**“Smart Virtual Assistant”**

A

Project report submitted to

**INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT, PUNE**

In partial fulfillment for the award of Degree

**In**

**Computer Engineering**

By

**Gajbhare Chandramani D.**

**Kale Rutuja P.**

**Londhe Ankita S.**

**Bhartani Rohit B.**

**Patil Ashvini V.**

**Tarange Swamini B.**

*Under the Guidance of*

**Mr. Tushar P.**

Department of Computer Engineering



INDIRA COLLEGE OF ENGINEERING AND MANAGEMENT, PUNE

(MAHARASHTRA, INDIA)

2020-21

***CERTIFICATE***

This is to certify that the dissertation entitled

**“Smart Virtual Assistant”**

Being submitted here with to the Indira College Of Engineering And Management, Pune for the award of the **‘Degree’** in **‘Computer Engineering’** is a record of bonafied work carried out by **Mr. Gajbhare Chandramani D., Mr. Bhartani Rohit B., Miss. Kale Rutuja P., Miss. Londhe Ankita S., Miss. Patil Aswini V., Miss. Tarange Swamini B.** under my supervision and guidance. The matter contained in this report has not been submitted to any other institute for the award of any degree. This work is carried out under the guidance of

**Mr. Tushar P.**

Guide,

Department of Computer Engineering

**Dr. Vinod Kumar**

Head of Department,

Department of Computer Engineering

**Dr.**

Principal,

Indira College Of Engineering And Management, Pune

# CERTIFICATE

This is to certify that **Mr. Gajbhare Chandramani Deepak**

Group No. 24 Division: DSE Branch: Computer Engineering has successfully completed the work associated with **Project Based Learning II (210258**) titled as SMART VIRTUAL ASSISTANT USING PYTHON and has submitted the work book associated under my supervision, in the partial fulfillment of Second Year Bachelor of Engineering (Choice Based Credit System) (2019 course) of Savitribai Phule Pune University.

Date:

Place: Pune

|  |  |  |
| --- | --- | --- |
| **Guide** | **Head** | **Coordinators** |
| Prof. Tushar Pinjan | Dr. Vinod Kumar | Prof. Uzmamasrat Shaikh |
|  |  | Prof. Manjusha Tatiya |

# CERTIFICATE

This is to certify that **Miss. Kale Rutuja Pravin**

Group No. 24 Division: DSE Branch: Computer Engineering has successfully completed the work associated with **Project Based Learning II (210258**) titled as SMART VIRTUAL ASSISTANT USING PYTHON and has submitted the work book associated under my supervision, in the partial fulfillment of Second Year Bachelor of Engineering (Choice Based Credit System) (2019 course) of Savitribai Phule Pune University.

Date:

Place: Pune

|  |  |  |
| --- | --- | --- |
| **Guide** | **Head** | **Coordinators** |
| Prof. Tushar Pinjan | Dr. Vinod Kumar | Prof. Uzmamasrat Shaikh |
|  |  | Prof. Manjusha Tatiya |

# CERTIFICATE

This is to certify that **Miss. Londhe Ankita Sahadev**

Group No. 24 Division: DSE Branch: Computer Engineering has successfully completed the work associated with **Project Based Learning II (210258**) titled as SMART VIRTUAL ASSISTANT USING PYTHON and has submitted the work book associated under my supervision, in the partial fulfillment of Second Year Bachelor of Engineering (Choice Based Credit System) (2019 course) of Savitribai Phule Pune University.

Date:

Place: Pune

|  |  |  |
| --- | --- | --- |
| **Guide** | **Head** | **Coordinators** |
| Prof. Tushar Pinjan | Dr. Vinod Kumar | Prof. Uzmamasrat Shaikh |
|  |  | Prof. Manjusha Tatiya |

# CERTIFICATE

This is to certify that **Mr. Bhartani Rohit Bhagwan**

Group No. 24 Division: DSE Branch: Computer Engineering has successfully completed the work associated with **Project Based Learning II (210258**) titled as SMART VIRTUAL ASSISTANT USING PYTHON and has submitted the work book associated under my supervision, in the partial fulfillment of Second Year Bachelor of Engineering (Choice Based Credit System) (2019 course) of Savitribai Phule Pune University.

Date:

Place: Pune

|  |  |  |
| --- | --- | --- |
| **Guide** | **Head** | **Coordinators** |
| Prof. Tushar Pinjan | Dr. Vinod Kumar | Prof. Uzmamasrat Shaikh |
|  |  | Prof. Manjusha Tatiya |

# CERTIFICATE

This is to certify that **Miss. Patil Ashwini Vijay**

Group No. 24 Division: DSE Branch: Computer Engineering has successfully completed the work associated with **Project Based Learning II (210258**) titled as SMART VIRTUAL ASSISTANT USING PYTHON and has submitted the work book associated under my supervision, in the partial fulfillment of Second Year Bachelor of Engineering (Choice Based Credit System) (2019 course) of Savitribai Phule Pune University.

Date:

Place: Pune

|  |  |  |
| --- | --- | --- |
| **Guide** | **Head** | **Coordinators** |
| Prof. Tushar Pinjan | Dr. Vinod Kumar | Prof. Uzmamasrat Shaikh |
|  |  | Prof. Manjusha Tatiya |

# CERTIFICATE

This is to certify that **Miss. Tarange Swamini Bharat**

Group No. 24 Division: DSE Branch: Computer Engineering has successfully completed the work associated with **Project Based Learning II (210258**) titled as SMART VIRTUAL ASSISTANT USING PYTHON and has submitted the work book associated under my supervision, in the partial fulfillment of Second Year Bachelor of Engineering (Choice Based Credit System) (2019 course) of Savitribai Phule Pune University.

Date:

Place: Pune

|  |  |  |
| --- | --- | --- |
| **Guide** | **Head** | **Coordinators** |
| Prof. Tushar Pinjan | Dr. Vinod Kumar | Prof. Uzmamasrat Shaikh |
|  |  | Prof. Manjusha Tatiya |

## DECLARATION

I undersigned hereby declare that I have completed and written Project Report entitled “Smart Virtual Assistant”, during academic year 2020-2021 for the fulfilment of the requirement of Savitribai Phule Pune University for the award of the Degree in of Computer Engineering / Information Technology.

This is my original work and it was not previously submitted for the award of any diploma or similar title in any other examining body or board as per my knowledge.

**Name of Student**:

Gajbhare Chandramani

Tarange Swamini B.

Bhartani Rohit B.

Kale Rutuja P.

Londhe Ankita S.

Patil Ashvini V.

