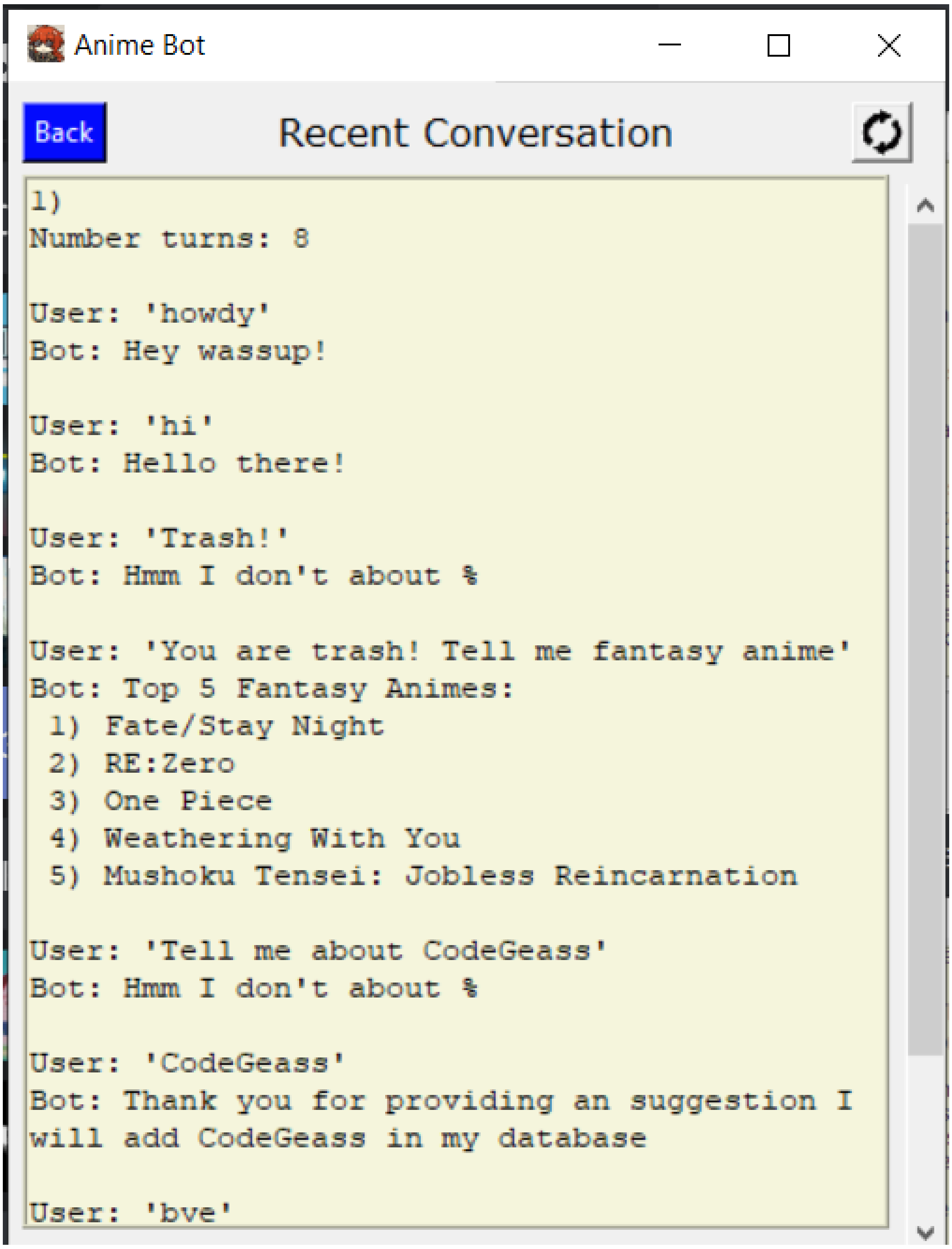


: CodeGeass

: 🧐 Thank you for providing an suggestion I will add CodeGeass in my database
This is what I found on Wikipedia about CodeGeass:
Code Geass: Lelouch of the Rebellion (Japanese: コードギアス 反逆のルルーシュ, Hepburn: Kōdo Giasu: Hangyaku no Rurūshu), often referred to simply as Code Geass, is a Japanese anime series produced by Sunrise. It was directed by Gorō Taniguchi and written by Ichirō Ōkouchi, with original character designs by Clamp. Set in an alternate timeline, the series follows the exiled prince Lelouch vi Britannia, who obtains the "power of absolute obedience" from a mysterious woman named C.C. Using this supernatural power, known as Geass, he leads a rebellion against the rule of the Holy Britannian

- Recent Conversation Page
  - Function: It saves dialogue information everytime our users say "bye" or close the app. With this feature, our users are able to check the dialogue history which is convenient.

  - Snippet:



```
1)
Number turns: 8

User: 'howdy'
Bot: Hey wassup!

User: 'hi'
Bot: Hello there!

User: 'Trash!'
Bot: Hmm I don't about %

User: 'You are trash! Tell me fantasy anime'
Bot: Top 5 Fantasy Animes:
 1) Fate/Stay Night
 2) RE:Zero
 3) One Piece
 4) Weathering With You
 5) Mushoku Tensei: Jobless Reincarnation

User: 'Tell me about CodeGeass'
Bot: Hmm I don't about %

User: 'CodeGeass'
Bot: Thank you for providing an suggestion I
will add CodeGeass in my database

User: 'bye'
```

2/The Software Development Life Cycles:

Our Choice: Incremental Development

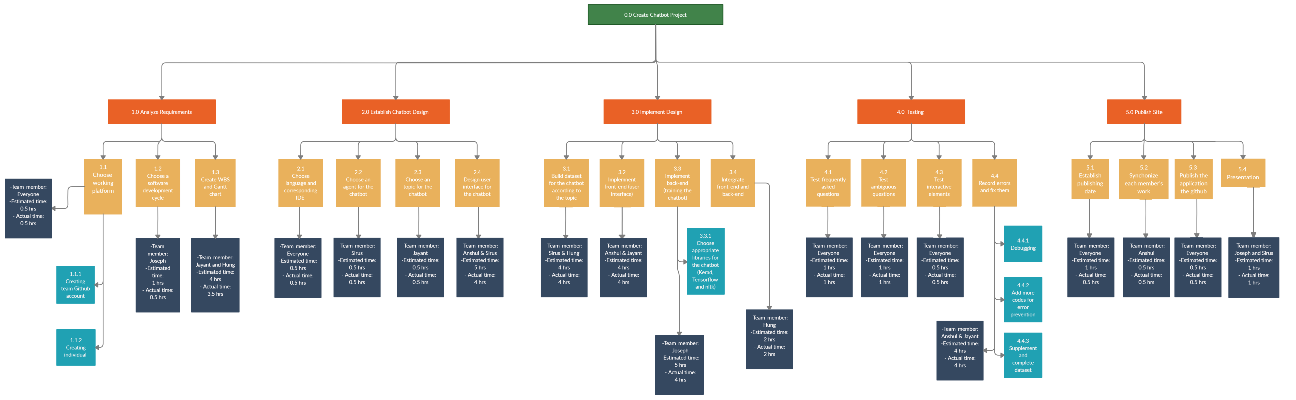
Rationale:

We choosed Incremental Development as our software development life cycle. Since more assignments about this project are expected to be published, some changes to this project might be required. As a result, we think that a model that is able to accommodate changes is most suitable for this project. Incremental development requires multiple versions of the project to be published frequently, so the model allows executable products to be generated earlier, and further modification can be done in later versions. In this way, modification is encouraged and the possibility of a complete failure of the product is reduced. The restriction of this model is that to rectify a task done in previous version requires modifications in all units and is time-consuming.

