NLP Chatbot Development using Dialogflow

**Software Requirements Specification**

Version 1.0



**Group Id:** F24PROJECTB3B8C

**Supervisor Name :***Abdullah Qamar*

|  |  |  |  |
| --- | --- | --- | --- |
| **Date (dd/mm/yyyy)** | **Version** | **Description** | **Author** |
| 19/11/2024 | 1.0 | The purpose of this project is to develop an AI-powered restaurant chatbot using **Google Dialogflow**. This chatbot aims to enhance customer interaction and streamline restaurant operations by automating key tasks such as **Make reservations, menu navigation ,order-taking,**  and **customer support**. | **BC210414048** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

1. [Scope (of the project)](#scope)
2. [Functional Requirements Non Functional requirements](#FRNFR)
3. [Use Case Diagram](#UCD)
4. [Usage Scen](#UCS)[arios](#UCS)
5. [Adopted Methodology](#Adopted)

[6.](#UCS) [Work Plan (Use MS Project to create Schedule/Work Plan)](#Gantt)

[SRS Document](#UCS)

## [1.Scope of the Project](#UCS)

[The project involves developing an](#UCS) **[AI-powered restaurant chatbot](#UCS)** [using](#UCS) **[Google Dialogflow](#UCS)** [to automate key customer interactions. The chatbot will focus on tasks such as managing reservations, navigating menus, placing orders, and answering customer queries. It will provide a text-based conversational interface accessible via the](#UCS) **[web platform](#UCS)**[.](#UCS)

**[Purpose](#UCS)**

[The purpose of the project is to:](#UCS)

* [Automate routine customer service tasks.](#UCS)
* [Enhance customer experience through quick and accurate responses.](#UCS)
* [Reduce the workload on restaurant staff by handling repetitive inquiries.](#UCS)

**[Intended Functionalities](#UCS)**

* [Table Reservations](#UCS)
* [Menu Navigation](#UCS)
* [Order Placement](#UCS)
* [Customer Support](#UCS)

### ****[2.Requirements](#UCS)****

### ****[2.1 Functional Requirements](#UCS)****

[These are the key features and functionalities the chatbot will perform:](#UCS)

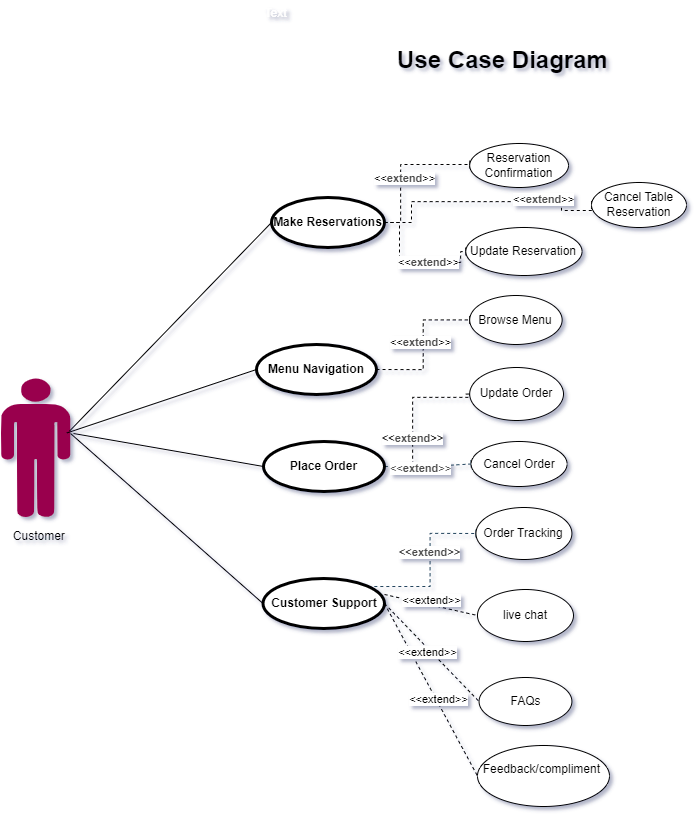
* **[Make Reservations](#UCS)**
  1. [Customers can check table availability and reserve a table by providing date, time, and number of guests.](#UCS)
  2. [The system will confirm the reservation or suggest an alternative if unavailable.](#UCS)
* **[Menu Navigation](#UCS)**
  1. [Display the restaurant menu for customers.](#UCS)
  2. [Allow users to filter menu items by category (e.g., desserts, beverages) or preferences (e.g., vegan, gluten-free).](#UCS)
* **[Order Placement](#UCS)**
  1. [Customers can place orders through the chatbot for dine-in or takeaway.](#UCS)
  2. [The system will confirm the order and show a summary before submission.](#UCS)
* **[Customer Support](#UCS)**
  1. [Answer common questions about restaurant hours, location, and services.](#UCS)
  2. [Forward complex issues to human staff if required.](#UCS)

