**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**

**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.

**Reminders and Alarms**: Enable users to set reminders and alarms for specific tasks or appointments.

**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.

**System Information:** Provide information about the user's computer system, such as CPU usage, memory usage, and disk space.

**Notes and To-Do Lists:** Create and manage notes and to-do lists for the user.

**Email and Messaging:** Send emails or messages on behalf of the user.

**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.

Software:

* **Python:** The core programming language for this project.
* **Python Libraries:**
* **Speechrecognition:** For speech recognition in project.
* **Pyttsx3:** pyttsx3 is a text-to-speech conversion library in Python
* **Jupyter Notebook or IDE (e.g., PyCharm):** For code development and interactive analysis.
* **NLTK or spaCy:** For natural language processing (if needed for text analysis).

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.

**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