NLP Chatbot Development using Dialogflow

**Software Requirements Specification**

Version 1.0



**Group Id: F24PROJECT9A4A9**

**Supervisor Name: Abdullah Qamar**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date (dd/mm/yyyy)** | **Version** | **Description** | **Author** |
| 30/11/2024 | 1.0 | Initial SRS Document of NLP Chatbot Development using DialogFlow | BC200401263 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

1. [Scope (of the project) Page: 04](#scope)
2. [Functional and Non-Functional Requirements Page: 06](#FRNFR)
3. [Use Case Diagram Page: 08](#UCD)
4. [Usage Scenarios Page: 09](#UCS)
5. [Adopted Methodology](#Adopted) Page: 13
6. [Work Plan (Use MS Project to create Schedule/Work Plan)](#Gantt) Page: 17

**SRS Document**

**Scope of Project:**

The primary purpose of the project is to develop an AI-powered chatbot specifically for a Training Company. This chatbot will streamline and automate various administrative tasks and student interactions, providing an efficient and user-friendly platform for handling inquiries. By leveraging Google DialogFlow’s advanced NLP capabilities, the chatbot aims to improve operational efficiency and enhance the user experience for students, staff, and prospective enrollees.

**Intended Functionalities:**

1. **Student Inquiries:**

* Provide instant responses to frequently asked questions about courses, fees, and schedules.
* Assist with details about training modules and certification requirements.

1. **Course Registration:**

* Enable students to browse available courses and register seamlessly.

1. **Schedule Management:**

* Provide details about upcoming classes, batch schedules, and training timelines.

1. **Personalized Interaction:**

* Offer tailored suggestions based on student interests and previous interactions.

1. **Resource Management:**

* Share downloadable resources such as brochures, course outlines, and class materials.

1. **Management Tasks:**

* Admin shall be able to manage Course Catalogue, FAQs and Resources Distribution.

**Specific Tasks the System Will Accomplish:**

* Automate the process of responding to routine inquiries, reducing staff workload.
* Simplify the registration and scheduling process for students.
* Facilitate course data distribution and management for better decision-making.
* Enhance accessibility by being available 24/7 to address user needs.

**What the System Will Not Address:**

* Advanced analytics or predictive analysis for training outcomes.
* In-depth financial transactions or payment handling.
* High-level decision-making tasks that require human judgment, such as trainer selection.
* Advanced Machine Learning functionalities beyond the basic DialogFlow chatbot.

**Project Overview:**

The chatbot will serve as a digital assistant for the Training Company, ensuring efficient and accurate communication. It will act as a first point of contact for students and staff, making the processes faster and more reliable while reducing the burden on administrative staff. The solution will utilize DialogFlow for its conversational interface and be supported by a robust backend and database for handling data effectively.

**Scope Boundaries:**

* **In-Scope:**
* Addressing queries about courses, registration, schedules, and administrative concerns.
* Managing the distribution of course resources.
* Provides personalized course recommendations based on user interaction history.
* **Out-of-Scope:**
* Financial transactions or e-commerce features.
* Advanced Machine Learning functionalities beyond basic chatbot interaction.
* Integration with external platforms like Learning Management Systems (LMS).

The scope of this project emphasizes the development of an intelligent and interactive chatbot tailored for a Training Company. By automating repetitive tasks and addressing user inquiries efficiently, the chatbot will enhance the operational workflow and improve the overall user experience. Its focused functionality ensures streamlined processes without overstepping into complex areas, making it a practical and impactful solution for the targeted industry.

**Functional and Non-Functional Requirements:**

**Functional Requirements:**

1. **Student Inquiry Handling:**

* The chatbot shall provide responses to frequently asked questions (FAQs) about courses, fees, schedules, and trainers.
* The system shall offer details on course content and certification requirements.

1. **Course Registration:**

* The chatbot shall allow students to view available courses and register for them.
* It shall send confirmation messages upon successful registration.

1. **Schedule Management:**

* The system shall provide information about class schedules and batch timings.
* Users shall be able to check the status of upcoming and ongoing classes.