**Place:** Indira Institute Of Engineering And Management, Pune

Contents

# Acknowledgment I

List of Figures II

Abstract III

1. SYNOPSIS 1

1.1 Relevant Objectives . . . . . . . . . . . . . . . . . . . . . . . . . . . . 1

* 1. Motivation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2

1.3 Plan Of Project Execution . . . . . . . . . . . . . . . . . . . . . . 3-5

1.4 Problem Statement . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6

1. TECHNICAL KEYWORDS 7
2. INTRODUCTION 8

3.1 Project Idea . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8

3.2 Motivational Survey . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9

1. [PROBLEM DEFINITION AND SCOPE 10](#_TOC_250002)
   1. Problem Definition 10
   2. Goal and Objective 10
   3. Technology and Associated Platform 11
      1. Hardware Specification 11
      2. Software Specification 11
2. [PROJECT PLAN 12](#_TOC_250001)
   1. Problem Definition 12
   2. Design and Implementation Constraints 12
   3. Software Requirement 13
   4. Feasibility Study 13
      1. Financial Feasibility 13
      2. Technical Feasibility 13
      3. Behavioral Feasibility 14
      4. Operational Feasibility 14
3. [SOFTWARE REQUIREMENT AND SPECIFICATION 15](#_TOC_250000)
   1. Purpose and Scope of Document 15
      1. Project Scope 15
   2. External Interface Requirement 15
      1. User Interfaces 15
   3. Assumption and Dependencies 16
   4. System Features 16
      1. System Features-I (Functional Requirements) 16
      2. System Features-II (Non-Functional Requirements) 16
   5. Data Flow Diagram 17
      1. DFD Level 0 17-18
      2. DFD Level 1 19
      3. DFD Level 2 20
   6. UML Diagrams 20
      1. Use Case Diagram 20
      2. Activity Diagram 21
      3. Sequence Diagram 22
      4. Class Diagram 23-24

# DETAILED DESIGN DOCUMENT 25

* 1. System Architecture 25-26

# PERFORMANCE EVALUATION AND TESTING 29

* 1. Introduction 29
     1. Goals and objectives 29
     2. Statement of Scope 39
     3. Major Constraints 30-32
  2. Testing 33
     1. Testing as a Continuous Process 33
     2. Implementation 33
     3. Types of Testing 34-35
     4. Testing Tools and Environment 36
  3. Test Cases 37

# RESULT AND ANALYSIS 38

* 1. Dataset Description 38
  2. Experimental Setup 38

ANNEXURE 42-48

Conclusion 49

REFERENCES 50

**ACKNOWLEDGMENT**

I have a great pleasure in presenting this Project Report on “Smart Virtual Assistant” and to express my deep regards towards those who have offered their valuable time and guidance in our hour of need.

I would like to express our sincere and whole hearted thanks to my project guide Mr. Tushar Sir., Head of the department Prof. Mr. and my project co-coordinator Prof. for contributing valuable time, knowledge, experience and providing valuable guidance in making this project a success.

I am also glad to express my gratitude and thanks to our Principal Prof. sir for their constant inspiration and encouragement.

Finally, before ending I would like to express once again my gratitude and thanks to all my friends who are involved directly and indirectly in making my project success.

.

**Name of Student:**

**Gajbhare Chandramani D.**

**Kale Rutuja P.**

**Londhe Ankita S.**

**Bhartani Rohit B.**

**Patil Ashwini V.**

**Tarange Swamini B.**

# **LIST OF FIGURES:**

* 1. Gantt chart of Timeline Schedule 1 . . . . . . . . . . . . . . . . . . . . 5
  2. Gantt chart of Timeline Schedule 2 . . . . . . . . . . . . . . . . . . . . 5
  3. DFD Level 0 17
  4. DFD Level 1 18
  5. DFD Level 2 19
  6. Use Case Diagram 20
  7. Activity Diagram 21
  8. Sequence Diagram 22
  9. Class Diagram 23

7.1 System architecture 24

8.1 Test Case…………………………………………………………………….37

# **ABSTRACT**

Smart Virtual Assistant (SVA) is the next generation of carrier services. SVA is believed to be the intelligent evolution of services to meet the ever increasing demand by the professionals for desktop and connectivity. This new generation of services will allow desktop users to remotely access and manage information using speech recognition technology over internet.

SVA responds to conversational voice commands and delivers a single point of contact that seamlessly engages a wide range of information. The SVA controls the desktop, manages the personal activities, enables the user to access his task manager via voice interface, and includes all the functions of Unified Messaging.

The SVA enables the user to optimize the user resources (time, cost), enhance his/her overall productivity, and minimize the interruptions to his regular workflow. The Smart Virtual Assistant (SVA) will enable the user to efficiently handle increasing demand of sending emails, messages, bot-chat and other activities.

SVA will enhance and optimize user experience. It will reduce human effort and speed up the tasks. SVA is an AI based application, as application uses Google and amazon speech engine

**Gajbhare Chandramani D.**

# 1 **SYNOPSIS**

Project Title: **Smart Virtual Assistant**

Internal Guide: **Mr. Tushar Sir.**

Technical Keywords: **Python, Google Speech Engine, Amazon Speech Engine, Voice Command, AI, SDK.**

* 1. **Relevant Objectives**

A voice assistant is a digital assistant that uses voice recognition, natural language processing and speech synthesis to provide aid to user through desktop and voice recognition application.

Following are the tasks carried out by assistant:

* Requesting information, similarly to inputting a search query.
* Playing Music.
* Performing Mathematical Calculation.
* Sending Mails

* 1. **Motivation**

We have selected this topic “Smart Virtual Assistant” because we want to help blind peoples. As blind people can’t see so they can’t interact with screen GUI. To overcome this problem we as student have tried to develop an S/W which will listen and execute query. The S/w is capable to perform all operations like send email, Music, chatbot etc.

This S/W will help blind peoples in many scenarios & also it will save their lot of time. Smart Virtual Assistant uses Google and amazon speech reorganization engine. We see in our day to day life technology is rapidly changing so, why not we should develop s/w for blind peoples which will help them to get there work done simple and fast.

In Smart Virtual Assistant, user has to ask a question or tell a machine what task to be done then Smart virtual assistant will listen it and perform the action or execute query .Thus Smart Virtual Assistant is an AI based application.

* 1. **Plan of Project Execution**

Table 1.1 summaries various tasks being carried out in estimated

Duration of weeks. Figure 1.1 and Figure 1.2 shows Gantt chart of various activities against timescale in weekly manner.

|  |  |  |
| --- | --- | --- |
| Work Task | Description | Duration |
| Basic Study | In the first week we were informed about what a project is and how to work on it. We also were told to form groups and choose a leader in the second week. In the third week we were informed to make a research on 10 topics for the project to be developed. Finally, in the fourth week a topic was finalized that would be unique among the other groups. | 3 weeks |
| Review of papers | We referred a lot of papers and even experienced a round of situations where the application will be boosting up the process that manually was a big headache. | 2 weeks |
| Problem Formulation | Currently we found that blind people cannot easily perform day to day task on pc.so to overcome this problem we have designed a s/w which will listen to their voice and perform action according to it.so for them smart virtual assistant is developed, which will reduce blind people works and make their task simple & faster. | 2 weeks |
| Literature Survey | We made a survey by discussing this topic with the friends and our guides. We even had a discussion for the sponsorship they liked the idea but refused due to financial crisis currently. We referred a lot of ieee paper for developing this system. | 3 weeks |
| Objective of Topic | A smart virtual assistant is a digital assistant that uses voice recognition, natural language processing and speech synthesis to provide aid to user through desktop and voice recognition application. | 2 weeks |
| Design of mathematical model | Firstly, we collected information required for preparing design of mathematical model of Project.  Then we prepared Design of Mathematical model of Our System. | 2 weeks |
| Software Requirement and Specification | The development of front end is done using Python.  The technology used are google speech engine, amazon speech engine. | 1 weeks |
| Detailed Design Document | Firstly, we collected all information about How to create detailed design documentation.  Then we design detailed design documentation. | 3 weeks |
| Selection Data set | First we collect all information about Siri and Alexa.  Then we collect information, how Siri and amazon Alexa work. We collected its detail documentation. | 2 weeks |
| Implementation of Proposed System | We are trying to help blind people so that they can do day to day life task easily by just commanding instructions. | 2 weeks |
| Testing of system | Tested the application on various android operating systems from windows 7 to windows 10, Ubuntu etc.  The testing was implemented using White Box and Black Box testing. | 2 weeks |
| Report Writing | We begin our report and shown to our guide for checking and improvements. | 2 weeks |
| Final Report with Modification | According to suggestion given by our guide and HOD sir we prepared our final report | 1 weeks |