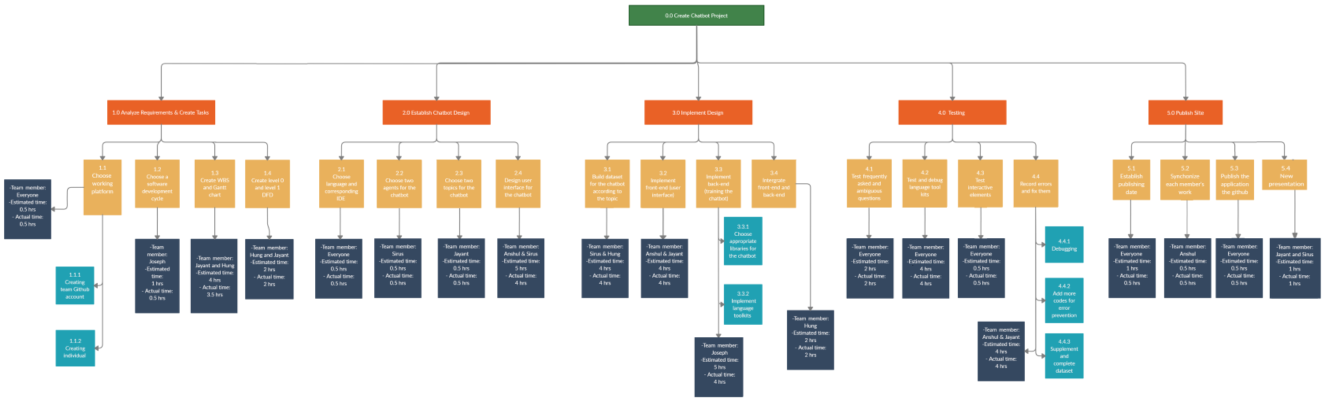
Process:

```
 1) Analyze requirements and Create Tasks:
    i) Choosing a platform to publish the application
        a) Creating github account for each team member
        b) Creating  team github account
    ii) Choose a software development cycle
    iii) Creating WBS and Gantt charts.
    iv) Create Level-0 and Level-1 Data Flow Diagrams (DFDs)

2) Establish design:
    i) Choose a language and its corresponding IDE
    ii) Choose two agents for the chatbot.
    iii) Choose two topic for the chatbot
    iv) Design the user interface for the chatbot

3) Implementing design:
    i) Build the dataset for the chatbot
    ii) Implement the user interface.
    iii) Implement back-end (Training the chatbot)
        a) Choose libraries
        b) Implement language toolkits
    iv) Integrate the front end and back end.

4) Testing:
    i) Testing frequently asked questions and ambiguous questions.
    ii) Testing and debugging language toolkits.
    iii) Testing the interactive elements
    iv) Record errors and fix them.
        a) Debugging
        b) Add code for error prevention
        c) Supplement and complete the dataset

5) Publishing application:
    i) Establish publishing date
    ii) Synchronize each member’s work
    iii) Publish the application to GitHub
    iv) New Presentation Video
```

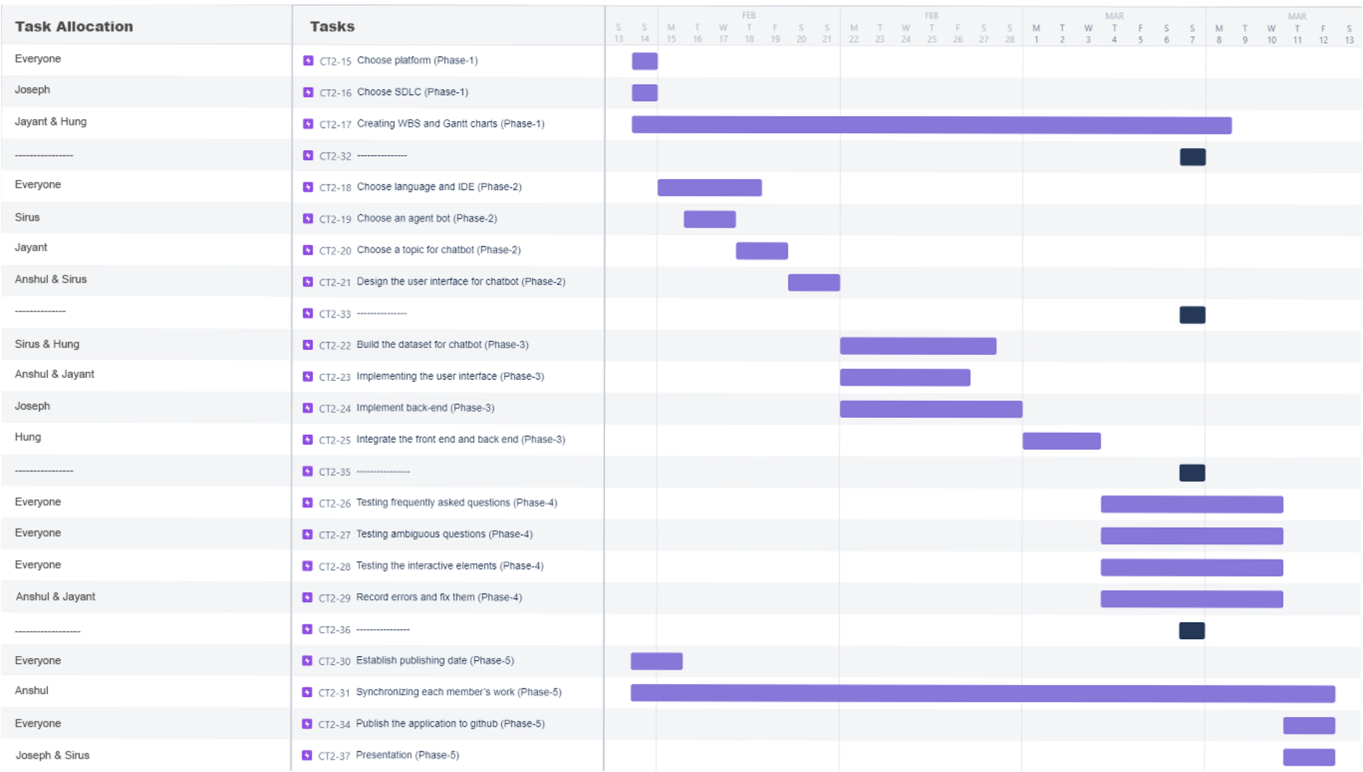
Old Work Breakdown Structure (WBS):



New Work Breakdown Structure (WBS):

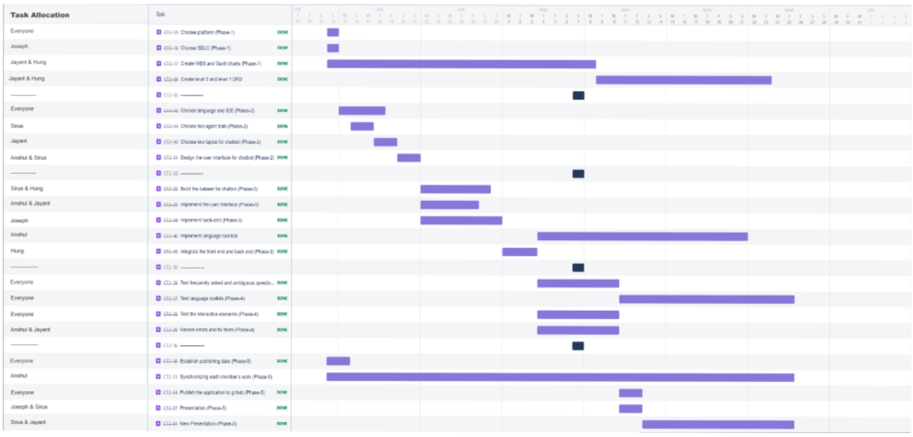

0.0 Create Chatbot Project
1.0 Analyze Requirements & Create Tasks
2.0 Establish Chatbot Design
3.0 Implement Design
4.0 Testing
5.0 Publish Site

Old Gantt chart:



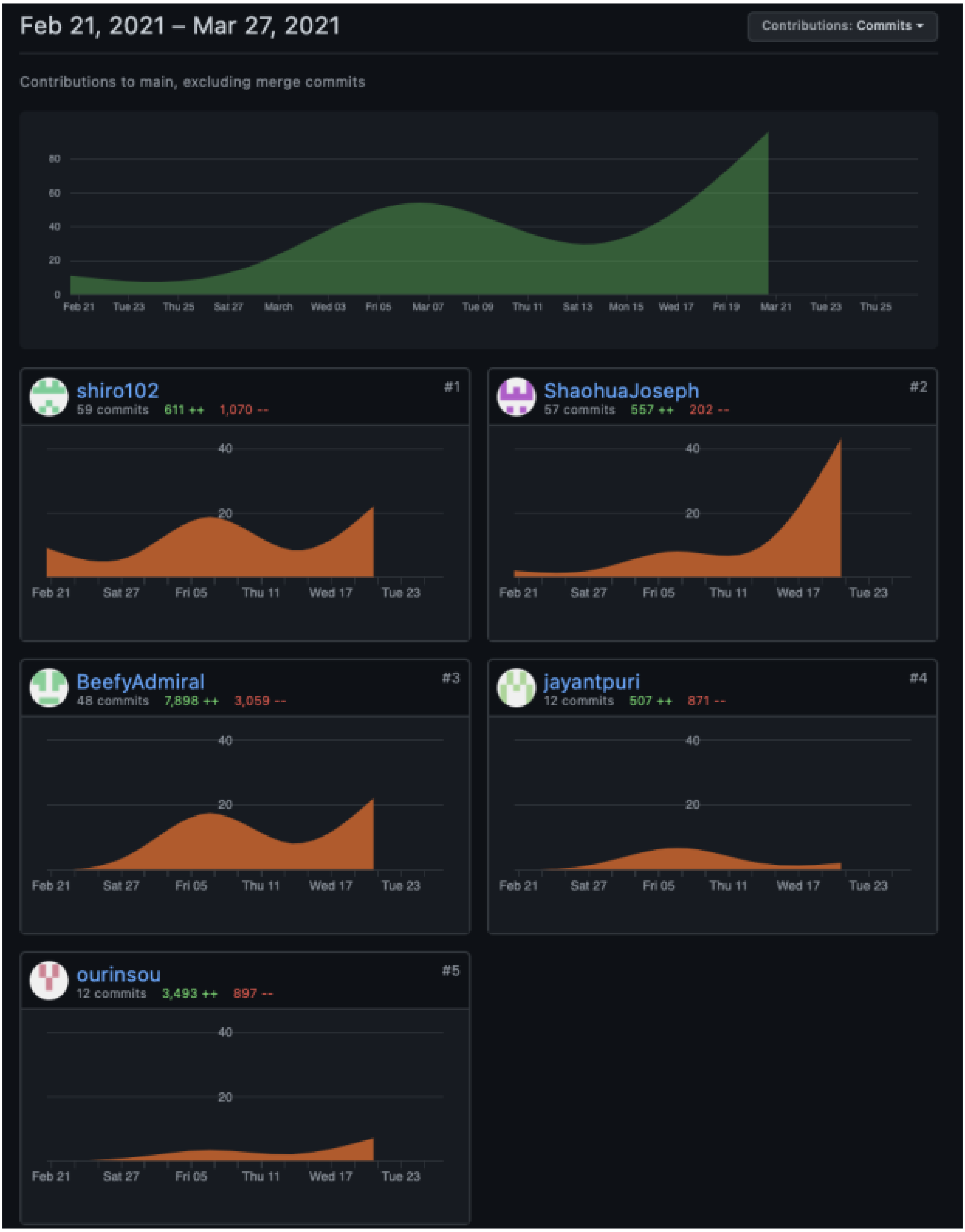
Task Allocation	Tasks
Everyone	CT2-15 Choose platform (Phase-1)
Joseph	CT2-16 Choose SDLC (Phase-1)
Jayant & Hung	CT2-17 Creating WBS and Gantt charts (Phase-1)
---------------	CT2-32 -------------
Everyone	CT2-18 Choose language and IDE (Phase-2)
Sirus	CT2-19 Choose an agent bot (Phase-2)
Jayant	CT2-20 Choose a topic for chatbot (Phase-2)
Anshul & Sirus	CT2-21 Design the user interface for chatbot (Phase-2)
-------------	CT2-33 -------------
Sirus & Hung	CT2-22 Build the dataset for chatbot (Phase-3)
Anshul & Jayant	CT2-23 Implementing the user interface (Phase-3)
Joseph	CT2-24 Implement back-end (Phase-3)
Hung	CT2-25 Integrate the front end and back end (Phase-3)
---------------	CT2-35 --------------
Everyone	CT2-26 Testing frequently asked questions (Phase-4)
Everyone	CT2-27 Testing ambiguous questions (Phase-4)
Everyone	CT2-28 Testing the interactive elements (Phase-4)
Anshul & Jayant	CT2-29 Record errors and fix them (Phase-4)
----------------	CT2-36 --------------
Everyone	CT2-30 Establish publishing date (Phase-5)
Anshul	CT2-31 Synchronizing each member's work (Phase-5)
Everyone	CT2-34 Publish the application to github (Phase-5)
Joseph & Sirus	CT2-37 Presentation (Phase-5)

New Gantt chart:



Task Allocation	Epic	
Everyone	CT2-15 Choose platform (Phase-1)	DONE
Joseph	CT2-16 Choose SDLC (Phase-1)	DONE
Jayant & Hung	CT2-17 Create WBS and Gantt charts (Phase-1)	DONE
Jayant & Hung	CT2-39 Create level 0 and level 1 DFD	DONE
------------	CT2-32 ------------	
Everyone	CT2-18 Choose language and IDE (Phase-2)	DONE
Sirus	CT2-19 Choose two agent bots (Phase-2)	DONE
Jayant	CT2-20 Choose two topics for chatbot (Phase-2)	DONE
Anshul & Sirus	CT2-21 Design the user interface for chatbot (Phase-2)	DONE
------------	CT2-33 ------------	
Sirus & Hung	CT2-22 Build the dataset for chatbot (Phase-3)	DONE
Anshul & Jayant	CT2-23 Implement the user interface (Phase-3)	DONE
Joseph	CT2-24 Implement back-end (Phase-3)	DONE
Anshul	CT2-40 Implement language tool kits	DONE
Hung	CT2-25 Integrate the front end and back end (Phase-3)	DONE
------------	CT2-35 ------------	
Everyone	CT2-26 Test frequently asked and ambiguous questio...	DONE
Everyone	CT2-27 Test language toolkits (Phase-4)	DONE
Everyone	CT2-28 Test the interactive elements (Phase-4)	DONE
Anshul & Jayant	CT2-29 Record errors and fix them (Phase-4)	DONE
------------	CT2-36 ------------	
Everyone	CT2-30 Establish publishing date (Phase-5)	DONE
Anshul	CT2-31 Synchronizing each member's work (Phase-5)	
Everyone	CT2-34 Publish the application to github (Phase-5)	DONE
Joseph & Sirus	CT2-37 Presentation (Phase-5)	DONE
Sirus & Jayant	CT2-44 New Presentation (Phase-5)	DONE

Insight of the contributors to this project



Feb 21, 2021 – Mar 27, 2021

Contributions: Commits

Contributions to main, excluding merge commits

shiro102 #1
59 commits 611 ++ 1,070 --

ShaohuaJoseph #2
57 commits 557 ++ 202 --

BeefyAdmiral #3
48 commits 7,898 ++ 3,059 --

jayantpuri #4
12 commits 507 ++ 871 --

ourinsou #5
12 commits 3,493 ++ 897 --

Limitations of the program submitted:

- The biggest limitation on the chatbot is that we don't have a dataset that is large enough. We don't have an enterprise-level development team and the dataset would be very limited and would only be applicable to certain situations.
- The chatbot cannot really "understand" what the user is actually saying since overall, it's a running algorithm.
- All responses are randomized, and no certain output is determined.

- There is no self-learning function implemented so the chatbot would not understand, or learn the user’s behavior, answers, patterns etc.
- Fuzzy search is not supported so misspelling would lead to totally different answers.
- The dialogues are not interconnected, and the chatbot would not make connections between them.
- Assume the user only enters valid (no spelling errors) and “understandable” (within the scope of dataset) or otherwise the chatbot would reply with a response saying it doesn't understand your questions and would try to search an unknown Proper noun on wikipedia.

Sample Outputs:

Possible Questions that you can ask from the chatbot:

1. Good day
2. What can you do?
3. Any anime to watch?
4. Top action anime?
5. What is the best adventure anime?
6. Any top-rated drama anime?
7. Best fantasy anime to recommend?
8. What are the top-rated comedy anime?
9. What is your favorite anime?
10. Tell me about Konosuba!
11. Who is your favorite character?
12. What anime are you watching recently?
13. Do you know about manga?
14. Top-rated manga?
15. What is your favorite manga?
16. Tell me about Beserk!
17. Who is your favorite Beserk character
18. Manga that you read recently
19. Recommend me manga
20. Do you know about Manhwa?
21. What are famous Manhwa?
22. What is Otaku culture?
23. What is anime convention?
24. What are famous conventions?
25. What convention do you want to vist?
26. Tell me about Jump Festa!
27. Tell me about Video games
28. Tell me about Gacha
29. What are famous gacha games?
30. Tell me about Anime RPG
31. What are famous Anime RPG?
32. That’s helpful
33. See you later!

