**PROJECT SYNOPSIS**

**GROUP STUDENTS INFORMATION:**

|  |  |  |
| --- | --- | --- |
| NAME | ROLL NUMBER | BRANCH & SEMESTER |
| Sejal | 210710304045 | CE 5th sem |
| Shabnam Thakur | 210710304046 | CE 5th sem |
| Yashswi Sharma | 200710304039 | CE 5th sem |
| Priyanshi Thakur | 220720304004 | CE 5th sem |

**TITLE: VERTIGO-Desktop assistant**

**Objective And Scope:**

**Objective:** The primary objective of Vertigo is to assist users with various tasks and provide a seamless user experience on their desktop or laptop computers.

**Scope:**

**voice Recognition:** Implement voice recognition capabilities using libraries like SpeechRecognition to allow users to interact with Vertigo using voice commands.

**User Interface:** Design a graphical user interface (GUI) for Vertigo that provides a user-friendly way to interact with the assistant. Tkinter or PyQt can be used for this purpose.

**Text-based Interaction**: Allow users to interact with Vertigo through text commands and responses in addition to voice commands.

**Basic Functions:**

**Web Search:** Implement web scraping or utilize APIs to perform web searches and provide users with information from the internet.

**Weather Updates:** Provide real-time weather information for a specified location.

**Calculator:** Implement a basic calculator function for mathematical calculations.

**File Operations:** Allow users to perform file-related tasks such as creating, deleting, moving, and searching for files.

**Communication:**Facilitating communication tasks, such as sending messages, making calls, or managing contacts..

**Methodology**:

1. **Define Objectives and Scope:**Clearly define the objectives and scope of your desktop assistant.
2. **Research and Planning:**Research existing desktop assistant solutions to understand their capabilities and limitations.
3. **Choose Technologies:**Decide on the technologies and libraries you'll use. Common choices include Python, speech recognition libraries (e.g., SpeechRecognition).Select a GUI framework for building the user interface, such as Tkinter or PyQt.
4. **Design User Interface:**Design the graphical user interface (GUI) for your desktop assistant. Consider the user experience and ensure it's user-friendly.
5. **Implement Core Functionality:** Start by implementing the core functionalities of your assistant, such as voice recognition, text input handling, and basic command execution.
6. **Implement Core Functionality:**Start by implementing the core functionalities of your assistant, such as voice recognition, text input handling, and basic command execution.
7. **Voice Recognition:** Set up voice recognition using SpeechRecognition to convert spoken commands into text.
8. **Error Handling:** Implement error handling to gracefully handle unexpected user inputs or system errors.
9. **Security and Privacy**: Implement security measures to protect user data and privacy, especially when handling sensitive information.

**Hardware and software**

**Hardware:**

* **Computer:** A machine with sufficient computational power to handle the dataset and perform analysis efficiently.
* **Storage:** Adequate storage space to accommodate the dataset and generated outputs.
* **Microphone:** Required for speech input. Ensure your system has a functioning microphone
* **Speakers or Headphones:** Necessary for the text-to-speech output.
* **Internet Connection:** Required for fetching weather information, news, and playing music from YouTube.
* **Adequate System Resources:** The code should run on a system with sufficient processing power and memory, as it involves voice recognition, image processing, and web requests

**Software:**

* **Python:** The core programming language for this project.
* **Vscode:**editor to type the code
* **Pycharm:**editor to type the python code
* **Python Libraries:**
* **tkinter:**tkinter is the standard GUI (Graphical User Interface) toolkit that comes with Python. It provides tools for creating desktop applications with graphical elements such as buttons, labels, and entry fields.
* **PIL (Python Imaging Library) or Pillow:**PIL is an older library for image processing in Python, but Pillow is a more up-to-date fork of PIL. It allows for opening, manipulating, and saving many different image file formats.
* **threading:**threading is a module that provides a way to run multiple threads (smaller units of a program) concurrently. It is often used for managing parallel tasks.
* **pyttsx3:**pyttsx3 is a Python library that interfaces with text-to-speech engines. It allows your Python script to convert text into speech.
* **speech\_recognition as sr:**speech\_recognition is a library that provides a simple interface to recognize speech using various speech engines. It allows your script to process audio input.
* **pywhatkit:**pywhatkit is a library that provides an interface to interact with WhatsApp using Python. It allows sending messages on WhatsApp.
* **wikipedia:**The wikipedia library provides an interface to query and retrieve information from Wikipedia.
* **os:**The os module provides a way to interact with the operating system. It is often used for tasks like file manipulation, directory operations, and process control.\
* **subprocess:**The subprocess module provides a way to spawn new processes, connect to their input/output/error pipes, and obtain their return codes.
* **webbrowser:**The webbrowser module provides a high-level interface to allow displaying Web-based documents to users.
* **datetime:**The datetime module supplies classes for working with dates and times.
* **pyowm:**The pyowm library is a Python wrapper around the OpenWeatherMap (OWM) web API. It allows you to fetch weather information.
* **requests:**The requests library allows you to send HTTP requests easily. It is commonly used for interacting with web APIs.
* **random:**The random module provides functions for generating random numbers. It is often used for introducing randomness or unpredictability in a program.

**Application and future scope of the project:**

**Applications:**

* **Application**: The data analysis project can be applied in various domains such as finance, healthcare, marketing, and more, to make informed decisions, optimize processes, and gain a competitive edge.
* **Personal Assistant:**Acts as a personal assistant, providing weather updates, jokes, news, and music playback.
* **Learning and Exploration:**Encourages exploration of information by searching on Wikipedia.
* **Entertainment:**Offers entertainment through jokes and music playback.
* **Information Retrieval:**Fetches real-time information such as weather and news headlines.
* **Voice-controlled Interaction:**Allows users to interact with the system using voice commands..

**Future Development**:

* **Integration with Big Data**: Extend the project to handle larger datasets and integrate with big data technologies like Apache Spark.
* **Real-time Analysis**: Incorporate real-time data processing and analysis for timely decision-making.
* **Enhanced Visualization**: Utilize emerging visualization techniques for more interactive and insightful displays.
* **AI and ML Advancements**: Integrate advanced machine learning and AI algorithms to improve predictive accuracy and analysis depth.

**Project timeline:**

**1. Planning and Research (1week):** Define project goals and scope, research existing tools, and lay the groundwork.

**2. Development (4-5weeks):** Write Python code for core features like timing and accuracy calculation.

**3. Testing and Debugging (2weeks):** Thoroughly test the application, identify and fix bugs, and improve based on feedback.

**4. Deployment (2weeks):** Make your typing speed test available to users, either on a web server or as an application.

**5. Ongoing Maintenance and Updates (ongoing):** Continuously monitor and improve the project, addressing user feedback and adding new features