Table 1.1: Timeline schedule of the Project

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Basic Study |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Review of  Papers |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Problem  Formulation |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Literature Survey |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Objectives of Topic |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Design of Mathematical  Model |  |  |  |  |  |  |  |  |  |  |  |  |  |
| No. of Weeks | W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 | W10 | W11 | W12 | W13 |
|  |  |  | | | |  | | |  | | | |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Design of Mathematical  Model |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Software requirement  And  Specification |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Detailed Design Documentation |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Selection of Data set |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Implementation on Of Proposed System |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Testing of System |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Report Writing |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Final Report with Modification |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| No. of Weeks | W14 | W15 | W16 | W17 | W18 | W19 | W20 | W21 | W22 | W23 | W24 | W25 | W26 | W27 |
|  |  | | |  | | | |  | | | |  | | |

Figure 1.1: Gantt chart of Timeline Schedule 1

Figure 1.2: Gantt chart of Timeline Schedule 2

* 1. **Problem Statement:**

Currently we found that blind people cannot easily perform day to day task on pc.so to overcome this problem designed a s/w which will listen to their voice and perform action according to it.so for them design a smart virtual assistant, which will reduce blind people works and make their task simple & faster.

# **2** **TECHNICAL KEYWORDS**

* **Python:** Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It supports functional and structured programming methods as well as OOP. It can be used as a scripting language or can be compiled to byte-code for building large applications. It provides very high-level dynamic data types and supports dynamic type checking.
* **Domain:** A domain is a field of study that defines a set of common requirements, terminology, and functionality for any software program constructed to solve a problem in the area of computer programming, known as domain engineering. The word domains also taken as a synonym of application domain.
* **Artificial Intelligence:** Artificial Intelligence (AI) refer to the simulation of human intelligences in machines that are programmed to think like humans and mimic their actions. The term may also be applied to any machine that exhibits traits associated with a human mind such as learning and problem-solving
* **Speech Engine:** Google Speech-to-Text enables developers to convert audio to text by applying powerful neural network models in an easy-to-use API. The API recognizes more than 120 languages and variants to support your global user base. You can enable voice command-and-control, transcribe audio from call centers, and more. It can process real-time streaming or prerecorded audio, using Google’s machine learning technology.
* **SDK:** A software development kit (SDK) is a collection of [software development](https://en.wikipedia.org/wiki/Software_development) tools in one installable package. They ease creation of [applications](https://en.wikipedia.org/wiki/Application_software) by having compiler, debugger and perhaps a [software framework](https://en.wikipedia.org/wiki/Software_framework). They are normally specific to a hardware platform and [operating system](https://en.wikipedia.org/wiki/Operating_system) combination. To create applications with advanced functionalities such as advertisements, push notifications,  etc. most application software developers use specific software development kits

# **3 INTRODUCTION**

* 1. **Project Idea**

The basic idea behind this project is to create a simple stand-alone application  
that helps less tech savvy people in the world to use the computer without feeling ignorant or computer illiterate. Computers have become a very important devices and as well as less expensive over time. The application works same like Siri/ Google Assistant etc. But the application deals with the computer itself mainly.

The U.I of the application is self-explanatory and minimal. Currently it takes text as input as most of the people are not very good at speaking. In this application user is able to access the services with their voice command. User can easily send an emails to the recipient available in their list as well as to the mobile number by voice command.

In this Project physically disable person or the person having less knowledge about smart phone or how to access the smart phone can easily access the phone with their voice or speech command. This application includes the functions and services such as: calling services, text message transformation, mail exchange, alarm, event handler, location services, music player service, checking weather, Google search, Wikipedia search, camera, Bluetooth headset support and help menu.

The main motive behind this software is to help end-user and make it work faster and efficient. S/w access all our system and provides output as per the request given by user.

* 1. **Motivational Survey**

We have selected this topic “Smart Virtual Assistant” because we want to help blind peoples. As blind people can’t see so they can’t interact with screen GUI. To overcome this problem we as student have tried to develop an S/W which will listen and execute query. The S/w is capable to perform all operations like convert speech to text, text to speech, send email, chatbot etc.

This S/W will help blind peoples in many scenarios & also it will save their lot of time. Smart Virtual Assistant uses Google and amazon speech reorganization engine. We see in our day to day life technology is rapidly changing so, why not we should develop s/w for blind peoples which will help them to get there work done simple and fast.

In Smart Virtual Assistant, user has to ask a question or tell a machine what task to be done then Smart virtual assistant will listen it and perform the action or execute query .Thus Smart Virtual Assistant is an AI based application.

By taking reference of all we came to know what actual requirements are, then we decided solution of that problem.

**4 PROBLEM DEFINITION AND SCOPE**

* 1. **Problem Definition**

This system developed will reduce the manual work and avoid redundant data. As repeated work will be carried out machine itself, so time will be saved. As S/W uses voice reorganization for performing task so we cans speak faster than by typing or writing. It will be helpful in various domain.

* 1. **Goal and Objective**

The goal of proposed is to improve performance and accuracy of are as follows:

* To solve all problems in current system.
* To fulfill the requirement of user and make the process faster and efficient
* To generate efficient Output based on voice input.
  1. **Technology and Associated Platform**
     1. **Hardware Specification**
* Memory Space: 100MB
* RAM: 512 MB Minimum & 2GB Maximum
* Processor: Recommended 1.5GHz or better
  + 1. **Software Specification**
* Application Libraries: Python and Web
* Language : Python
* Tools : Pycharm, Google API, Amazon API
* Operating System : Windows 7 or above

# **5 PROJECT PLANS**

* 1. **Problem Definition**

This system developed will reduce the manual work and avoid redundant data. As repeated work will be carried out machine itself, so time will be saved. As S/W uses voice reorganization for performing task so we cans speak faster than by typing or writing. It will be helpful in various domain.

* 1. **Design and Implementation Constraints**
* Displaying system information such as network usage, ram etc.
* Opening and Closing an application.
* Altering system Files, brightness, volume.
* Accessing Bluetooth, WIFI , H/W
* Bookmarking a webpage.
* Creating a new window or a tab in browser.
* The Easy Graphic Interface for ease of Usages
* Chatting like Alexa.
* Sending Mails via Voice Command
  1. **Software Requirement**
* Technology Used: Python, Google Cloud API
* Tools : Pycharm, Google & Amazon Speech Engine
* Operating System : Windows 7, Ubuntu, Linux or Above
  1. **Feasibility Study**

The feasibility study comprise of an initial investigation into personnel will be required. Feasibility study will help to make informed and transparent decisions at crucial points during the developmental process.