34. Do you know about Jujutsu Kaisen?
35. I like Jujutsu Kaisen!

Text demo:

Turn-1 User : Good day!

```
Anime bot : Hey wassup!
```

Turn-2 User : What can you do?

```
Anime bot : I can talk about anime and their rankings. I can also talk about my favorite animes and discuss about them in detail.
```

Turn-3 User : Any anime to watch?

```
Anime bot : Koe no Katachi
```

Turn-4 User : Top action anime?

```
Anime bot : Top 5 Action Animes:
1) Fullmetal Alchemist: Brotherhood
2) Attack on Titans
3) Gintama
4) Hunter x Hunter
5) Code Geass
```

Turn-5 User : What is the best adventure anime?

```
Anime bot : Top 5 Adventure Animes:
1) Cowboy Bebop
2) Spirited Away
3) Mushishi
4) Made in Abyss
5) Vinland Saga
```

Turn-6 User : Any top-rated drama anime?

```
Anime bot : Top 5 Drama Animes:
1) March Comes In Like A Lion
2) A Silent Voice
```

```
3) Clannad
4) Your Name.
5) Haikyuu!
```

Turn-7 User : Best fantasy anime to recommend?

```

Anime bot : Top 5 Fantasy Animes:
1) Fate/Stay Night
2) RE:Zero
3) One Piece
4) Weathering With You
5) Mushoku Tensei: Jobless Reincarnation
```

Turn-8 User : What are the top-rated comedy anime?

```

Anime bot : Top 5 Comedy Animes:
1) Owarimonogatari
2) Grand Blue
3) Konosuba
4) Kaguya-Sama: Love is war
5) Horimiya
```

Turn-9 User : What is your favorite anime?

```

Anime bot : I absolutely love Konosuba
```

Turn-10 User : Tell me about Konosuba!

```

Anime bot : Plot for Konosuba: Story about a High
school student who is reincarnated in a
new world but his pathetic luck follows him
even there. Given the task of defeating the
demon lord with his incompetent companions
Will our protagonist be able to live the lavish
life he dreamt of?
```

Turn-11 User : Who is your favorite character?

```

Anime bot : I like Kazuma, the main protagantist of
Konosuba, because he is really funny!
```

Turn-12 User : That's cool

```
Anime bot : My pleasure
```

Turn-13 User : What anime are you watching recently?

```
Anime bot : Anime I am watching nowadays:
1) Attack on Titans Final Season
2) Dr. Stone: Stone Wars
3) Horimiya
4) Re:Zero Season 2 Part 2
5) Mushoku Tensei
```

Turn-14 User : Do you know about manga?

```
Anime bot : Oh Yes! I love mangas too, they are
just like comic books but japanese style.
Many animes are adapted from manga.
```

Turn-15 User : Top-rated manga?

```
 Anime bot : Top 5 Mangas:
 1) Beserk
 2) JoJo's Bizarre Adventure
 3) One Piece
 4) Vagabond
 5) Monster
```

Turn-16 User : What is your favorite manga?

```
Anime bot : Beserk is the best manga!
```

Turn-17 User : Tell me about Beserk!

```
Anime bot : Plot for Beserk: Former mercenary,after
a tumultuous childhood, finally finds someone
he respects and believes he can trust, only to
have everything fall apart when this person
takes away everything important to him. Now
```

```
marked for death he is relentlessly pursued
by demonic beings.
```

Turn-18 User : Who is your favorite Beserk character

```
Anime bot : I like Guts, the main protagantist of
Beserk, because he has a will of iron!
```

Turn-19 User : Manga that you read recently

```
 Anime bot : Manga I am reading nowadays:
 1) Tokyo Ghoul
 2) Bleach
 3) Fairy Tail
 4) Solo Leveling
 5) Tower of God
```

Turn-20 User: Recommend me manga

```
 Anime bot : Nisekoi
```

Turn-21 User : Do you know about Manhwa?

```
  Anime bot : Manhwa are korean comic books just
like manga from japanese culture. They are
very popular too
```

Turn-22 User : What are famous Manhwa?

```
Anime bot : Famous Manhwa:
1) Solo Leveling
2) Tower of God
3) God of Highschool
4) Noblesse
5) Sweet Home
```

Turn-23 User : What is Otaku culture?

```
 Anime bot : Otaku is a string of Japanese culture
 where specifically people in foreign countries
```

```
    have a passion for certain Japanese things
    such as manga, anime, video games, idols
    and anime conventions.
```

Turn- 24: What is anime convention?

```
    Anime bot : An anime convention is an event or
    gathering with a primary focus on anime,
    manga and Japanese culture. Commonly,
    anime conventions are multi-day events
    hosted at convention centers, hotels or
    college campuses.
```

Turn-25 User : What are famous conventions?

```
  Anime bot : Famous Conventions:
  1) Comiket
  2) Jump Festa
  3) Tokyo Toy Show
  4) AnimeJapan
  5) World Cosplay Summit
```

Turn-26 User : What convention do you want to vist?

```
    Anime bot : I really really want to visit Jump Festa,
    to witness the glory of shounen.
```

Turn-27 User : Tell me about Jump Festa!

```
    Anime bot : Jump Festa is sponsored by Shueisha
    Inc. publisher of the Jump manga anthologies
    whose roster includes One Piece, Dragon Ball,
    Naruto, Gintama, Haikyuu!!, Assassination
    Classroomand many more. The event is a
    prime forum for introducing new movies,
    manga,games and merchandise. Keep an
    eye out for plenty of cosplay as well!
```

Turn-28 User : Tell me about Video games

```
Anime bot : Video games are a very important part
of otaku culture, in which the most famous
categories are Gacha games and Anime
RPGs.
```

Turn-29 User : Tell me about Gacha

```
 Anime bot : Gacha games are video games that
 implement the gacha mechanic. Similar to
 loot boxes in video games, gacha games
 induce players to spend in-game currency to
 receive a random virtual item.
```

Turn-30 User : What are famous gacha games?

```
 Anime bot : Popular Gacha Games:
1) Genshin Impact
2) Dragon Ball Legends
3) Dragon Ball Dokkan Battle
4) Fate/Stay Night
5) Runterra
```

Turn-31 User : Tell me about Anime RPG

```
Anime bot : Anime RPGs are video games that
allows players to take control of their favorite
character from an anime and live their story.
It ranges from open world games to hack &
slash games.
```

Turn-32 User : What are famous Anime RPG?

```
Anime bot : Popular Anime RPG:
1) Person 5:Strikers
2) Dragon Ball: Kakarot
3) Final Fantasy 15
4) Dragon Quest XI
5) Dragon Ball: Xenoverse 2
```

Turn-33 User : Tell me about Japan?

```
Anime bot : Japan has a fascinating and multifaceted
culture; on the one hand it is steeped in the deepest
of traditions dating back thousands of years; on the
other it is a society in a continual state of rapid
flux, with continually shifting fads and fashions and
technological development that constantly pushes back
the boundaries of the possible.
```

Turn-34 User : Tell me about People in Japan?

```
Anime bot: Japanese people appear at first glance to be
one of the most socially and ethnically homogenous groups
in the world.
```

Turn-35 User : Tell me about Religion in Japan?

```
Anime bot : Japan's de facto national sport (although this
is not an official status) is the enthralling and at times
bewildering spectacle that is sumo. Deeply rooted in Japan's
culture, sumo has a history of over 1,500 years. Legend has
it that the very survival of the Japanese people balanced on
the outcome of a sumo match between the gods, and indeed sumo
originated as a form of Shinto ritual.
```

Turn-36 User : Tell me about food in Japan?

```
Anime bot : When it comes to food, the Japanese are among the
most enthusiastic and passionate of any race. Ask any Japanese
person about a recent trip within Japan and the conversation
almost always includes talk of the local food. In fact, for many
Japanese travelling outside of their hometowns, food is often one
of the primary motivators for travelling.
```

Turn-37 User : Tell me about Samurais?

```
Anime bot : In Japanese history, samurai culture and the role this
military class played in shaping Japan is a fascinating one.
Although samurai no longer exist, the influence of these great
warriors still manifests itself deeply in Japanese culture and
samurai heritage can be seen all over Japan - be it a great castle,
a carefully planned garden, or beautifully preserved samurai
```

```
residences. It is also deeply ingrained in the psyche of the Japanese
people.
```

Turn-38 User : That's helpful!

```
Anime bot : My pleasure
```

Turn-39 User : See you later

```
Anime bot : Bye! Come back again soon
```

Here are two unique cases where the chatbot doesn't work as intended:

Case 1:

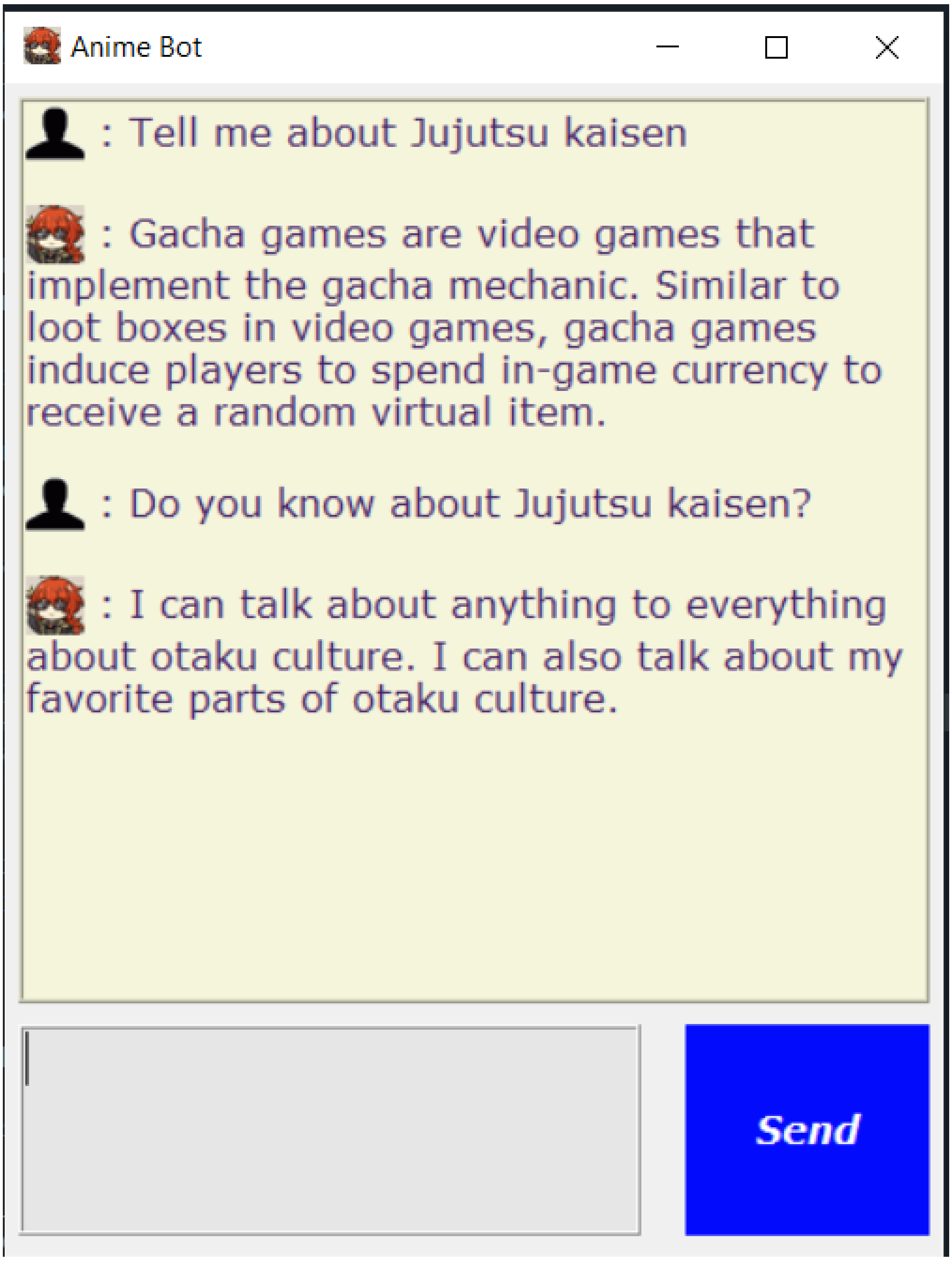
Turn-1 User : Tell me about Jujutsu Kaisen

```
   Chatbot :  Anime bot : Gacha games are video games that
 implement the gacha mechanic. Similar to
 loot boxes in video games, gacha games
 induce players to spend in-game currency to
 receive a random virtual item.
```

Case 2:

Turn-2 User: Do you know about Jujutsu Kaisen?

```
   Chatbot : I can talk about anything to everything
     about otaku culture. I can also talk about my
 favourite part of otaku culture.
```



: Tell me about Jujutsu kaisen

: Gacha games are video games that implement the gacha mechanic. Similar to loot boxes in video games, gacha games induce players to spend in-game currency to receive a random virtual item.

: Do you know about Jujutsu kaisen?

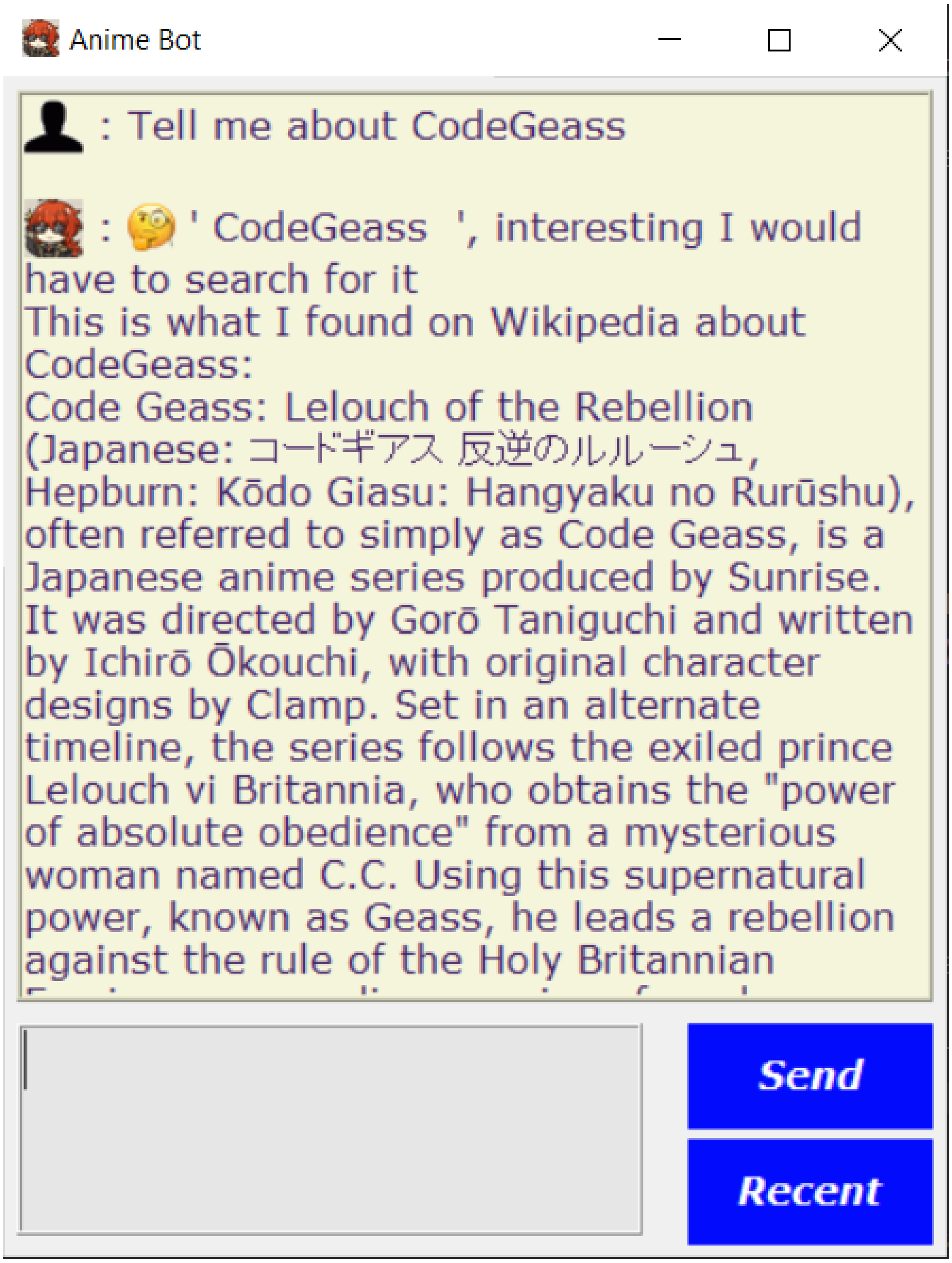
: I can talk about anything to everything about otaku culture. I can also talk about my favorite parts of otaku culture.

