

CHAPTER 1: INTRODUCTION

1.1 Background:

A **voice assistant** or **intelligent personal assistant** is a software agent that can perform tasks or services for an individual based on verbal commands i.e. by interpreting human speech and respond via synthesized voices.

Users can ask their assistants' questions, control home automation devices, and media playback via voice, and manage other basic tasks such as email, to-do lists, open or close any application etc with verbal commands.

Who doesn't want to have the luxury to own an assistant who always listens for your call, anticipates your every need, and takes action when necessary? That luxury is now available thanks to artificial intelligence-based voice assistants.

Voice assistants come in somewhat small packages and can perform a variety of actions after hearing your command. They can answer questions, play music, place online orders and do all kinds of AI-based stuff.

Voice assistants are not to be confused with virtual assistants, which are people who work remotely and can, therefore, handle all kinds of tasks. Rather, voice assistants are technology based. As voice assistants become more robust, their utility in both the personal and business realms will grow as well.

1.2 Feasibility study:

Feasibility study is conducted once the problem is understood. It also quantifies benefits and cost. Feasibility study is the study which involves the analysis of problem domain and collection of all vital information in relation to the product to select the best system that meets the requirement. It is the important phase of software development process.

The main objective of feasibility study are:

- To analyze if software meets organizational requirements
- To determine if software can be implemented using the current technology.
- To find whether the software can be integrated with other existing software.

Basically the feasibility study can be performed using



0



0

1.2.1 Technical feasibility study:

Technical feasibility is the process of validating the technology assumption, architecture and design of project. In this, the organization has to decide which technologies are suitable to develop our project. The technology involved in this project are:

- Tkinter as front end.
- Pyttsx3, Smtplib, Speech recognition as back end.

Each of this technology are freely available and compatible on device

1.2.2 Operational feasibility

Operational feasibility study is concerned with issues like whether the system will be used, if it is developed or implemented. It also focuses on whether the proposed project fits with existing business environment. It is used to identify the problem and how it is to be solved. The proposed system is operationally feasible due to following reason:

- It is convenient, flexible and easy to use.
- It saves time and effort.
- Can work with variety of commands
- Secure and Artificial intelligent

1.2.2 Economic feasibility study:

In economic feasibility cost benefit analysis is done in which expected cost and benefits are evaluated. Economic analysis is used for evaluating the effectiveness of proposed system. The project is being developed by freely available software so no cost of development is needed. The cost of human efforts and maintenance are taken into consideration

1.3 Objective:

- JARVIS is a personal desktop assistant that keeps users informed and productive, helping them get things done across devices and platforms.
- Skills define the tasks that jarvis can accomplish. You can extend jarvis by adding your own skills that let your users interact with your service via jarvis.
- jarvis invokes the skills based on input from the user, spoken.
- Because jarvis can be used in laptop or pc, some may have a screen, while others may have a speaker, and some may have both.



0



0

- You should ensure that your bot is capable of handling any of these devices. The bot framework provides device information to your skill

1.4 Scope:

- There is an increased overall awareness and a higher level of comfort demonstrated specifically by millennial consumers.
- In this ever-evolving digital world where speed, efficiency, and convenience are constantly being optimized.
- The mass adoption of artificial intelligence in users' everyday lives is also fueling the shift towards voice applications.
- The number of IoT devices such as smart thermostats, appliances, and speakers are giving voice assistants more utility in a connected user's life.

1.5 Advantages:

- Easy to use
- Can work with variety of commands
- Custom commands
- Secure
- Artificial intelligent
- Single code base

1.6 Purpose:



0



0

A voice assistant is a digital assistant that uses voice recognition, natural language processing and speech synthesis to provide aid to users through desktop and voice recognition. Voice assistants are built on artificial intelligence (AI), machine learning and voice recognition technology. As the end user interacts with the digital assistant, the AI programming uses sophisticated algorithms to learn from data input and better itself at predicting the user's needs. Some assistants are built with more advanced cognitive computing technologies which will allow a digital assistant to understand and carry out multi-step requests with numerous interactions and perform more tasks

Digital assistants can be contrasted with another application of consumer-facing AI called smart advisors. Smart advisor programs are knowledge-oriented, while digital assistants are task-oriented, although some perform both roles. Popular voice assistants currently include Apple's Siri, Amazon's Alexa, Google Now, Google Assistant and Microsoft's Cortana.