Feasibility has following dimensions and here is a brief description in context to present system.

* + 1. **Financial Feasibility**

The financial investment is less for creating this system. Almost all the software to be used are available on the internet. So there is no problem regarding the financial feasibility of the project.

* + 1. **Technical Feasibility**

Technical feasibility deals with the study of function, performance, and constraints like resources availability, technology, development risk that may affect the ability to achieve an acceptable system. It identifies whether the work can be done, whether technology used is compatible or not with current system. Since the anomaly detection-based service will be developed using python. Therefore, it is platform independent.

The technical issues investigated during the study are as under: -

* + - * The technology for implementation of the project system is readily available.
      * The system is capable of expansion.
      * The presented system will provide adequate accuracy and reliability
    1. **Behavioral Feasibility**

This proposed system is found to be:

* + - * Efficient.
      * Time saving
      * Accurate i.e. the accuracy of the proposed system should be good as compared to other summarization systems.
      * Scalable and reliable
    1. **Operational Feasibility**

The system provides a user friendly graphical interface and is therefore easy to use. The system is operationally feasible for implementation for the following reasons:

* The system is readily acceptable by the users since it requires no special skills for them to use the system.

# 6 **SOFTWARE REQUIREMENT AND SPECIFICATION**

* 1. **Purpose and Scope of Document**
  2. **Project Scope**

The project has a very vast scope in future. The project can be implemented on intranet in future. Project can be updated in near future as and when requirement for the same arises, as it is very flexible in terms of expansion. With the proposed software of database Space Manager ready and fully functional the client is now able to manage and hence run the entire work in a much better, accurate and error free manner. The following are the future scope for the project.

* Alexa.
* Crotona.
* Siri
  1. **Assumption and Dependencies**
* We Assume that user have basic knowledge of using App.
* This App will require & use user’s data/internet
* This App will be Accessing User’s device storage for writing data.
* This App will be accusing GPS of user for using GMap.
  1. **System Features**
     1. **System Features-I (Functional Requirements)**
* Displaying system information such as network usage, ram etc.
* It provides an easy interface to user.
* Creating a new window or a tab in browser
* Sending mails, performing Calculations, Chatbot.
  + 1. **System Features-II (Non-Functional Requirements)**
* The application is scalable and reliable.
* Time saving i.e. as automatically performs repeated task again & again.
* Accurate i.e. Takes I/P from user in form of voice and process it and generate accurate O/P
* Portable i.e. this software is portable to any system with the requirements specified.
  1. **Data Flow Diagram**

A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of processes, or information about whether processes will operate in sequence or in parallel.

* + 1. **DFD Level 0**

In DFD level 0, we see the functions Input, modules and Output. In the following DFD level 0 the input is used for registration and login into app, after that data is stores in database and validation is done and the final output is obtained.

Example:

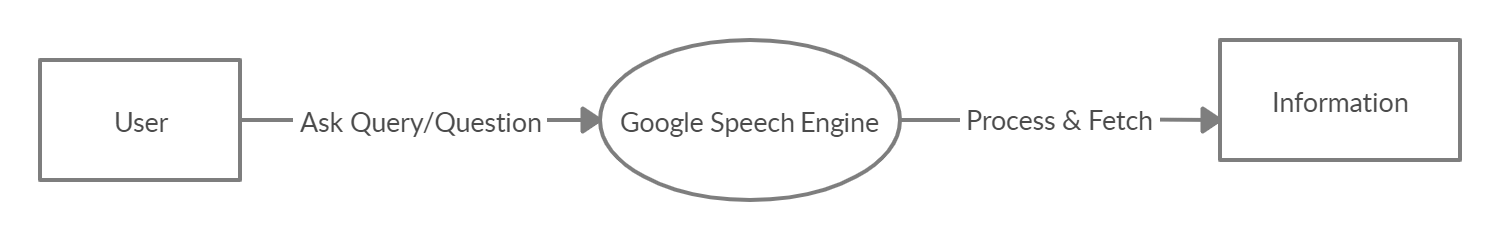


Figure 6.1: DFD Level 0

* + 1. **DFD Level 1**

In DFD level 1, it provides detailed process information. DFD level 1 shows how exact process takes place, which conditions are necessary to follow.

Example:

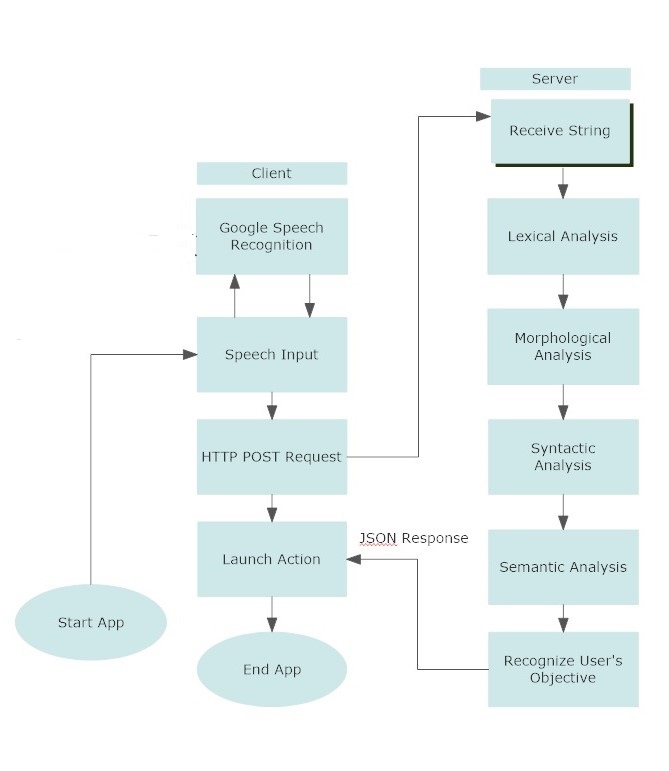


Figure 6.2: DFD Level 1

* + 1. **DFD Level 2**

In DFD level 1 provides Information about how each Module can Login, Provide Input & what it’s Output. In DFD level 2, it provides Information about how actual process takes place.