1. **Resource Distribution:**

* The chatbot shall enable students to download course brochures and training materials.

1. **Personalized Interaction:**

* The system shall provide course recommendations based on user inquiries and interaction history.

1. **Management Tasks:**

* The admin shall be able to Manage Course Catalogue (add, update, or remove course details), and Update FAQs.
* The admin shall upload and manage course brochures and training materials.

**Non-Functional Requirements:**

1. **Performance:**

* **Requirement:** The chatbot shall respond to user queries within 2 seconds.
* **Action:** Optimize DialogFlow’s intent handling and backend queries to ensure minimal latency.

1. **Availability:**

* **Requirement:** The system shall be available 24/7 with minimal downtime.
* **Action:** Deploy on a reliable cloud platform with high availability and redundancy.

1. **Scalability:**

* **Requirement:** The system shall handle up to 500 concurrent users without performance degradation.
* **Action:** Use efficient database queries and asynchronous processing where applicable.

1. **Security:**

* **Requirement:** Ensure data privacy and secure communication with SSL encryption.
* **Action:** Implement encryption protocols and secure coding practices in the backend.

1. **Usability:**

* **Requirement:** The chatbot interface shall be intuitive and easy to navigate for all users.
* **Action:** Conduct usability testing and ensure adherence to UI/UX design principles.