1.7 Achievements:

The result of this project is an effective and efficient system that will help the user to save the data with the advantage of data backup. It makes the working a lot more easier. It has made the management system a lot more convenient than before. It also helps us to reduce the use of paper. It is also user- friendly. It provides access to data easily and very quick. It helps us to save time, efforts and resources.



0



0

CHAPTER 2: SURVEY OF TECHNOLOGIES

PYTHON:

What is Python?

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

It is used for:

- web development (server-side),
- software development,
- mathematics,
- system scripting.

What can Python do?

- Python can be used on a server to create web applications.
- Python can be used alongside software to create workflows.
- Python can connect to database systems. It can also read and modify files.
- Python can be used to handle big data and perform complex mathematics.
- Python can be used for rapid prototyping, or for production-ready software development.

Why Python?

- Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
- Python has a simple syntax similar to the English language.
- Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
- Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.
- Python can be treated in a procedural way, an object-orientated way or a functional way.

Good to know

- The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.



0



0

- In this tutorial Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse which are particularly useful when managing larger collections of Python files.

Python Syntax compared to other programming languages

- Python was designed for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.



0



0

CHAPTER 3: REQUIREMENT AND ANALYSIS

3.1 Problem Definition:

Mostly all renting system is done manually therefore it is not effective. More problem will aroused by managing manually. From registration until check out problems exist. It is difficult for a admin that wants to recall a customer history & his preference during customer next encounter. Paperwork is less secure & unreliable. Customer feedback is not provided in manual work which helps in decisioning to customer which one to opt for.

3.2 Problem Specification:

The admin has to login into the system.

1. Provide vehicle catalogue for users as an alternative for them to select vehicle if they want to choose vehicle by their own.
2. Admin can search user information from the database based on the userID card number or their name.
3. Admin can add a new vehicle.
4. Admin can manage booking of vehicles.
5. Admin can give vehicle on rent to the proper and legal customers.
6. Allow the user to view feedback and enquires posted by users.

3.3 Planning and scheduling:

Planning:

Planning is the process of deciding in detail how to do something before you actually start to do it. It is a vital part of project management process. SDLC:

SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

The following figure is a graphical representation of the various stages of a typical SDLC.



0



0

A typical Software Development Life Cycle consists of the following stages –

Requirement Analysis Stage 1: Planning and Requirement analysis is the most important and fundamental stage in SDLC. It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry.

Stage 2: Defining Requirements

Once the requirement analysis is done the next step is to clearly define and document the product requirements and get them approved from the customer or the market analysts. This is done through an SRS (Software Requirement Specification) document which consists of all the product requirements to be designed and developed during the project life cycle.

Stage 3: Designing the Product Architecture

SRS is the reference for product architects to come out with the best architecture for the product to be developed. Based on the requirements specified in SRS, usually more than one design approach for the product architecture is proposed and documented in a DDS - Design Document Specification.

Stage 4: Building or Developing the Product

In this stage of SDLC the actual development starts and the product is built. The programming code is generated as per DDS during this stage. If the design is performed in a detailed and organized manner, code generation can be accomplished without much hassle.

Developers must follow the coding guidelines defined by their organization and programming tools like compilers, interpreters, debuggers, etc. are used to generate the code. Different high level programming languages such as C, C++, Pascal, Java and PHP are used for coding. The programming language is chosen with respect to the type of software being developed.

Stage 5: Testing the Product

This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However, this stage refers to the testing only stage of the product where product defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.

Stage 6: Deployment in the Market and Maintenance

Once the product is tested and ready to be deployed it is released formally in the appropriate market.

