**Software Requirement Specification**

Project:- Wake Up Detection

For:- Natural Language Processing

Prepared by

|  |  |  |
| --- | --- | --- |
| **Specialization** | **SAP ID** | **Name** |
| AIML (N.H) | 500093960 | Eklavya Gupta |
| AIML(N.H) | 500093996 | Aaryak Bhargava |
| AIML(N.H) | 500093622 | Aryaman Jain |



School Of Computer Science

UNIVERSITY OF PETROLEUM & ENERGY STUDIES,

DEHRADUN- 248007. Uttarakhand

Table of Contents

|  |  |  |
| --- | --- | --- |
| **Topic** | | **Page No** |
| Table of Content | |  |
| Revision History | |  |
| 1 | Introduction |  |
|  | 1.1 Purpose of the Project |  |
|  | 1.2 Target Beneficiary |  |
|  | 1.3 Project Scope |  |
|  | 1.4 References |  |
| 2 | Project Description |  |
|  | 2.1 Reference Algorithm |  |
|  | 2.2 Data/ Data structure |  |
|  | 2.3 SWOT Analysis |  |
|  | 2.4 Project Features |  |
|  | 2.5 User Classes and Characteristics |  |
|  | 2.6 Design and Implementation Constraints |  |
|  | 2.7 Design diagrams |  |
|  | 2.8 Assumption and Dependencies |  |
| 3 | System Requirements |  |
|  | 3.1 User Interface |  |
|  | 3.2 Software Interface |  |
|  | 3.3 Database Interface |  |
|  | 3.4 Protocols |  |
| 4 | Non-functional Requirements |  |
|  | 4.1 Performance requirements |  |
|  | 4.2 Security requirements |  |
|  | 4.3 Software Quality Attributes |  |
| 5 | Other Requirements |  |
| Appendix A: Glossary | |  |
| Appendix B: Analysis Model | |  |
| Appendix C: Issues List | |  |

1. Introduction:-

1.1 Project Objective:

* Our project aims to develop a system for detecting wake words. This solution will improve the user experience by allowing for seamless interaction with gadgets via voice commands. The objective is to make initiating activities and controlling equipment as simple as saying a word or phrase, removing the requirement for physical engagement.

1.2 Target Users

* Our solution is intended for everyone who wants a handy, hands-free way to interact with their gadgets. Individuals with disabilities or limitations who find typical input techniques problematic. But it's not only about accessibility; our solution is intended for anybody who appreciates efficiency and ease in their everyday technological interactions.

1.3 Scope of the Project:

* The project entails creating a dependable wake word detection system that can properly recognize certain trigger words or phrases in real time. We want to employ per-trained models or, if necessary, new models to provide high accuracy and dependability. The final result will be a software solution with documentation and recommendations for deployment across several devices and apps.

1.4 Reference Material:

* Several major sources will guide our efforts to develop an effective wake word detection system.
* Mozilla Deep Speech,
* Julius
* Kaldi

2 Project Descriptions:-

2.1 Choosen Algorithm:

* Our project will use per-trained models for wake word detection, such as TensorFlow's Wake Word Detection (WWDL) model or other publicly accessible models. We may also use transfer learning techniques to fine-tune these models so that they specialize in recognizing the exact wake word(s) that fit with our project's objectives.

2.2 Data Structure:

* The data for training and testing our model will be audio recordings. These recordings will include the wake word(s), background noise, and various speech patterns. The recordings will be categorized and labeled according to the presence or absence of the wake word(s), which will help with model training and evaluation.

2.3 SWOT Analysis:

Strengths.

* Voice instructions provide for intuitive engagement.
* Improved accessibility for those who find physical input methods difficult.
* Potential for integration with different interaction modalities.

Weaknesses:

* Potential accuracy limits exist in loud surroundings or with pronunciation differences.
* Sensitivity to background noise and different speech patterns
* Potential lag in waking word detection and reaction times.

Opportunities:

* Scope of optimization to increase wake word detection accuracy and resilience.
* Integration possibilities with current voice-controlled apps and devices
* Options for customization to meet the requirements and tastes of individual users

Threats:

* Competition from current wake-word detection systems and voice assistants
* Technical problems for decreasing false positives and false negatives
* Concerns about privacy and security while collecting and processing voice data.