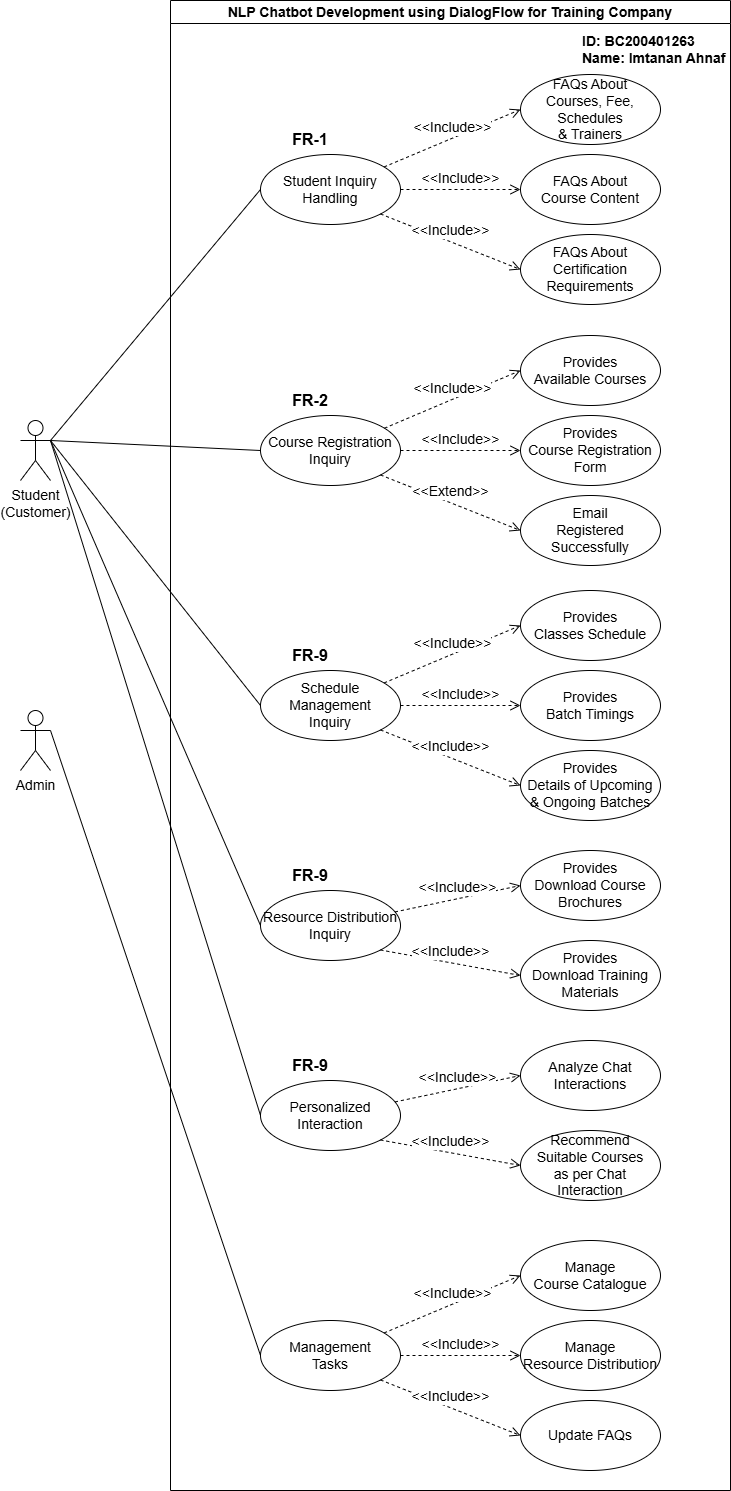
1. **Maintainability:**

* **Requirement:** The system shall be easy to update and maintain.
* **Action:** Use modular code architecture to simplify updates and bug fixes.

1. **Reliability:**

* **Requirement:** The chatbot shall maintain an uptime of 99.9%.
* **Action:** Monitor system performance and implement automatic error recovery mechanisms.

**Use Case Diagram(s):**



**Usage Scenarios:**

|  |  |
| --- | --- |
| **FR ID:** | **FR-01** |
| **Use Case Title:** | Student Inquiry Handling |
| **Use Case Id:** | UC-01 |
| **Actor:** | Student (Customer) |
| **Actions:** | 1. User asks an inquiry. 2. Chatbot retrieves FAQs. 3. Chatbot provides a relevant response. |
| **Description:** | The chatbot answers FAQs related to courses, fees, schedules, trainers, content, and certification requirements. |
| **Alternative Paths:** | User may refine or rephrase their query if the chatbot does not initially find a match. |
| **Pre-Conditions:** | User has access to the chatbot interface. |
| **Post Conditions:** | User receives an accurate response. |
| **Exception:** | Chatbot fails to find an answer due to incomplete FAQs or incorrect query interpretation. |
| **Author:** | BC200401263 |

|  |  |
| --- | --- |
| **FR ID:** | **FR-02** |
| **Use Case Title:** | Course Registration Inquiry |
| **Use Case Id:** | UC-02 |
| **Actor:** | Student (Customer) |
| **Actions:** | 1. User views course catalog. 2. User selects a course. 3. Chatbot provides a registration form. |
| **Description:** | Allows students to register for available courses, completing the process with email confirmation. |
| **Alternative Paths:** | User may choose another course after viewing the catalog. |
| **Pre-Conditions:** | Courses are available and updated in the catalog. |
| **Post Conditions:** | User successfully registers for a course and receives a confirmation. |
| **Exception:** | Registration fails if required fields are incomplete or if there is a technical issue. |
| **Author:** | BC200401263 |

|  |  |
| --- | --- |
| **FR ID:** | **FR-03** |
| **Use Case Title:** | Schedule Management Inquiry |
| **Use Case Id:** | UC-03 |
| **Actor:** | Student (Customer) |
| **Actions:** | 1. User asks about schedule. 2. Chatbot provides class details, timings, or batch statuses. |
| **Description:** | Students can inquire about ongoing/upcoming class schedules, batch timings, or statuses. |
| **Alternative Paths:** | User can request details of a different class or batch if needed. |
| **Pre-Conditions:** | Schedule information is available in the system. |
| **Post Conditions:** | User gets accurate schedule details. |
| **Exception:** | Details cannot be retrieved due to missing or outdated schedule data. |
| **Author:** | BC200401263 |

|  |  |
| --- | --- |
| **FR ID:** | **FR-04** |
| **Use Case Title:** | Resource Distribution Inquiry |
| **Use Case Id:** | UC-04 |
| **Actor:** | Student (Customer) |
| **Actions:** | 1. User requests materials. 2. Chatbot provides links for download. |
| **Description:** | The chatbot enables students to download course brochures or training materials. |
| **Alternative Paths:** | User can try downloading alternate resources if initial selection fails. |
| **Pre-Conditions:** | Resources are uploaded and available in the system. |
| **Post Conditions:** | User successfully downloads requested resources. |
| **Exception:** | Resources unavailable or downloads fail due to server or file issues. |
| **Author:** | BC200401263 |