### ****[2.2 Non-Functional Requirements](#UCS)****

[These focus on how the system will perform and its quality attributes:](#UCS)

* **[Performance](#UCS)**
  1. **[Requirement](#UCS)**[: The chatbot should respond quickly to user inputs (2-3 seconds maximum).](#UCS)
  2. **[Activity](#UCS)**[: Use fast and reliable hosting servers; optimize queries and responses in Dialogflow. Conduct load testing to ensure the system performs well under high traffic.](#UCS)
* **[Scalability](#UCS)**
  1. **[Requirement](#UCS)**[: Handle multiple customer interactions simultaneously, especially during busy hours.](#UCS)
  2. **[Activity](#UCS)**[: Implement cloud hosting (e.g., Google Cloud) to dynamically scale server resources during peak times.](#UCS)
* **[Usability](#UCS)**
  1. **[Requirement](#UCS)**[: The chatbot must be simple and easy to use for all users.](#UCS)
  2. **[Activity](#UCS)**[: Perform user testing to identify areas of improvement in the design and flow. Provide clear instructions and error messages during interactions.](#UCS)
* **[Security](#UCS)**
  1. **[Requirement](#UCS)**[: Protect customer data and ensure secure transactions.](#UCS)
  2. **[Activity](#UCS)**[: Use HTTPS for secure communication, encrypt sensitive data, and regularly update the system to patch vulnerabilities.](#UCS)
* **[Accessibility](#UCS)**
  1. **[Requirement](#UCS)**[: The chatbot should work seamlessly on desktops, tablets, and smartphones.](#UCS)
  2. **[Activity](#UCS)**[: Test the chatbot on multiple devices and browsers to ensure compatibility. Adjust the layout for proper viewing on small screens.](#UCS)

### ****[3. Use Case Diagram](#UCS)****

[](#UCS)

**[4.](#UCS)**[Usage Scenarios](#UCS)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **[Use Case Title](#UCS)** | **[Use Case ID](#UCS)** | **[Actions](#UCS)** | **[Description](#UCS)** | **[Alternative Paths](#UCS)** | **[Pre-Condition](#UCS)** | **[Post-Condition](#UCS)** | **[Author](#UCS)** | **[Exceptions](#UCS)** |
| **[Make Reservation](#UCS)** | [UC001](#UCS) | [1. Customer selects "Make Reservation."  2. Customer provides details (date, time, guests).  3. Chatbot confirms reservation.  4. Intents: Reservation Reminder, Update Reservation, Cancel Reservation.](#UCS) | [Allows customers to create, modify, or cancel table reservations easily.](#UCS) | [- If modification intent is selected, customer updates reservation details. - If cancel intent is selected, reservation is removed.](#UCS) | [Valid date, time, and guest information. System supports reservation processing.](#UCS) | [Reservation is successfully made, updated, or canceled.](#UCS) | [bc210414048](#UCS) | [Invalid or incomplete details provided. No available slots for the selected date/time. Reservation system error.](#UCS) |
| **[Menu Navigation](#UCS)** | [UC002](#UCS) | [1. Customer selects "Menu Navigation."  2. Customer uses Explore intent to browse menu categories or search for specific items.](#UCS) | [Enables customers to interactively browse the menu or search for specific items.](#UCS) | [- If no input is provided, chatbot displays all menu categories. - Recommendations can be shown based on past orders or preferences.](#UCS) | [Menu categories and items are pre-defined and available.](#UCS) | [Menu is displayed, or relevant search results are provided to the customer.](#UCS) | [bc210414048](#UCS) | [Menu data is unavailable or outdated. Search query fails to match menu items. System fails to retrieve menu data.](#UCS) |
| **[Place Order](#UCS)** | [UC003](#UCS) | [1. Customer selects "Place Order."  2. Customer uses Modification intent to:   - Add items to the order.   - Remove items from the order. 3. Chatbot calculates the total.  4. Customer confirms or cancels the order.](#UCS) | [Allows customers to add, remove, or modify items in their order. Consolidates these actions into a single Modification intent.](#UCS) | [- If an item is unavailable, chatbot suggests alternatives. - If the customer cancels, the order is cleared.](#UCS) | [Items are available and listed in the menu. System supports modifications and cancellation.](#UCS) | [Order is successfully placed, modified, or canceled. Order confirmation is shared with the customer.](#UCS) | [bc210414048](#UCS) | [Invalid input for order details. Items unavailable or out of stock. System fails to calculate or confirm the order.](#UCS) |
| **[Customer Support](#UCS)** | [UC004](#UCS) | [1. Customer selects "Customer Support."  2. Customer interacts with intents: Track Order, Live Chat, FAQs, Feedback. 3. Chatbot resolves or escalates the issue.](#UCS) | [Provides assistance to customers by resolving queries, tracking orders, answering FAQs, collecting feedback, or escalating to live chat support for unresolved issues.](#UCS) | [- If chatbot cannot resolve an issue, it connects to live support. - If Track Order is selected, chatbot provides current order status. - Feedback intent collects suggestions or complaints.](#UCS) | [Customer has a query or issue requiring support. Chatbot can access order and feedback data.](#UCS) | [Customer issue is resolved. Feedback is recorded. Live chat is initiated for unresolved queries.](#UCS) | [bc210414048](#UCS) | [Order tracking fails due to incomplete data. No agents available for live chat. Feedback submission fails. FAQs are outdated.](#UCS) |

