# CSI 5180 TOPICS IN AI : VIRTUAL ASSISTANTS

## Programming Project : Vision - The Virtual Assistant

**Prepared By:**

Shubham Kulkarni (300276475)
Snehal Sudhir Bhole (300318105)

**Under the guidance of:**

Professor Professor Caroline Barrière (Ph. D)

# Project Summary

## Objective

To develop a virtual assistant capable of performing a range of tasks including answering questions, getting the weather, getting stock prices, summarizing research papers, retrieving bus schedules, and loading news.

## Deliverables

- Integration of Automatic Speech Recognition (ASR) for getting commands
- Question - Answering capability for general queries using OpenAI endpoints
- Integration with external APIs for loading information about Stocks, News, Weather
- Text Summarization capability for .pdf files on google drive using OpenAI endpoints

# Methodology

**Project Definition, Requirements Gathering and Analysis**

- Identified the features that are most desirable in a Virtual Assistant
- Analyzed the APIs that can be integrated and their associated costs

**Development Phase I - Basic Functionality**

- Implemented Google Speech to Text ASR to get queries from user and Speech Synthesis Utterance to convert text to speech
- Processed the input using keyword matching and Named Entity Recognition
- Based on the keyword in the input, determined the task that had to be performed

# Methodology

**Development Phase 2 - Developing Features**

- Question-Answering - Handled by gpt-3.5-turbo model from OpenAI ($0.002/1K Tokens)
- Stocks - NER is used to identify Company name. Company name and ticker symbol mapping is done from NASDAQ and realstonks api from rapidAPI is used to get real time stock updates
- Transport - Station Name and Number mappings are taken from **OC Transpo** Website and their APIs have been used to get Bus timings for specified routes
- Weather - NER is used to identify location. Using OpenWeatherAPI, current weather of that location is inferred

# Methodology

**Development Phase 3 - Developing Features**

- Text Summarization -
  - Google Developer and OAuth tokens generated to access files on google.
  - Using PyPDF2 content inside the files is loaded.
  - By using a recursive function we point to text-davinci-002 to get the summary

**Testing Phase**

- Designed End-to-End testing workflows
- Made the API Keys accessible through a config file
- Handled missing and edge cases in individual features

# Activity Table

| Activity | Why | Time | Deliverable |
|---|---|---|---|
| Reading Existing Articles and Technology Analysis | Gathering Knowledge and Understanding capacity of existing models | **5h** | Comparative analysis of existing systems |
| Integrating speech to text and text-to-speech models | Utilizing existing speech recognition models and implementing them in chatbot. | **5h** | Successful integration of modules with accurate outputs. |
| NER Detection and Keyword Trigger | To correctly identify the intent of the task from the input | **3h** | Specific Tasks should get processed based on the keywords in text |
| Integrating APIs for processing the requests | Adding modules to fetch real-time information from inputs | **20h** | APIs should work by gathering real time information |
| Front End GUI development | For user-friendly design and easy navigation. | **10h** | UI Should be able to process text and speech inputs and responses |
| Testing | To make sure all the components are integrated and work in sync. | **7h** | Test the components individually and collaboratively. |

# Demo

Working of the Software Developed

# Challenges

## Time Delay in ASR

Determining the time duration for which query should be listened

Waiting for response generation and output

Triggering ASR again

## Integrating API Calls

Setting up account for each API service

Understanding the workflow of individual APIs

Processing responses for individual models

## Edge Cases

Case Sensitivity in queries

Mapping the correct entities

Missing cases discovered through Testing and Continuous Development

# Learnings

## Natural Language Understanding

- Understanding the intent of a query and processing it accordingly
- Developing a list of keywords and acing Named Entity Recognition

## Continuous Development

- The process of building a software is iterative, multiple components are involved
- New use cases and scenarios can be discovered

## Leveraging Existing Technology

- Utilize State of the Art Models
- Understand workflows for multiple API endpoints

## Workflow

- The workflow need to be conversational
- Error Handling should be robust

# Conclusion

- We were able to successfully build a chatbot that can answer basic queries
- We integrated multiple API endpoints to process requests for the chatbot
- This chatbot can work in real world setting and is ready to ship as-is
- Building a Virtual Assistant needs combination of Software Development and Machine Learning Skills

# Resources

- Source Code - Github
- OpenAI APIs - Source
- OC Transpo API - Source
- RealStonks RapidAPI - Source
- OpenWeather API - Source
- Bing News Search RapidAPI - Source

# Thank You!