Example:

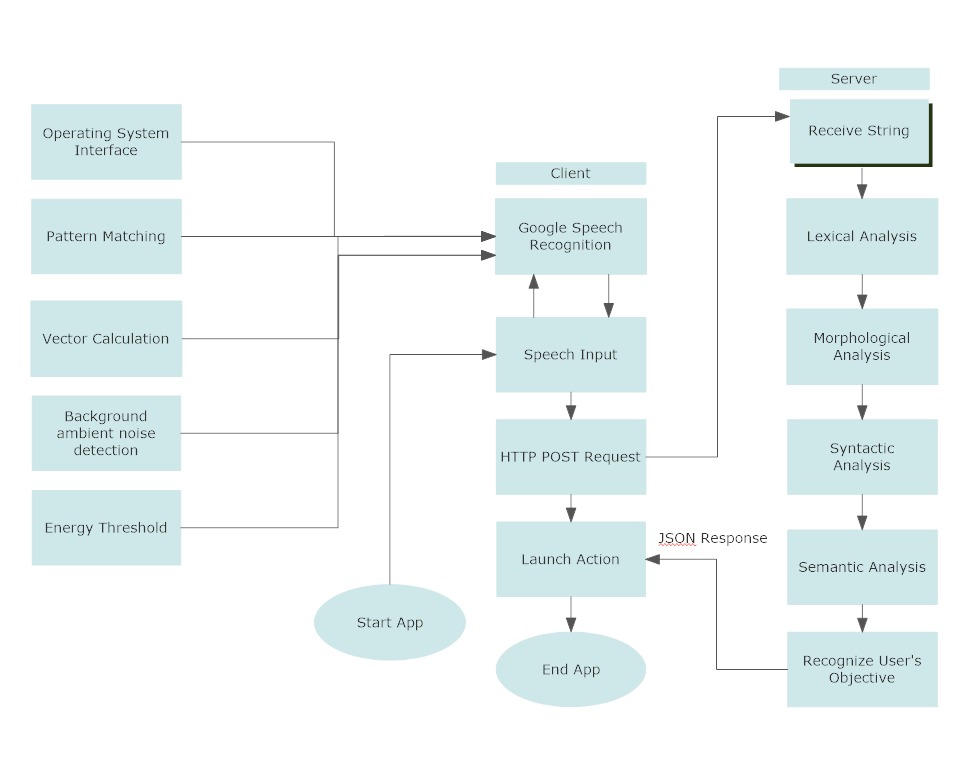


Figure 6.3: DFD Level 2

* 1. **UML Diagrams**
     1. **Use Case Diagram**

The use case diagram shows the interaction between the actor and the system. Initially Admin should have to make Login in his own Login. Then Admin Approves Player. If player wants use App then have to make Login & then search for Sport List. Admin & mediator can update, remove player from database.

To access full feature of app admin as well as player both have to login with their devices.

Example:

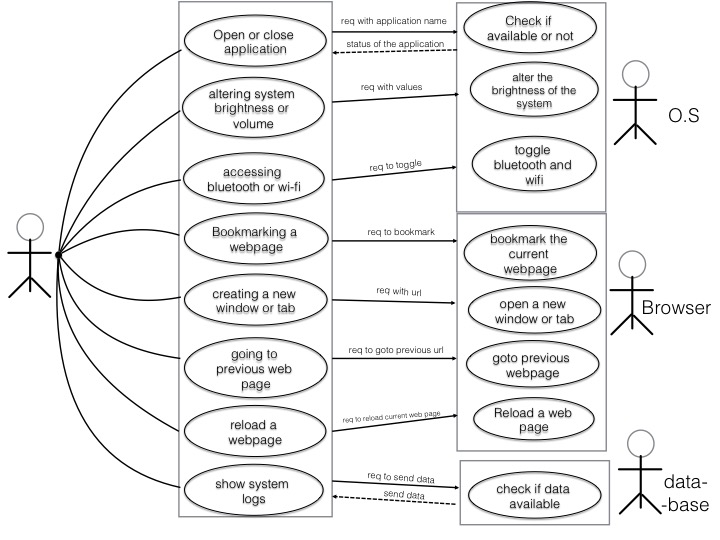


Figure 6.4: Use Case Diagram

* + 1. **Activity Diagram**

In general Activity diagrams is used to show different activities which are performed by the system. Following figure shows different activities like capturing voice, processing voice, execute voice etc.

Example:

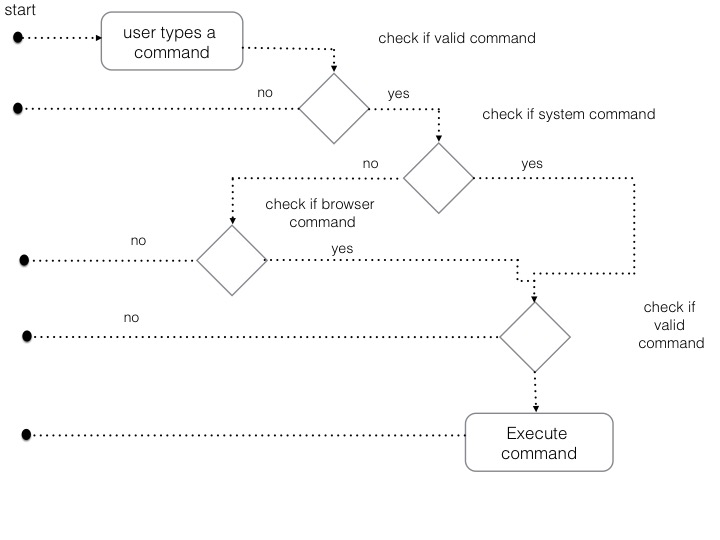


Figure 6.5: Activity Diagram

* + 1. **Sequence Diagram**

In the following sequence diagram, the step by step sequence of the proposed work is shown.

Step 1: User gives command or ask a query.

Step 2: Smart Virtual Assistant listen & send query to google speech engine.

Step 3: Google Speech Engine execute Query.

Step 4: Response/Action is performed by system.

Step 5: Stop

Example:

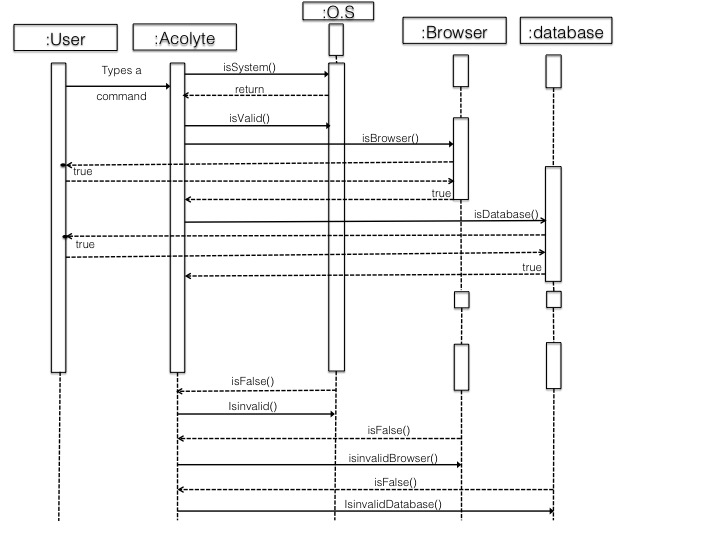


Figure 6.6: Sequence Diagram

* + 1. **Class Diagram**

In general class diagram represent the classes and functions used for the developing the proposed system. In the following class diagram the classes used are Student, Admin, and Teacher etc.