Then based on the feedback, the product may be released as it is or with suggested enhancements in the targeting market segment. After the product is released in the market, its maintenance is done for the existing customer base.

SDLC Models



0



0

There are various software development life cycle models defined and designed which are followed during the software development process. These models are also referred as Software Development Process Models". Each process model follows a Series of steps unique to its type to ensure success in the process of software development.

Following are the most important and popular SDLC models followed in the industry –

- **Waterfall Model**
- **Iterative Model**
- **Spiral Model**
- **V-Model**
- **Big Bang Model**

Other related methodologies are Agile Model, RAD Model, Rapid Application Development and Prototyping Models.

Waterfall Model:

The Waterfall Model was the first Process Model to be introduced. It is also referred to as a linear-sequential life cycle model. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.

The Waterfall model is the earliest SDLC approach that was used for software development.

The waterfall Model illustrates the software development process in a linear sequential flow. This means that any phase in the development process begins only if the previous phase is complete. In this waterfall model, the phases do not overlap.

Waterfall Model - Design

Waterfall approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially.

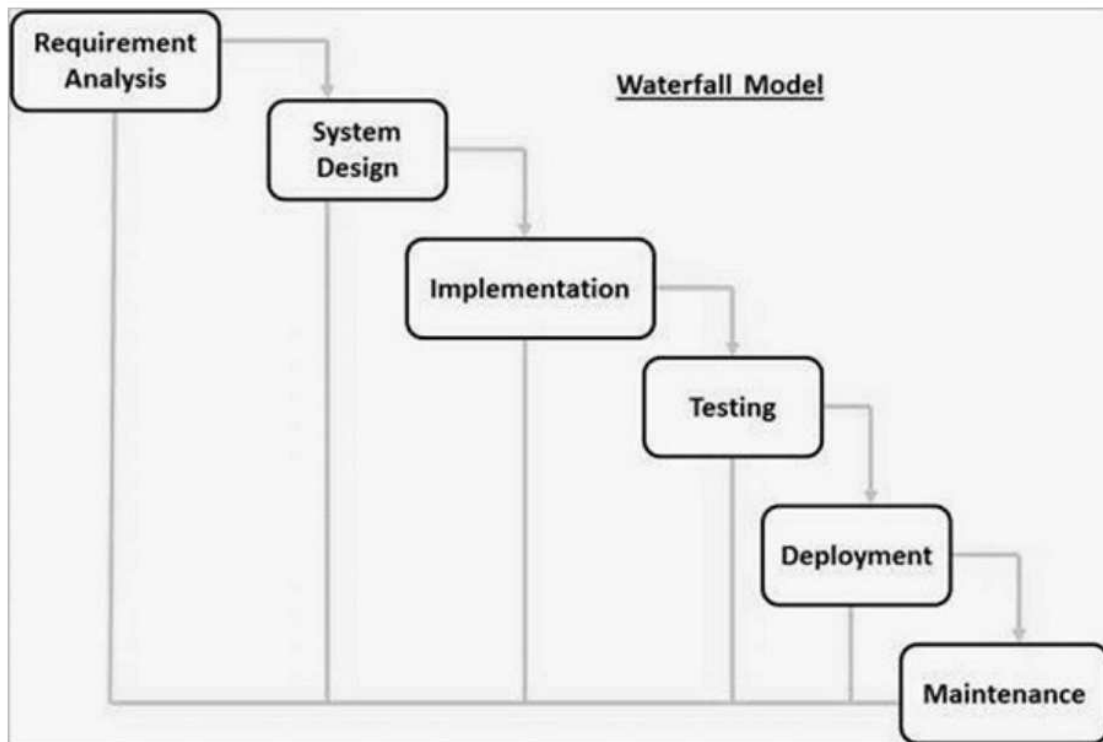
The following illustration is a representation of the different phases of the Waterfall Model.



