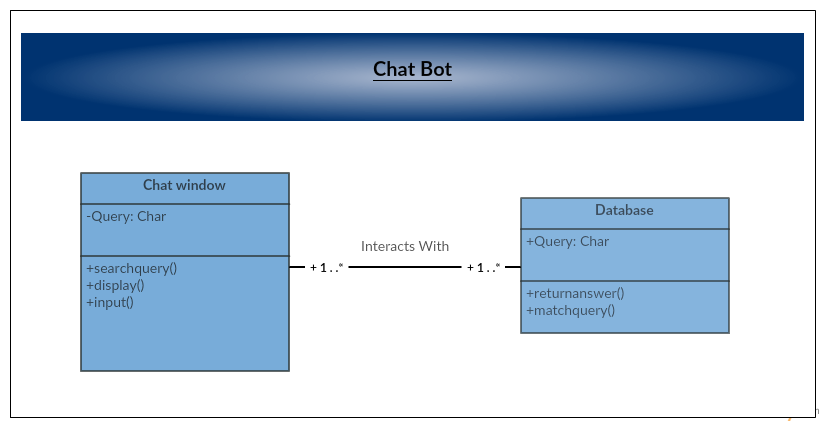
**Product Design**

|  |  |
| --- | --- |
|  | **SSAD 03**  **Anubhab Sen, Vighnesh Chenthil Kumar, Ananya Trivedi** |

**Design Model**

1. **Chat**

|  |  |
| --- | --- |
| Customer | Class state:  Types his/ her queries into the chat window text box provided  Receives response from the bot  Class behavior:  Thinks of query.  Inputs it |
| Chat window | Class state:  Acts as a middle-man between the customer and the database  Sends the query from customer to database  Gets response from database and displays to customer  Class behavior:  Input(): Gets query from customer  Display(): Displays whatever is sent  Retrieve(): Retrieves response from database |
| Database | Class state:  Processes the input string sent by customer  Matches with a response by keywords and substitutions  Returns appropriate response  Class behavior  Matchquery(): processing and matching the input  Returnanswer(): Send answer back to interface |

**2. Extract**

|  |  |
| --- | --- |
| Database | Class State:  Processes the input string sent by the customer  Stores the contact information  Returns appropriate response  Class Behaviour:  Store(): store the information |
| Customer | Class State:  Types his/her information into the chat window text box provided  Receives response from the bot  Class Behaviour:  Inputs information |
| Chat Bot | Class State:  Acts as a middleman between customer and database  Interact with the user to extract the user information in a friendly manner  Sends the information from customer to database  Get response from the database and display it to the customer  Class Behaviour:  Searchforemail(): Search for email in the database, and update if necessary  Display(): Shows whatever is sent  Return(): Get response from the database |

**3. Train**

|  |  |
| --- | --- |
| Programmer | Class State:  Train the Bots with FAQs  Use AIML to train how the bot replies to queries  Adds some casual chat functionalities  Class Behaviour:  Update(): Update the pand\_learn file, each time a new FAQ is found, and train the bot with those. |
| Chat Bot | Class State:  Bot should be trained with the FAQs as well some casual chat dialogues to enhance the user experience.  Class Behaviour:  AcceptQuery(): Bot should be trained to accept query from the user, in a friendly manner  Display(): Shows whatever is sent  Return(): Get response from the database |