### ****[5. Adopted Methodology](#UCS)****

### ****[Adopted Methodology: VU Process Model](#UCS)****

[For the development of the restaurant-based chatbot system, we have chosen the](#UCS) **[VU Process Model](#UCS)**[, which is a combination of the](#UCS) **[Waterfall](#UCS)** [and](#UCS) **[Spiral](#UCS)** [models. This methodology provides the structure of the Waterfall model while incorporating the flexibility and iterative nature of the Spiral model, making it suitable for projects that require gradual refinement and risk mitigation.](#UCS)

### [Water fall model:](#UCS)

[](#UCS)

[The waterfall model was 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. 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; hence it is also referred to as a linear-sequential life cycle model. This means that any phase in the development process begins only if the previous phase is complete. In waterfall model phases do not overlap. Water fall model have different stages which is starting from requirements phase and ends at acceptance phase.](#UCS)

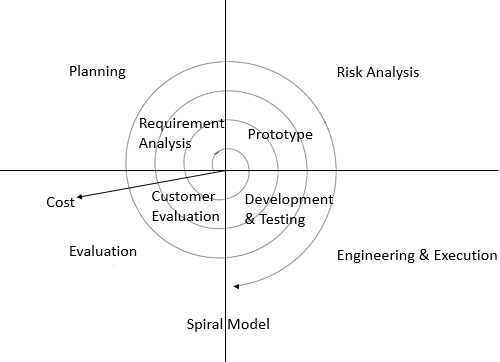
**[Now we will discuss the spiral model](#UCS)**

### [Spiral model:](#UCS)

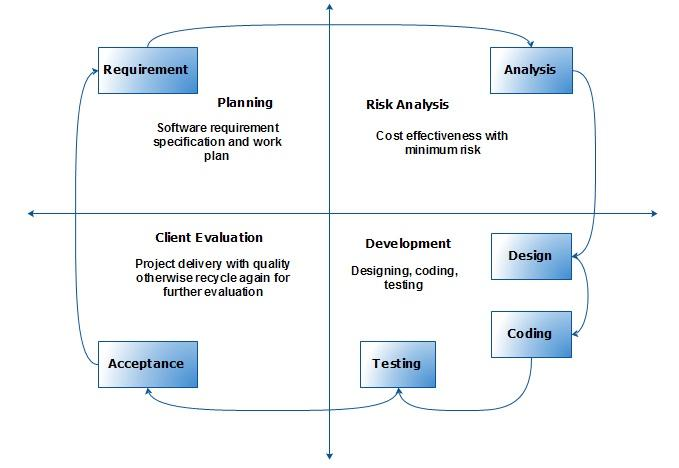
**[1.](#UCS)** [To eliminate the risk that could be faced in development of software the use of spiral methodology is adopted. For example, the risk might be a resignation from the key person](#UCS)

**[2.](#UCS)** [Spiral model has two dimensions. One is radial dimension which represents the cumulative cost to date, and the other is angular dimension which represents the progress through the spiral.](#UCS)

**[3.](#UCS)** [The spiral model is very sensitive to risks. In this model development and maintenance run in parallel. This method is used for development of large-scale and in-house software.](#UCS)

[](#UCS)

## [Vu process model:](#UCS)

[](#UCS)

[This process model is combination of water fall and spiral model. This process model maximizes the quality of system and reduces the risk and disadvantages](#UCS)

### ****[Why VU Process Model?](#UCS)****

[The VU Process Model is ideal for this project because it:](#UCS)