Example:

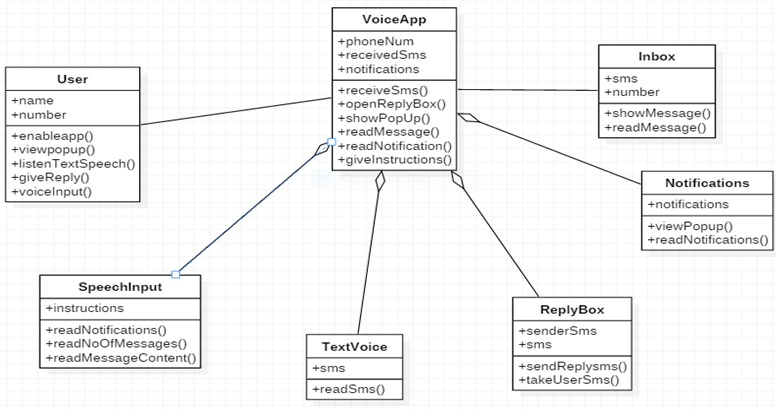
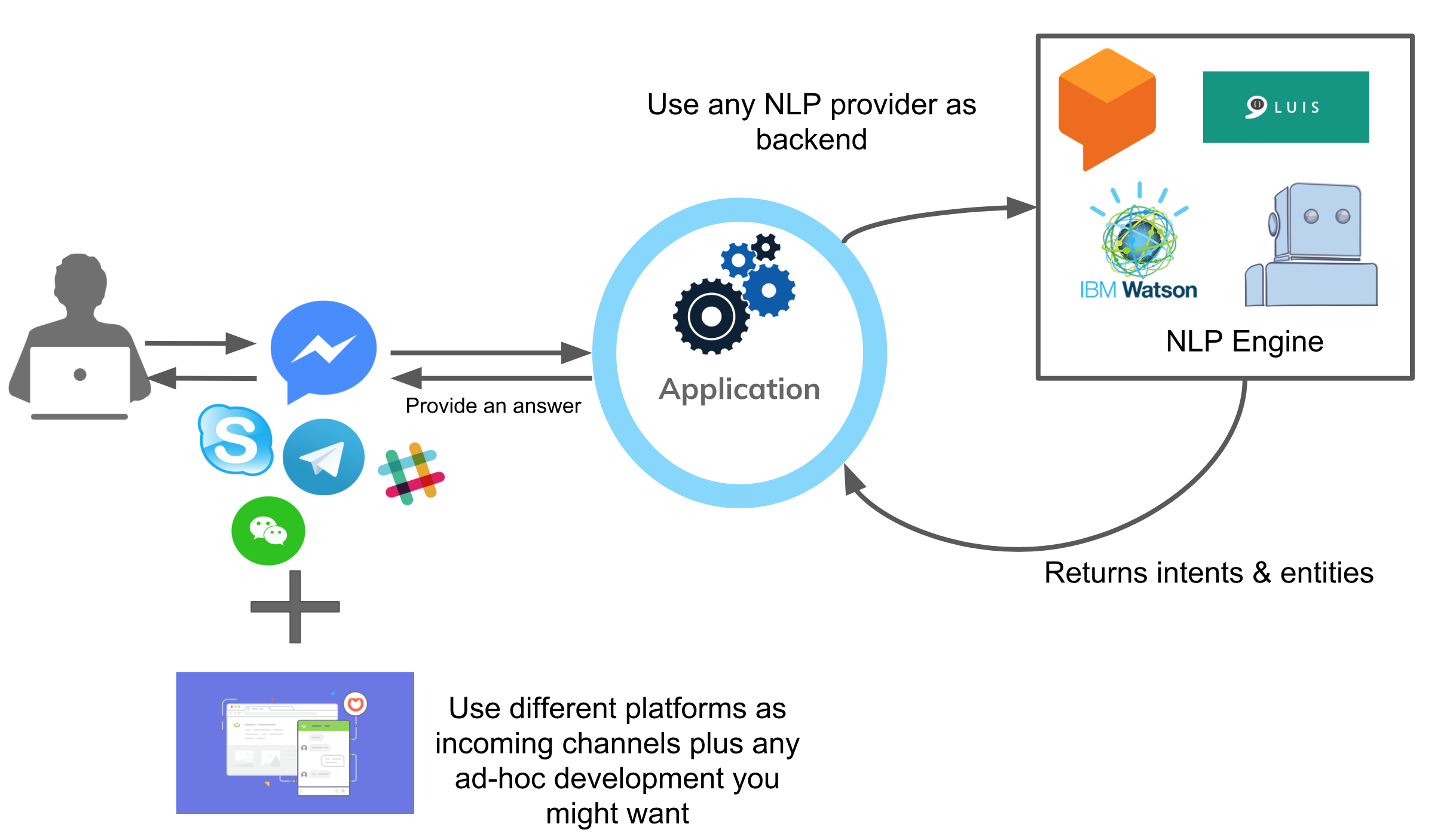


Figure 6.7: Class Diagram

# **7 DETAILED DESIGN DOCUMENTS**

**7.1 System Architecture**



Example Figure 7.1: System architecture

**System Architecture**

The system architecture shows overall functionality of App.

Firstly, user have to say ‘hello Friday!’

Once the Smart Virtual Assistant is successfully initialized and loaded properly then user can ask any query/question to SVA.

Smart Virtual Assistant captures user voice via mic and then NLP engine i.e. Natural Language Processing Engine is used to process user voice.

NLP converts user language into machine level query and then query is processed. In this project Google Speech Engine & amazon Speech Engine is used.

Google Speech Engine Process user query and bring search result from google and shown to the user in form of text, audio, video etc.

User can ask any Question to Smart Virtual Assistant and then SVM will process & generate O/P according to it. SVM can open any Windows application like opening browser, MS office, my pc etc.

As SVM does not have its own database as it uses google and amazon database. SVM take user query and send to google speech engine and then google speech engine process it and show result to user

Google and Amazon Speech API are real-time and uses internet to process user query.

User can ask SVM about Weather, chat like a friend and can do a lot of things.

All information shown into the application is complete real-time i.e. all updates are made quickly with fast response.

# **PERFORMANCE EVALUATION AND TESTING**

**Introduction**

Testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs. It describes software test environment for testing, identifies the tests to be performed, and provides schedules for test activates.

* + 1. **Goals and objectives**

The main goal to perform software testing is to validate and verify the user needs and functional specifications.

* + 1. **Statement of Scope**

The scope of software testing is to gain confidence in and providing information about the level of quality and preventing defects.

* + 1. **Major Constraints**

Constraints are the limitations, hurdles which stop the tester from fulfilling their responsibility. Constraints are the limitations, hurdles which stop the tester from fulfilling their responsibility. Here we are considering following constraints in our application development:

* + - 1. **Budget and schedule:** Budget and Schedule constraint will not allow the tester to complete test plan. In order to remove the constraint of budget and schedule first we need to understand the impact of testing on budget and schedule. As we know that cost of fixing a bug is more at the later phase. Therefore, identifying and correcting bugs at the earlier stage of development life cycle is most cost-effective way for creating it.
      2. **Incomplete statement of work:** A test objective is simply a testing a goal. ˆ It is a statement of what the test team or tester is expected to accomplish or validate during a specific testing activity. Test objectives, usually defined by the test man- ager or test team leader during requirements analysis, guide the development of test cases, test scripts, and test data. Each test objective should contain a statement of the objective, and a high-level description of the expected results stated in measurable terms. The users and project team must prioritize the test objectives. Usually the highest priority is assigned to objectives that validate high priority or high-risk requirements defined for the project. In cases where test time is cut short, test cases supporting the highest priority objectives would be executed first.
      3. **Changes in Technology:**

Effective testing must be done by a team comprised of information services professionals and users. In corporations where the users are not readily available, i.e., they are in a remote location, a professional test group can represent the users. Also, vendors of software may not be able, or may not want to have users testing their systems during the developmental process. Again, in these instances, a professional test group can represent the users. The test group is known by different names, including IT testing, quality control, quality assurance, and inspectors.