0



0



The sequential phases in Waterfall model are –

- **Requirement Gathering and analysis** – All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.
- **System Design** – The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
- **Implementation** – With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing.
- **Integration and Testing** – All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
- **Deployment of system** – Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.
- **Maintenance** – There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

Waterfall Model - Application

Some situations where the use of Waterfall model is most appropriate are –

- Requirements are very well documented, clear and fixed.
- Product definition is stable.
- Technology is understood and is not dynamic.
- There are no ambiguous requirements.
- Ample resources with required expertise are available to support the product.
- The project is short.

Waterfall Model - Advantages

The advantages of waterfall development are that it allows for departmentalization and control. A schedule can be set with deadlines for each stage of development and a product can proceed through the development process model phases one by one.

Development moves from concept, through design, implementation, testing, installation, troubleshooting, and ends up at operation and maintenance. Each phase of development proceeds in strict order.

Some of the major advantages of the Waterfall Model are as follows –

- Simple and easy to understand and use
- Phases are processed and completed one at a time.
- Works well for smaller projects where requirements are very well understood.

Waterfall Model - Disadvantages

The disadvantage of waterfall development is that it does not allow much reflection or revision. Once an application is in the testing stage, it is very difficult to go back and change something that was not well-documented or thought upon in the concept stage.

The major disadvantages of the Waterfall Model are as follows –

- No working software is produced until late during the life cycle.
- High amounts of risk and uncertainty.
- Not a good model for complex and object-oriented projects.
- Poor model for long and ongoing projects.
- Cannot accommodate changing requirements.
- Integration is done as a "big-bang" at the very end, which doesn't allow identifying any technological or business bottleneck or challenges early.



0



0

3.4 Hardware and Software Requirements:

This section lists the minimum hardware and software requirements needed to run system efficiently.

➤ **Hardware Requirements:**

- Processor minimum i3 required.
- Hard Disk: Minimum 250GB HDD
- CPU Type: 1.6GHz or faster processor
- RAM: Minimum 1 GB
- Internet Connection

➤ **Software Requirements:**

- Microsoft windows operating system.
- Python 3.7.4
- Visual studio code
- Pycharm
- OS: WINDOWS 8, 10



0



0

CHAPTER 4: SYSTEM DESIGN

4.1 Basic Modules

There following modules in Desktop Voice Assistant.

Pyttsx3

pyttsx3 is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline, and is compatible with both Python 2 and 3.

Speech Recognition

Speech Recognition is an important feature in several applications used such as home automation, artificial intelligence, etc. This article aims to provide an introduction on how to make use of the SpeechRecognition library of Python

Date Time

The datetime module supplies classes for manipulating dates and times in both simple and complex ways. While date and time arithmetic is supported, the focus of the implementation is on efficient attribute extraction for output formatting and manipulation. For related functionality, see also the time and calendar modules.

Wikipedia

Wikipedia is a Python library that makes it easy to access and parse data from Wikipedia. Search Wikipedia, get article summaries, get data like links and images from a page, and more. Wikipedia wraps the MediaWiki API so you can focus on using Wikipedia data, not getting it.

Web browser



0



0

The webbrowser module provides a high-level interface to allow displaying Web-based documents to users. Under most circumstances, simply calling the `open()` function from this module will do the right thing.

Os

This module provides a portable way of using operating system dependent functionality. If you just want to read or write a file see `open()`, if you want to manipulate paths, see the `os.path` module, and if you want to read all the lines in all the files on the command line see the `fileinput` module. For creating temporary files and directories see the `tempfile` module, and for high-level file and directory handling see the `shutil` module.

Smtplib

The `smtplib` module defines an SMTP client session object that can be used to send mail to any Internet machine with an SMTP or ESMTP listener daemon. For details of SMTP and ESMTP operation, consult **RFC 821** (Simple Mail Transfer Protocol) and **RFC 1869** (SMTP Service Extensions).

Json

JSON (JavaScript Object Notation), specified by RFC 7159 (which obsoletes RFC 4627) and by ECMA-404, is a lightweight data interchange format inspired by JavaScript object literal syntax (although it is not a strict subset of JavaScript 1). `json` exposes an API familiar to users of the standard library `marshal` and `pickle` modules.