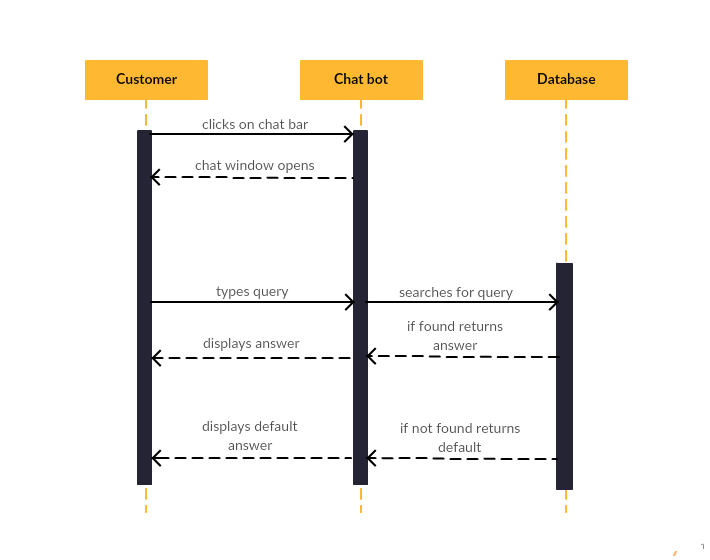
**4. Recommend**

|  |  |
| --- | --- |
| Customer | Class State:  Types his/her queries into the chat window text box provided  Receives response from the bot  Class Behaviour:  Inputs queries |
| Chat Bot | Class State:  Find similar questions to recommend to the user  Sends the information from customer to database  Get response from the database and display it to the customer  Class Behaviour:  Search(): Search for similar responses to recommend in the database  Input(): Allows user to input the queries  Display(): Shows whatever is sent  Return(): Get response from the database |

**5. Embed**

|  |  |
| --- | --- |
| Heroku | Class State:  Allows to deploy the bot on a personalised website  Allows bot to function at its prime  Class Behaviour:  Display(): Display the bot fronted  Input(): Enables proper input functionalities to bot  Return(): Enables responses by the bot from the database |
| Customer | Class State:  Types his/her queries into the chat window text box provided  Receives response from the bot  Class Behaviour:  Inputs queries |
| Pandora Bot | Class State:  Provides a platform to train the bot  Provides API to deploy the bot to whichever site of our site  Class Behaviour:  Find(): Retrieve any similar queries from the database  Match(): Match the queries with any in the database on finding them  ReturnAnswer(): Return the answer to the query on matching them  Default(): If no matching query found, return a default answer. |

**Sequence Diagram(s)**

1. **Chat**

**2. Extract**



**3.Recommend**

****

**4. Embed**

**5. Train**

**Design Rationale**

Our objective was to make a chat-bot to replace FAQ sections of web pages. It had been ascertained by our client that lots of major businesses have archaic help sections in their web pages. It didn’t help that these sections skipped some questions, which the consumers might have, and so she would have no way to get an answer to that other than calling up the customer care. This process is often tiresome and sometimes doesn’t resolve the problem quick enough for it to be of any value to the consumer.

Our bot comes in here and replaces the entire mess the help sections are with a simple chat box interface. The business gives a set of pre decided questions and answers to us and we train the bot with well enough so that it understands the questions even when asked in a different way.

Sometimes a business cares about the amount of time a person spends on their website and want to maximize it. This could be to increase revenue flow from ads or any other thing. To facilitate this, we also trained the bot extensively to be able to converse casually. We have given the bot human attributes such as a name, age, nationality etc to give it a more humane feel. It can hold conversation with a person without getting repetitive for a long time. One thing our bot can’t do is use crowd sourced information or current affairs. So if we were to ask it about the scores of a football match last night, it would be clueless. But we figured this wouldn’t be necessary since the primary reason for the bot would be answering pre determined questions not dynamic ones.

Also we will have a database which will store all the questions asked to the bot, this way in case new questions are asked which the bot can’t answer, can be recovered from the database and can be added to the bot’s knowledge base i.e. trained.

Also a big red button will be added which can be used by the user as an escape route when the bot isn’t able to answer their question, the button will link to a messenger live chat with a real company representative who can answer the question immediately.

We have also added inappropriate filters so that the bot can’t be used to talk about stuff the site owners won’t approve of. This can easily be reverted if the owner wants to.

Overall this would be a beautiful replacement for clumsy current help sites and make it really easy for the end users.