* + - 1. Randomly generated tests may not satisfy programs assumptions such as method preconditions. While constraint solving can satisfy such assumptions, it does not necessarily generate diverse tests and is hard to apply to large programs.
  1. **Testing**

Tests are the individual tests specified in a test plan document. Each test is typically described by:

* + An initial system states.
  + A set of actions to be performed.
  + The expected results of the test.
    1. **Testing as a Continuous Process**

All testing follows a pre-planned process, which is agreed to. All tests consider not only a nominal system condition but also address anomalous and recovery aspects of the system. The system is tested in a stressed environment, nominally in excess of 150 percent of its rated capacities. All test products (test cases, data, tools, configuration, and criteria) are documented in a software description document. Every test shall be described in traceable procedures and have pass-fail criteria included.

* + 1. **Implementation**

Test cases are planned in accordance to the test process and documented with detailed test descriptions. These test cases use cases based on projected operational mission scenarios. The testing process also includes stress / load testing for stability purpose (i.e., at 95% CPU use, system stability is still guaranteed). The test process thoroughly tests the interfaces and modules. Software testing includes a traceable white box testing, black box testing and other test processes verifying Implemented software against design documentation and requirements specified.

* + 1. **Testing Type**

**White Box Testing:**

A level of white box test coverage is specified that is appropriate for the software being tested. The white box and other testing uses automated tools to instrument the software to measure test coverage.

**Black Box Testing:**

A black box test of integration builds includes functional, interface, error recovery, stress and out-of-bounds input testing. All black box software tests are traced to control requirements. In addition to static requirements, a black box of a fully integrated system against scenario sequences of events is designed to model operation. Performance testing for systems is integrated as an integral part of the black box test process.

**Unit Testing:**

Unit testing is used to check the execution path of the module, function, and procedure of the system. Test is conducted with the help of normal data and abnormal data. This testing includes the different factors like statement coverage, branch coverage, loop processing, abnormality, and circulation etc. With the help of this Unit testing we check that all the statement in the code is executed or not so it avoids the dead code statement. It checks all the branches and execution path of the code. It ensures that all the internal method of program is executed and properly integrated with program.

**Integrated system:**

In integrated testing, all the modules are checked together to ensure that all the modules are executing together according to the program specification. Once all the modules have been tested individually, the most legitimate question can be asked is that when all the modules are working properly, why there is need of integrated testing. The answer is, though all modules are working properly problem may occur while interfacing individual module. Data can lose across an interface one module can have an adverse on another.

1. **Testing Tools and Environment**

**Junit:** It is a very good testing tool for using with unit testing. It is a java-based tool, which is capable of going through code especially with extreme programming as it takes up less time to do.

**QTP:** A testing tool for functional regression. It is an automated functional Graphic User Interface (GUI). The GUI is automated so actions can be made automatically online or offline. QTP has many functions such as letting you compare existing terns with current ones. QTP can test many applications such as Java, visual basic application, .net and many more.

**Coverage Tools:** Code coverage, test case coverage, test coverage and so on. Design: Calibre-RBT (No 1) uses the requirements as a basis to design the minimum number of test cases needed for full functional coverage.

**GUI:** Smalltalk Test Mentor (No 15) automatically gathers execution and method coverage metrics.

1. **Test Cases**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case Id | Test Objective | Pre-requisite | Steps | Expected result | Actual Result | Status |
| TC01 | S/W load and initialize properly. | Error free software should be load and work. | 1.run exe file | S/W should start & launch without any error. | Successfully S/W launched | Pass |
| TC02 | Network Detection and Speed | Internet is required for login and Signup | 1.Goto Application and check status | Successfully account created | Account created and logged into App | Pass |
| TC03 | Screen Sever | Application loading time | 1.Click on App  2.wait until App will start | Took 20000ms to load | App should take 70000ms to load | Fail |
| TC04 | Screen Sever | Application loading time | 1.Click on App  2.wait until App will start | Took 20000ms to load | App taking 20000ms to load | Pass |
| TC05 | Properly mic has been connected and is working. | System mic should be working and connected to S/W during initialization time. | 1.Start S/W  2. After initialized say hello Friday!  3. System should respond to hey Friday! | System should respond to hey Friday! | System successfully respond to hey Friday! | Pass |
|  |  |  |  |  |  |  |

Table 8.1: Test Case

* + **RESULT AND ANALYSIS**

1. **Dataset Description**

For our experiments, a set of research article are collected directly from the Web. The structure is similar for all of the articles in the corpus: title, authors and affiliations, keywords, abstract, outline and a variable number of sections (the content of these section is what we considered as the text of the article), depending on each article. At the end of each one, there are also two special sections: one is the take-home messages and the last one contains the references. In some articles, an acknowledgement section is included as well. In addition, each article may contain figures and tables, which are not taken into consideration for generating the summaries.

It is worth mentioning that for generating the summaries neither the keywords of the original article nor the information in the titles or in the abstract have been taken into consideration. Moreover, before passing the articles through system, the data set is cleaned and only the main content of each article is kept for further processing. In other words, the outline, bibliographic entries, keywords, figures and tables are removed.