3/New APIs implementations:

Wikipedia API

- Whenever the user asks for something that is not in the database, the chatbot looks up for the keyword and search for that keyword in Wikipedia database.

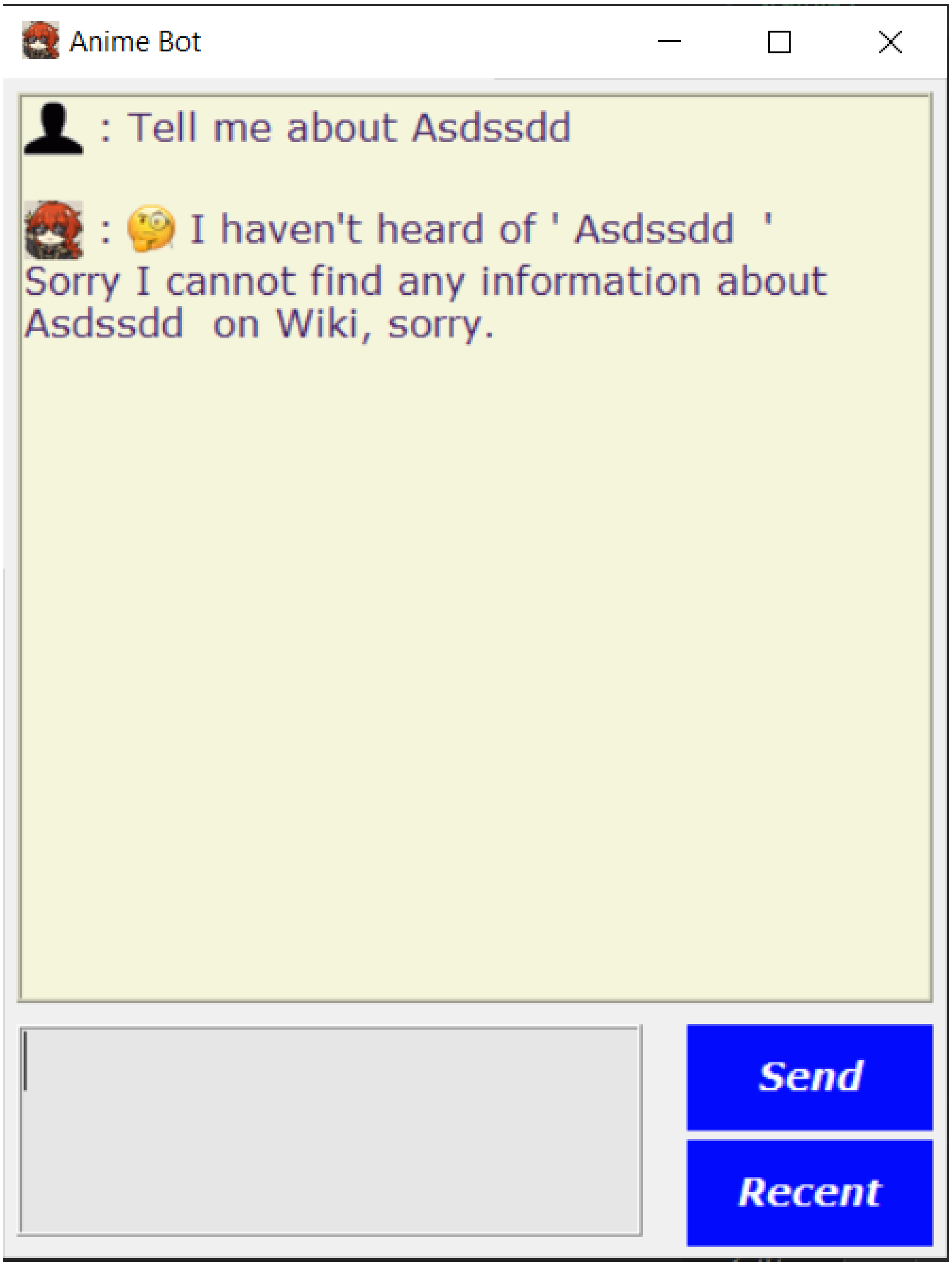
- In the example below, the bot recognizes "CodeGeass" as the keyword and tries searching in wikipedia.



: Tell me about CodeGeass

: ' CodeGeass ', interesting I would have to search for it
This is what I found on Wikipedia about CodeGeass:
Code Geass: Lelouch of the Rebellion (Japanese: コードギアス 反逆のルルーシュ, Hepburn: Kōdo Giasu: Hangyaku no Rurūshu), often referred to simply as Code Geass, is a Japanese anime series produced by Sunrise. It was directed by Gorō Taniguchi and written by Ichirō Ōkouchi, with original character designs by Clamp. Set in an alternate timeline, the series follows the exiled prince Lelouch vi Britannia, who obtains the "power of absolute obedience" from a mysterious woman named C.C. Using this supernatural power, known as Geass, he leads a rebellion against the rule of the Holy Britannian

- If there is no page that matches the information that is asked, the bot will say that it cannot find any match.

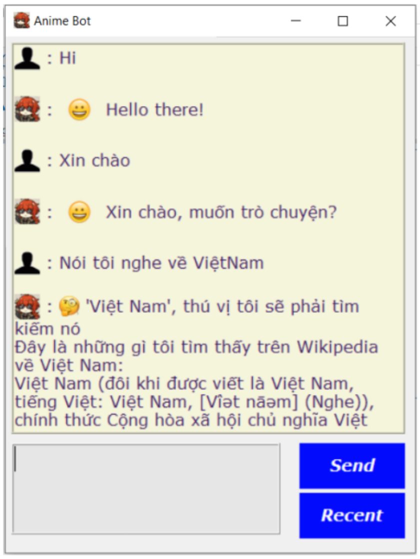


: Tell me about Asdssdd

: 🧐 I haven't heard of ' Asdssdd '
Sorry I cannot find any information about Asdssdd on Wiki, sorry.

Google translate API

- Whenever the user asks for something in another language than English, the chatbot automatically translates the sentence to English and processes using the data in the translated sentence. At the end, it translate the reply into the language that the user used. It also apply to the Wiki API.
- In the example below, the user uses Vietnamese and Chinese as the input, and the bot then replies in the according languages. However, sometimes because the translation is not accurate, the bot will not receive correct translation of the sentence which is demonstrated in the third line of the Chinese-translation image where the bot says that "I dont understand what 'goodbye' is"



: Hi

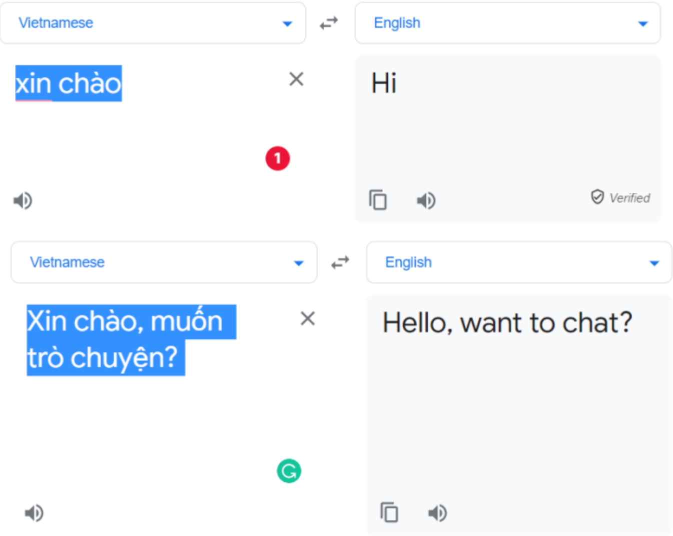
: 😀 Hello there!

: Xin chào

: 😀 Xin chào, muốn trò chuyện?