1. **[Ensures Clear Phases:](#UCS)** [The structured approach of the Waterfall model helps in defining well-documented requirements, design, and implementation phases, critical for maintaining project clarity.](#UCS)
2. **[Supports Iterative Refinement:](#UCS)** [The iterative nature of the Spiral model allows us to revisit phases (like design and testing) based on feedback or emerging requirements.](#UCS)
3. **[Manages Risks Effectively:](#UCS)** [By iteratively addressing risks and incorporating feedback, the methodology reduces the chances of failure or unexpected issues.](#UCS)
4. **[Balances Predictability and Flexibility:](#UCS)** [It provides a predictable timeline and milestones, while also allowing flexibility for changes during iterations.](#UCS)

### [Rather](#UCS)

[Our task is isolated as of now into various stages for example assembling and examining prerequisites stage, arranging stage, examination and configuration stage, improvement and last undertaking report stage, and last report/viva stage.](#UCS)

[We will finish each stage in succession and will submit it to our supervisor. He will recommend us about the improvement in each stage before beginning the following stage. We will make improvement in that stage. This procedure will be embraced because of the winding idea of vu procedure model. At the point when the stage is well-improved and very much worked-out, and furthermore acknowledged by our supervisor then we will continue to next stage. This will be done because of the cascade idea of the vu procedure model. Consequently, the two models are smothered in single structure and that will be our vu procedure model. It will give mistake free outcome as each progression is done in a succession.](#UCS)

### ****[Phases of the VU Process Model](#UCS)****

#### ****[1. Requirements Gathering and Analysis](#UCS)****

* [We start by gathering all functional and non-functional requirements of the restaurant chatbot.](#UCS)
* [Stakeholders (e.g., restaurant owners, managers) are involved to ensure all business needs are captured.](#UCS)
* [Deliverable:](#UCS) **[SRS Document](#UCS)**[.](#UCS)

#### ****[2. System Design](#UCS)****

* [Based on the requirements, we design the system architecture and use case diagrams.](#UCS)
* [The design will cover chatbot interaction flows, backend integrations, and data flow.](#UCS)
* [Deliverable:](#UCS) **[System Design Document (UML diagrams, architecture).](#UCS)**

#### ****[3. Prototype Development (Spiral Model Aspect)](#UCS)****

* [A prototype of the chatbot is developed to showcase basic functionality like menu navigation or FAQs.](#UCS)
* [Stakeholder feedback is incorporated to refine the system iteratively.](#UCS)
* [Deliverable:](#UCS) **[Prototype Demonstration.](#UCS)**

#### ****[4. Implementation](#UCS)****

* [After the design and prototyping are finalized, the complete chatbot is implemented using](#UCS) **[Google Dialogflow](#UCS)**[.](#UCS)
* [Core functionalities like table reservations, order placement, menu browsing, and customer support are developed.](#UCS)
* [Deliverable:](#UCS) **[Fully Functional Chatbot.](#UCS)**

#### ****[5. Testing and Validation](#UCS)****

* [The system is tested thoroughly to ensure functionality, usability, and performance.](#UCS)
* [This includes:](#UCS)
  + **[Unit Testing](#UCS)** [for individual components.](#UCS)
  + **[Integration Testing](#UCS)** [for interactions between the chatbot and backend systems.](#UCS)
  + **[User Acceptance Testing (UAT)](#UCS)** [to validate the system with real users.](#UCS)
* [Deliverable:](#UCS) **[Test Report.](#UCS)**

#### ****[6. Deployment and Maintenance](#UCS)****

* [The chatbot is deployed on the restaurant's web platform.](#UCS)
* [Post-deployment, regular updates and maintenance are carried out to enhance features or fix issues.](#UCS)
* [Deliverable:](#UCS) **[Live Chatbot System.](#UCS)**