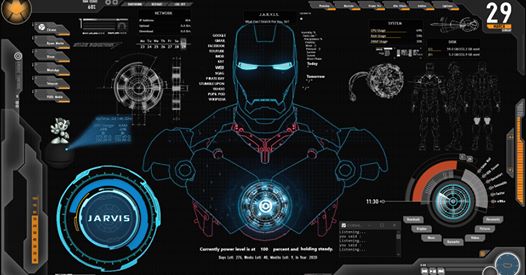
9.2 **Experimental Setup**

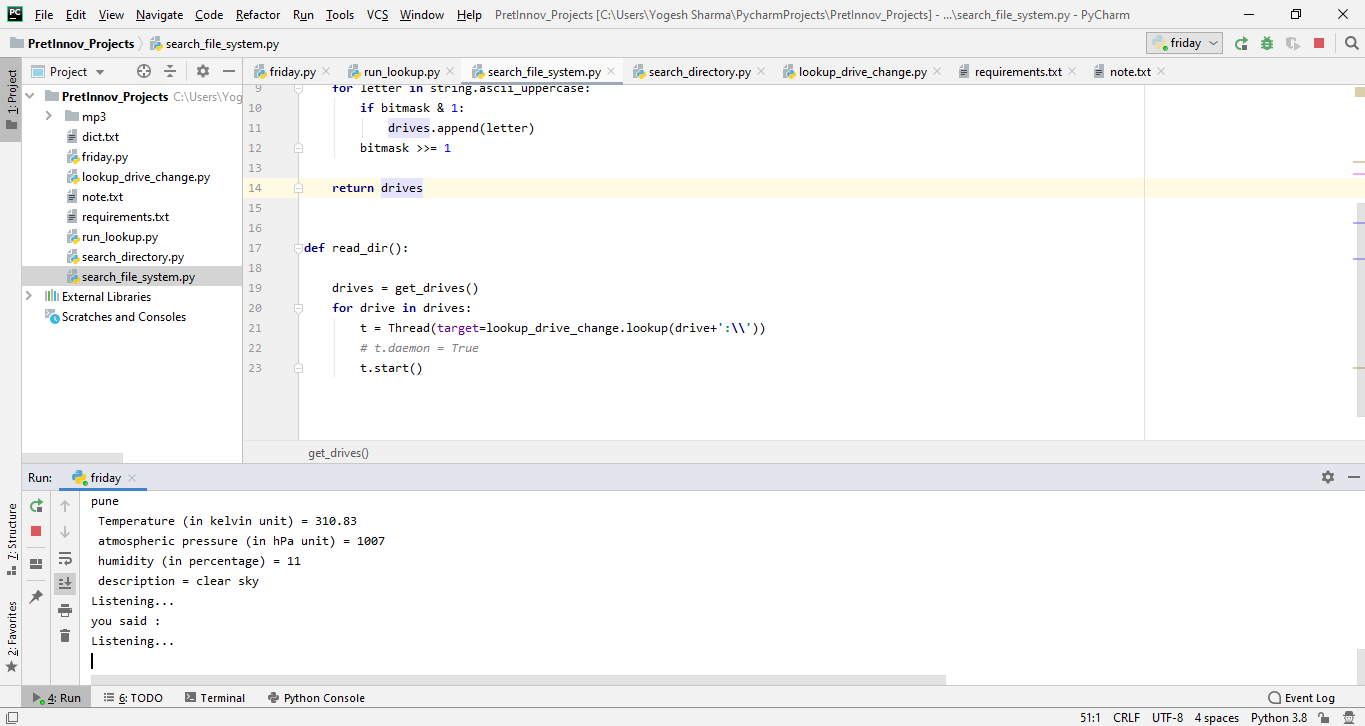
The system is built using python (version 3.8.2) on Windows platform. The Pycharm (version 3.4) & Google API is used as development tool. The system does require mic as hardware to run. As any standard machine is capable of running the application.

**ANNEXURE A: PROJECT SCREENSHOT**

**GUI – S/W**

Figure A: S/W GUI





**Project Source Code**

**Python Code for Speech:**

import ast

import os

from watchdog.events import PatternMatchingEventHandler

from watchdog.observers import Observer

import search\_directory

lookup\_dict = search\_directory.file\_system\_dict

class ExampleHandler(PatternMatchingEventHandler):

@staticmethod

def get\_dup\_key(key, keys):

count = 0

for k in keys:

if k in key:

count += 1

return count

@staticmethod

def get\_event(self, event\_name, method):

src = event\_name.src\_path

key = src.rsplit('\\')[-1].lower()

if method == 'on\_created':

lookup\_dict.update({key + '\_{}'.format(

ExampleHandler.get\_dup\_key(key, lookup\_dict.keys())): src})

print('KEY ADDED')

print(key)

elif method == 'on\_deleted':

key\_list = lookup\_dict.keys()

try:

for key in list(key\_list):

if lookup\_dict.get(key) == src:

del lookup\_dict[key]

print('KEY DELETED')

print(key)

except KeyError:

pass

elif method == 'on\_moved' or method=='on\_modified':

key\_list = lookup\_dict.keys()

try:

for key in list(key\_list):

if lookup\_dict.get(key) == src:

del lookup\_dict[key]

print('KEY MOVED')

print(key)

lookup\_dict.update({key + '\_{}'.format(

ExampleHandler.get\_dup\_key(key, lookup\_dict.keys())): src})

dir\_txt = {}

file\_name='dict.txt'

if os.path.exists(file\_name):

with open(file\_name, 'r') as f:

h = ast.literal\_eval(f.read())

dir\_txt.update(h)

f.close()

with open(file\_name, 'w') as f:

dir\_txt.update({key + '\_{}'.format(

ExampleHandler.get\_dup\_key(key, lookup\_dict.keys())): src})

f.write(str(dir\_txt))

f.close()

print('NEW KEY ADDED')

print(key)

except KeyError:

pass

def on\_created(self, event): # when file is created

ExampleHandler.get\_event(self, event, 'on\_created')

def on\_modified(self, event):

ExampleHandler.get\_event(self, event, 'on\_modified')

def on\_deleted(self, event):

ExampleHandler.get\_event(self, event, 'on\_deleted')

def on\_moved(self, event):

ExampleHandler.get\_event(self, event, 'on\_moved')

# search\_directory.find()

def lookup(drive):

observer = Observer()

event\_handler = ExampleHandler(ignore\_patterns=['\*.tmp','\*AppData\*','\*Temp\*','\*$\*','\*ProgramData\*','\*\_\_\*']) # create event handler

# set observer to use created handler in directory

observer.schedule(event\_handler, path='C:\\Program Files (x86)\\', recursive=True)

observer.start()

observer.join()

**Conclusion**

From my perspective, Chatbots or smart assistants with artificial intelligence are dramatically changing businesses. There is a wide range of chatbot building platforms that are available for various enterprises, such as e-commerce, retail, banking, leisure, travel, healthcare, and so on.

Chatbots can reach out to a large audience on messaging apps and be more effective than humans. They may develop into a capable information-gathering tool in the near future.

**REFERENCES**

**List of Websites: -**

* **Python: - https://www.python.org/**
* **W3Schools : - https://www.w3schools.com/**
* **YouTube: - www.youtube.com**
* **Tutorials Point: - www.tutorialspoint.com**
* **Google Speech Engine :- https://cloud.google.com/speech-to-text**
* **Amazon Speech Engine :- https://aws.amazon.com/polly/**

**List of Articles Referred: -**

[1] Prof.Rakhi Bhardwaj, Poonam Gupta, PoojaJadhav, BhagyashreeKadam, AmrutaKedari,

"Android Based Automated Smart Wheel Chair", International Journal of Innovative

Research in Computer and Communication Engineering, vol 4, Issue 3, March 2016.

[2] Kiran N.V, Madhan Mohan Reddy, Deepika Yadav, Babitha K.M, "Optimal driving system

for two wheelers", Proceedings of IRF International Conference, 05th April-2014

[3] Yu Zhong, T. V. Raman, Casey Burkhardt, FadiBiadsy and Jerey P. Bigham, "JustSpeak:

Enabling Universal Voice Control on Android", April 2014

[4] NorhazahbtAripin, M. B. Othman, "Voice control of home appliances using Android",

Electrical Power, Electronics, Communications, Controls, and Informatics Seminar

(EECCIS), 2014

[5] ArunGopi, Shobana Devi P, Sajini T, Bhadran V K, "Implementation of Malayalam Text to

Speech Using Concatenative Based TTS for Android Platform",

[6]Dr.Mohd Maqsood Ali,”Role of data mining in education sector,” International Jour- nal of Computer Science and Mobile Computing(IJCSMC),vol 2,issue 4,April 2013

[7]Suchita Borkar K. Rajeswari, “Python with Google aPI”, IJCSMC, Vol. 2, Issue. 7, July 2013, pg.273 – 279