|  |  |
| --- | --- |
| **FR ID:** | **FR-05** |
| **Use Case Title:** | Personalized Interaction |
| **Use Case Id:** | UC-05 |
| **Actor:** | Student (Customer) |
| **Actions:** | 1. User interacts with chatbot. 2. Chatbot analyses interaction. 3. Chatbot suggests courses. |
| **Description:** | The chatbot recommends suitable courses based on interaction history or specific inquiries. |
| **Alternative Paths:** | User may ignore the chatbot's recommendation and explore the course catalog manually. |
| **Pre-Conditions:** | User provides enough interaction data for analysis. |
| **Post Conditions:** | User receives personalized course suggestions. |
| **Exception:** | Recommendation fails if user data is insufficient or interaction history is incomplete. |
| **Author:** | BC200401263 |

|  |  |
| --- | --- |
| **FR ID:** | **FR-06** |
| **Use Case Title:** | Management Tasks |
| **Use Case Id:** | UC-06 |
| **Actor:** | Admin |
| **Actions:** | 1. Admin logs in. 2. Admin performs tasks like managing course catalog, FAQs, or resources. |
| **Description:** | The chatbot answers FAQs related to courses, fees, schedules, trainers, content, and certification requirements. |
| **Alternative Paths:** | Admin can manage the course catalog, upload/download training materials, and update FAQs for improved system responses. |
| **Pre-Conditions:** | Admin may choose to delegate specific tasks (e.g., uploading brochures) to a subordinate. |
| **Post Conditions:** | System is updated with new courses, FAQs, and resources. |
| **Exception:** | Changes fail to save due to incorrect inputs or system errors (e.g., server downtime). |
| **Author:** | BC200401263 |

**Adopted Methodology:**

Methodology is a framework that is used to structure control and process of developing information system. VU process model is a combination of Water Fall model and Spiral model.

**Water Fall Model:**

This model is also known as linear sequential model because of starting with one stage then onto the next. This model is known as Water Fall model.

Water fall model is depicted in following diagram.

Requirement Definition

System and software design

Implementation and unit testing

Integration and system testing

Operation and maintenance

It recommends a systematic, sequential approach to software development that begins at system level and progresses through the analysis, design, coding, testing and maintenance. No phase is complete until the documentation for that phase has been complete.

**Spiral Model:**

Spiral model is Water Fall model with risk analysis. In that case each stage is preceded by identification of alternatives and risk analysis, and is then followed by evaluation and planning for next phase. Spiral model is depicted in following diagram.

Rapid Prototyp

Specification

Design

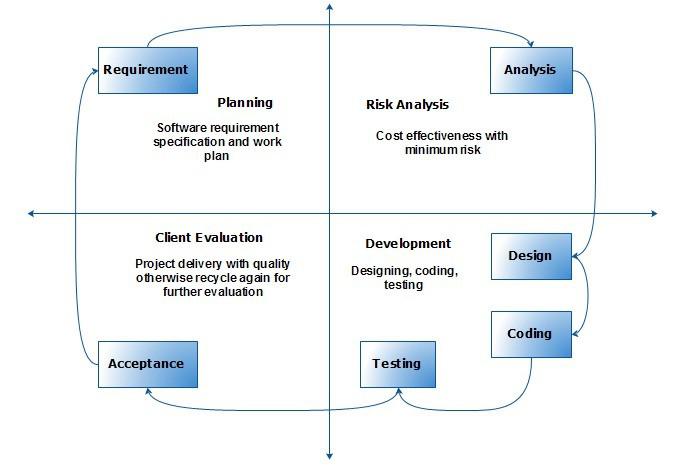
Implementation

Verify

Risk Analysis

**VU Process Model:**

Having detailed knowledge of Water Fall model and Spiral model, and keeping in view nature of my project, I choose VU process model that is the combination of Water Fall model and Spiral model. It explains the simplified description of software processes in iteration to avoid maximum risk, Both spiral and water fall methods combine to develop a hybrid approach of system development that maximize the quality of the system and minimizes the risk and disadvantages to enhance the results of the system.

