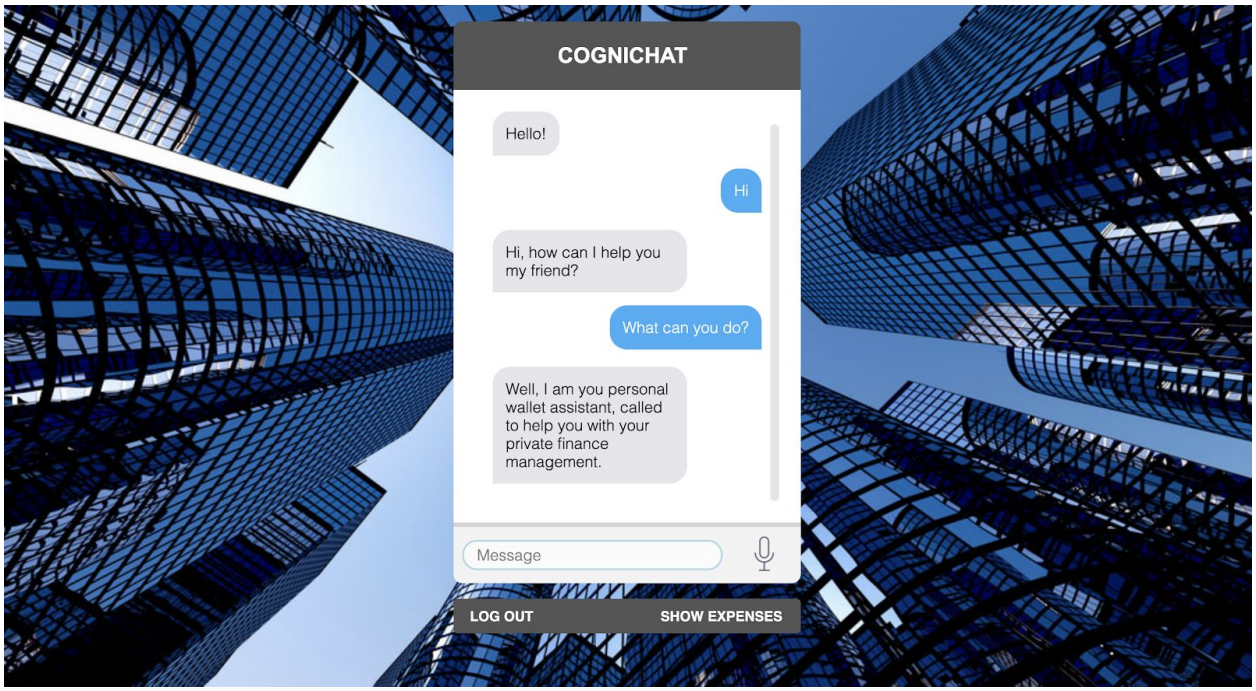

COGNICHAT

An Intelligent Chatbot for Personalised Financial Services

Project Mentor: Dr. Poonam Saini, Faculty, Computer Science and Engineering

Team Members:
Ishika Agarwal (18103034)
Devanshi Garg (18103045)
Riddhi Mahajan (18103110)
Ishita Arora (18103130)

About CogniChat

CogniChat is a solution for personalized financial assistance with an easy to use interface. It takes the form of a responsive web application, wherein the user can post his/her queries and get appropriate responses. CogniChat acts as a simple, intelligent bot that keeps track of information regarding your private finances. It enables the user to check his/her account balance, total expenditure, analyze his/her spending habits and inspect where and how the money is being spent.

Motivation

In today's digital world, managing finance is a challenge being faced by individuals. Most often, users are unable to keep a track of where they are spending their money and hence they need some assistance to keep a check on them.

Our aim was to build a chatbot which can provide the user with information about his/her financial status in an interactive manner. In addition to some basic financial assistance, CogniChat also intelligently categorizes the expenses thus keeping the user informed of his/her spending habits.

Features

- The user can input his/her query either verbally or by typing in a message on the interface.
- The user can access his/her total account balance.
- The user can access his/her total expenditure.
- The user can verify whether or not s/he can buy an item based on the amount of money present in his/her wallet.
- The user can inform the chatbot about every activity where s/he has spent money. The bot automatically infers the category of expenditure and updates it in the database.
- The various categories our chatbot can recognize are:
    - Food
    - Housing
    - Recreation
    - Transportation
    - Healthcare
    - Utilities
    - Miscellaneous
- Categorical expenses can be accessed and analysed using a pie-chart.

Web Technologies

- **DJANGO:** Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design.

  Inbuilt Django libraries have been used for user authentication and to add “forgot password” functionality.

- **SQLite:** SQLite is a relational database management system (RDBMS) contained in a C library. It is the default database system provided by Django. We have SQLite to store information about the user and his/her various expenses.
- **HTML:** Hyper Text Markup Language is the standard markup language for creating Web pages. We have described the basic structure of our templates using HTML.

  We have 4 web pages: **Login, Registration, ChatBot, PieChart Representation**.

- **CSS and BOOTSTRAP:** Cascading Style Sheets describes how HTML elements are to be displayed on screen. We have styled our web pages using CSS and Bootstrap.
- **JAVASCRIPT:** It is used to program the behavior of web pages and make them interactive.
  - **SPEECH RECOGNITION**: We have used the Javascript framework **webkitspeechrecognition** to facilitate speech to text conversion in our chatbot.
  - **AUTOMATIC SCROLLING:** Facilitates automatic scroll down to the bottom of the page in case the length of the page increases
  - **LOADING SYMBOL:** Makes the chatbots conversation more interactive by inserting a loading symbol during the buffer time taken by the chatbot to predict the output.
  - **DYNAMIC SIZED TEXT-BOX:** JavaScript has been used to make our input text-box dynamic in size by automatic insertion of div tags
  - **PIE-CHART REPRESENTATION**: Chart.js is an open source library for data visualization which we have used to represent our categorical expenses.

- **AJAX:** Asynchronous JavaScript And XML is used to send and retrieve data from a server asynchronously without interfering with the display and behaviour of the existing page.
  - **SEND USER QUERY:** Intents corresponding to the user queries are sent as a request to the server.
  - **RETRIEVE PREDICTIONS:** Utterances which are predicted by the AI model are sent back as response from the server.