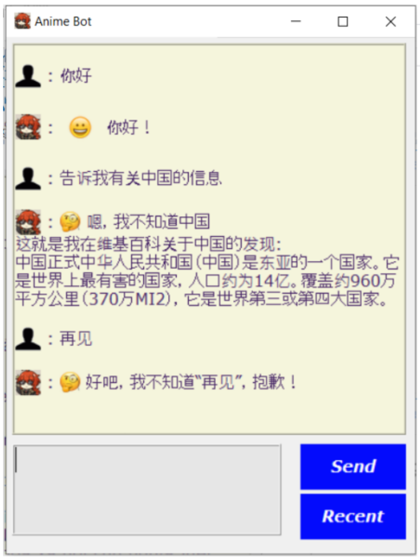
: Nói tôi nghe về ViệtNam

: 🧐 'Việt Nam', thú vị tôi sẽ phải tìm kiếm nó
Đây là những gì tôi tìm thấy trên Wikipedia về Việt Nam:
Việt Nam (đôi khi được viết là Việt Nam, tiếng Việt: Việt Nam, [Vìət nāəm] (Nghe)), chính thức Cộng hòa xã hội chủ nghĩa Việt



xin chào → Hi

Xin chào, muốn trò chuyện? → Hello, want to chat?



: 你好

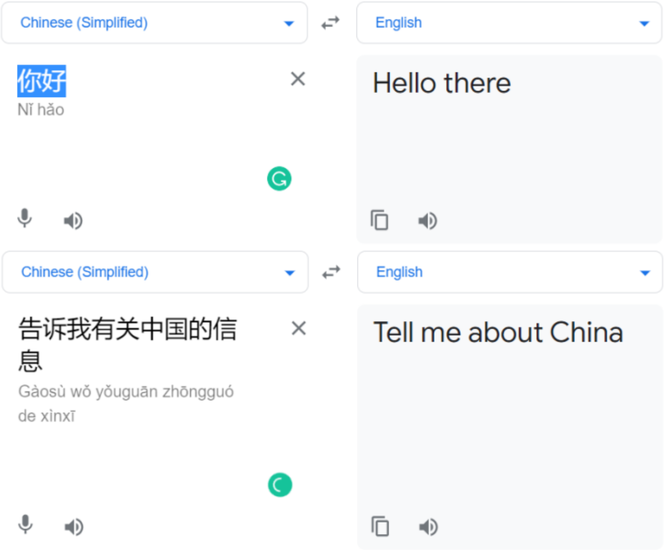
: 😀 你好！

: 告诉我有关中国的信息

: 🧐 嗯，我不知道中国
这就是我在维基百科关于中国的发现:
中国正式中华人民共和国（中国）是东亚的一个国家。它是世界上最有害的国家，人口约为14亿。覆盖约960万平方公里（370万MI2），它是世界第三或第四大国家。

: 再见

: 🧐 好吧，我不知道“再见”，抱歉！



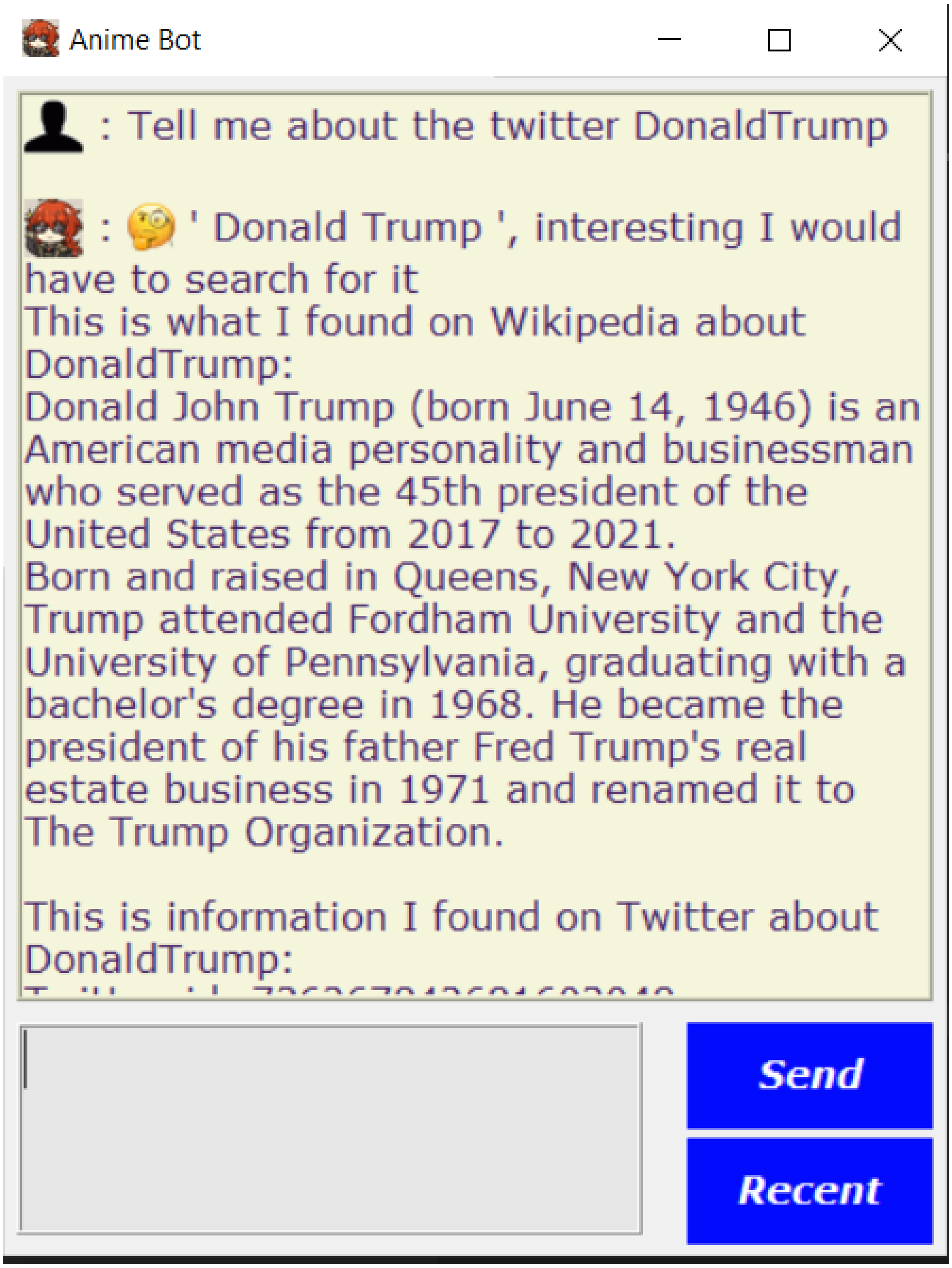
你好
Nǐ hǎo
→ Hello there

告诉我有关中国的信息
Gàosù wǒ yǒuguān zhōngguó de xìnxī
→ Tell me about China

Twitter API

- Whenver user inputs a sentence that includes the keyword "twitter" or "tweet", the chatbot will automatically uses Twitter API to look for any tweets that contain the keyword or profile that has the same name as the keyword. The wikipedia and googletranslate APIs will also work with this API.