2.4 Main Features:

* Wake word detection is the ability to recognize certain trigger words or phrases that launch actions or orders.
* Real-time processing: effective handling of speech inputs with low latency.
* Customization: ability to alter wake word detection parameters based on user preferences.
* Error handling: mechanisms for minimizing false positives and false negatives in wake word detection.

2.5 User Categories and Characteristics:

* General users are those who want to communicate with products and programs using voice commands without having to use their hands.
* Users with disabilities: people who may benefit from voice-controlled interfaces owing to physical or cognitive problems.

2.6 Limitations in Design and Implementation:

* Hardware compatibility: the necessity to assure compatibility with devices that include microphones capable of recording voice inputs.
* Noise reduction: the requirement to incorporate ways for filtering out background noise and improving voice input intelligibility.
* Latency optimization refers to the employment of techniques and optimizations to reduce the latency between wake word detection and system response.

2.7 Design diagrams:

* Use-case diagram: This graphic depicts user interactions with the wake word detection system, such as voice command initiation and system response.
* A class diagram depicts the components and relationships of the wake word detection system, which includes audio processing modules and decision-making algorithms.
* Activity diagram: a diagram that depicts the control flow between wake word detection and subsequent action execution.
* A sequence diagram depicts the order of interactions between system components during wake word detection and response.

2.8 Assumptions and dependencies:

* Assumptions: Users have access to devices with appropriate microphones and enough processing power to handle real-time voice processing.
* Dependencies include the availability of per-trained wake word detection models, audio processing libraries, and compatibility with current voice recognition frameworks.

3 System Requirements:-

3.1 User Interface:

* A graphical interface for system settings and feedback, which includes customization wake word settings and system status display.
* Voice and visual signals are used to provide user feedback during wake word detection, either signaling successful identification or requesting repetition.

3.2 Software Interface:

* Integration with wake word detection API or libraries to process audio inputs and identify wake words.
* Compatible with libraries for real-time voice analysis and transcription.

3.3 Database interface:

* Databases are used to contain models for wake word identification as well as user-defined wake words for more personalized interactions.
* Integration with authentication systems to validate user identities and restrict access to sensitive functions.

3.4 Protocol:

* Standard communication protocols, such as HTTP and WebSocket, are used to interface with external services and applications, allowing for smooth system integration.
* Encryption protocols (e.g., SSL/TLS) are used to secure data transmission between the wake word detection system and external entities, ensuring user interaction privacy and integrity.

4 Non-Functional Requirements:-

4.1 Performance Requirement:

* Real-time wake word detection with low latency ensures rapid system reaction.
* Efficient utilization of system resources to reduce processing overhead while maintaining peak performance.
* Scalability to handle variable loads and meet rising demand for wake-word detection services as the user base expands.

4.2 Security requirements:

* Implemented user authentication measures to prevent unwanted access to wake word detection functions and sensitive user data.
* Encryption of voice input data and wake word detection results protects user privacy and prevents eavesdropping or tampering.
* Regular security assessments and upgrades are performed to identify and address potential vulnerabilities in the wake-word detection system.

4.3 Software Quality Characteristics:

* Adaptability: To ensure constant performance, the system should react to variations in ambient noise levels and user speaking patterns.
* Reliability: The system should detect wake words in a variety of contexts and user scenarios while minimizing false positives and false negatives.
* Usability: The user interface should be simple and easy to use, allowing users to quickly establish wake-word settings and interpret system feedback.
* Maintainability: The system should be modular and well-documented, allowing for simple maintenance, upgrades, and enhancements that increase wake word recognition accuracy and reliability over time.

5 Other Requirements :-

* Integration with Voice Assistants: We want our system to integrate seamlessly with prominent voice assistants like Amazon Alexa and Google Assistant. This enables customers to combine existing voice assistant functionalities with our proprietary wake word detection capabilities.
* We intend to provide open API and SDKs. This allows developers to embed our wake word recognition technologies into their own applications and services, promoting creativity and widening the system's use cases.