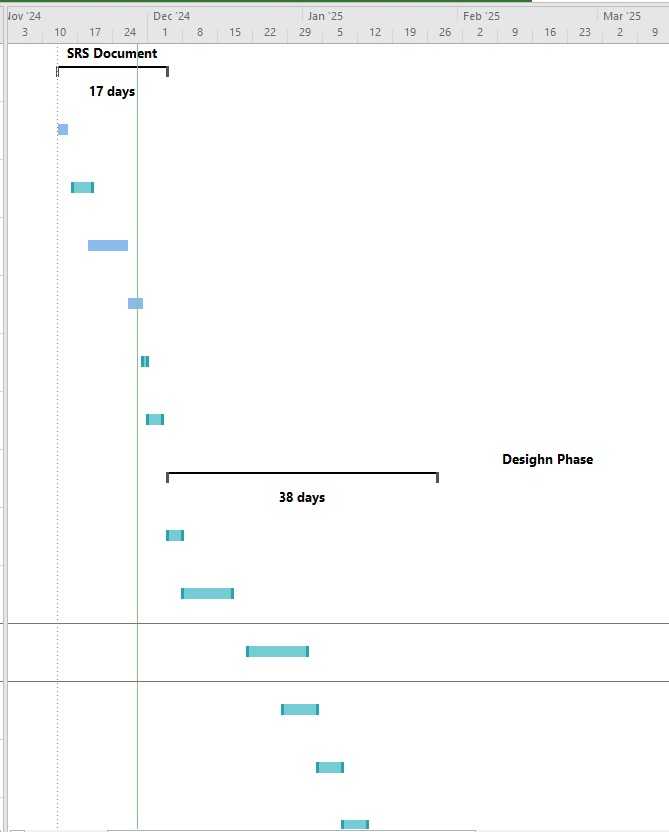
### ****[Key Features of the VU Model in This Project](#UCS)****

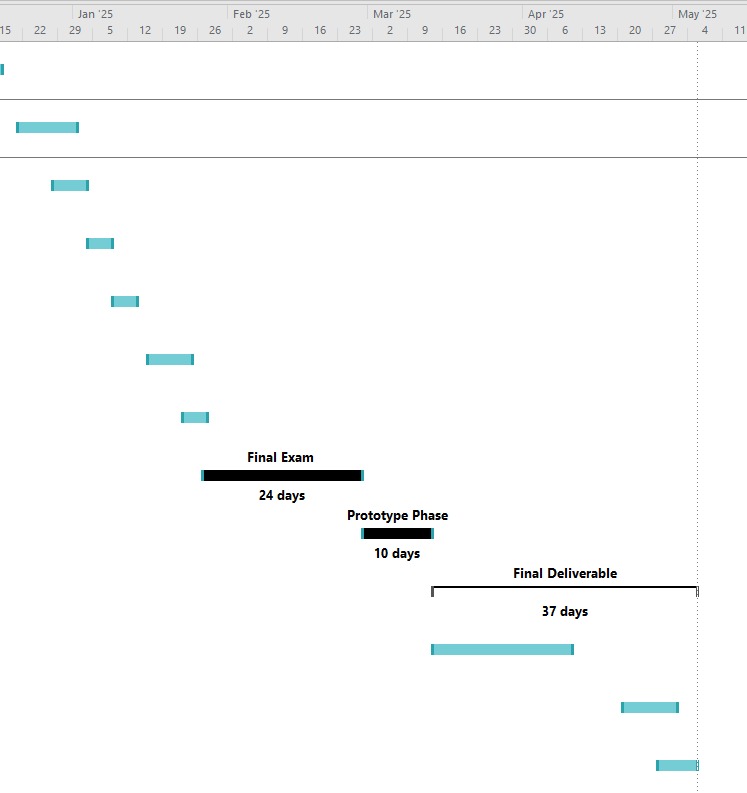
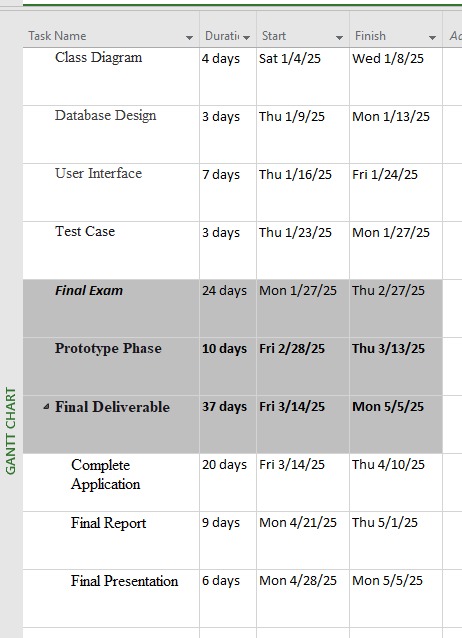
* **[Structured Phases:](#UCS)** [Each phase is clearly defined and documented, ensuring smooth transitions and organized work.](#UCS)
* **[Iterative Prototyping:](#UCS)** [Early prototypes allow stakeholder involvement and reduce the risk of building a system that does not meet user needs.](#UCS)
* **[Risk Analysis:](#UCS)** [During each iteration, potential risks (e.g., failure of chatbot responses, system integration issues) are identified and mitigated.](#UCS)

[窗体顶端](#UCS)

[窗体底端](#UCS)

### ****[6. Work Plan](#UCS)****

[](#UCS)

[](#UCS)