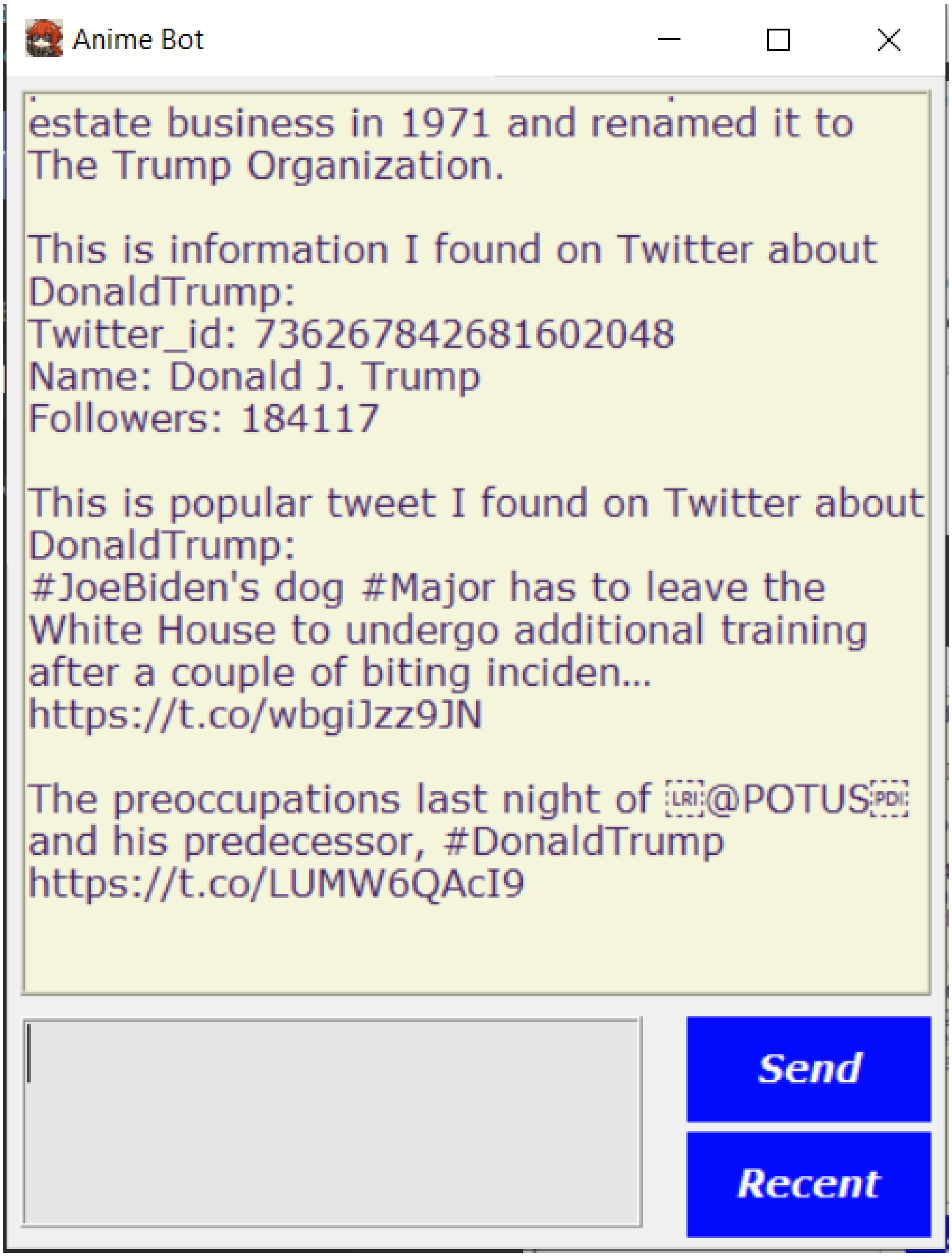
- In the example below, it searches the twitter of Donald Trump and any tweets that contain the keyword "Donald Trump"



: Tell me about the twitter DonaldTrump

: ' Donald Trump ', interesting I would have to search for it
This is what I found on Wikipedia about DonaldTrump:
Donald John Trump (born June 14, 1946) is an American media personality and businessman who served as the 45th president of the United States from 2017 to 2021.
Born and raised in Queens, New York City, Trump attended Fordham University and the University of Pennsylvania, graduating with a bachelor's degree in 1968. He became the president of his father Fred Trump's real estate business in 1971 and renamed it to The Trump Organization.

This is information I found on Twitter about DonaldTrump:



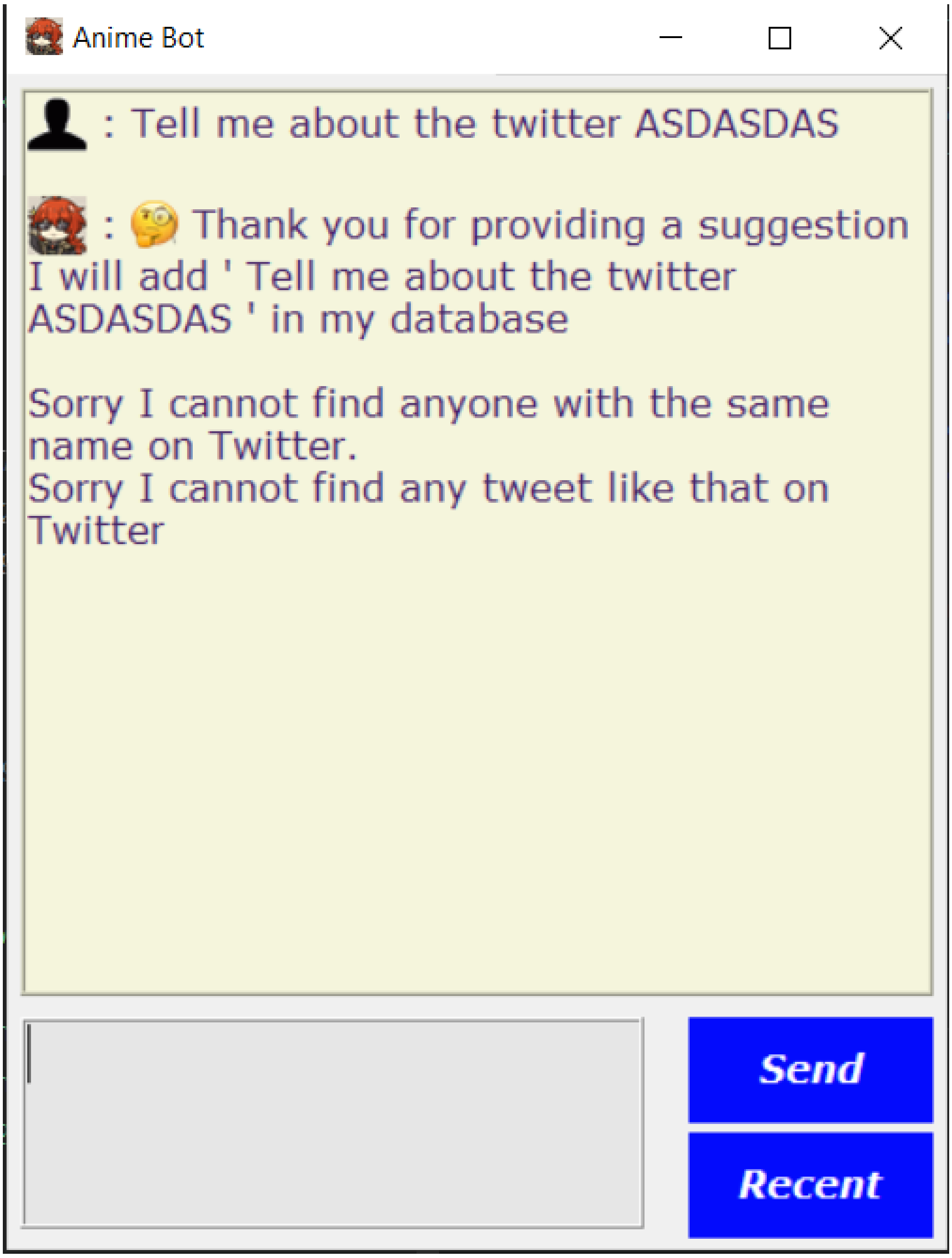
estate business in 1971 and renamed it to The Trump Organization.

This is information I found on Twitter about DonaldTrump:
Twitter_id: 736267842681602048
Name: Donald J. Trump
Followers: 184117

This is popular tweet I found on Twitter about DonaldTrump:
#JoeBiden's dog #Major has to leave the White House to undergo additional training after a couple of biting inciden… https://t.co/wbgiJzz9JN

The preoccupations last night of @POTUS and his predecessor, #DonaldTrump https://t.co/LUMW6QAcI9

- If there is no profile or tweets that match the keyword, the bot will indicate that there is no one or no tweet that matches the keyword.



: Tell me about the twitter ASDASDAS

: 🧐 Thank you for providing a suggestion
I will add ' Tell me about the twitter ASDASDAS ' in my database

Sorry I cannot find anyone with the same name on Twitter.
Sorry I cannot find any tweet like that on Twitter

4/Credits

Team Members and nick name used in the project:

- **Khai Hung Luong (shiro102)**: I'm 3rd year comsci student who loves reading books !
- **Anshul Dhariwal (Anshul)**: I am 3rd year COSC student and loves anything that is interesting to do!!
- **Jayant Puri (Jayant)**: I'm a 3rd year COSC student who loves watching anime!
- **Sirus Wang (ourinsou)**: I'm a 3rd year COSC student who loves taking photos!
- **Shaohua Jiang (Joseph)**: I'm a 3rd year math student who loves snowboarding!

References

Here is the link of an open-source program that helps us build our chatbot https://data-flair.training/blogs/python-chatbot-project/

Here is the link of stanfordcorenlp which is a Python wrapper for Stanford CoreNLP. https://github.com/Lynten/stanford-corenlp

Here is the link of googletrans API https://py-googletrans.readthedocs.io/en/latest/

Here is the link of wikipedia API https://pypi.org/project/Wikipedia-API/

Here is the link of twitter API https://python-twitter.readthedocs.io/en/latest/