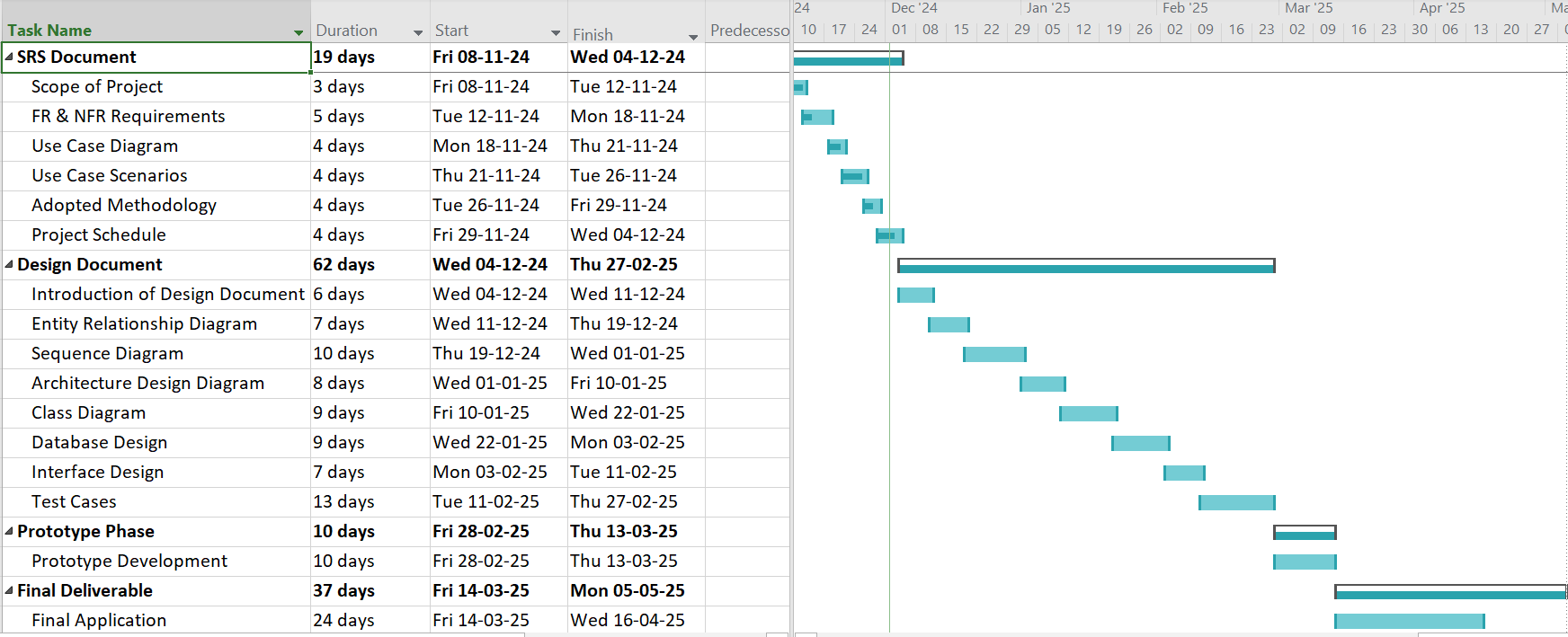


**Reasons for choosing VU Process Model:**

VU Process model has benefits of probability from the signs-based planning of Water Fall model as well as the benefit of feedbacks and creativity from Spiral model, so there are some following reasons.

* The VU process model is documentation- driven model. It therefore generates complete and comprehensive documentation and hence makes the maintenance much easier.
* VU process model works on phases so it is better to complete our project. It is sequential Model with backward repetition
* All activities are performed in sequence in vu process model.
* Another reason for choosing vu process model is we can do correction or maintenance at any stage whenever necessary.

**Work Plan (Use MS Project to create Schedule/Work Plan):**



*The End!*