**Final Project Report**

**NLP Chatbot Development using Dialogflow**



**Project Supervisor**

**Abdullah Qamar**

**Submitted By**

**F24PROJECT8DE28**

Samer Yousaf(BC210415622)

**Software Projects & Research Section,**

**Department of Computer Sciences,**

**Virtual University of Pakistan**

|  |
| --- |
|  |



**CERTIFICATE**

This is to certify that **F24PROJECT8DE28 (bc210415622)** has worked on and completed their Software Project at Software & Research Projects Section, Department of Computer Sciences, Virtual University of Pakistan in partial fulfillment of the requirement for the degree of BS in Computer Sciences under my guidance and supervision.

In our opinion, it is satisfactory and up to the mark and therefore fulfills the requirements of BS in Computer Sciences.

**Supervisor / Internal Examiner**

**Abdullah Qamar**

Supervisor,

Software Projects & Research Section,

Department of Computer Sciences

Virtual University of Pakistan

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Signature)

**External Examiner/Subject Specialist**

<<External Supervisor Name>>

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Signature)

**Accepted By:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_**

(For office use)

**EXORDIUM**

**In the name of Allah, the Compassionate, the Merciful.**

**Praise be to Allah, Lord of Creation,**

**The Compassionate, the Merciful,**

**King of Judgment-day!**

**You alone we worship, and to You alone we pray for help,**

**Guide us to the straight path**

**The path of those who You have favored,**

**Not of those who have incurred Your wrath,**

**Nor of those who have gone astray.**

**DEDICATION**

I dedicate this achievement first and foremost to God, my Creator. I truly believe He blessed me with the inspiration, wisdom, knowledge, and understanding needed for this journey. His strength carried me through challenges, and His guidance led me to this successful outcome. My heartfelt thanks and dedication also go to my dear friends. His unwavering support and encouragement meant the world to me and were vital in finishing this work. Finally, I dedicate this to my valued teacher Sir Abdullah Qamar, who instilled in me a deep appreciation for the importance of seeking knowledge purely for the sake of learning.

**ACKNOWLEDGEMENT**

I am profoundly grateful to Allah for providing the resources and strength to bring this project to fruition. His constant guidance and protection were invaluable, especially during difficult phases. My sincere thanks also extend to our project manager Sir Abdullah Qamar, whose insightful suggestions and unwavering support were instrumental to our success. I also deeply appreciate the contributions of my parents and friends, whose generous time and financial assistance were crucial throughout this journey.**PREFACE**

This NLP Chatbot Development using Dialogflow Web project represents my own effort and the skills I've developed during this learning process. As a newcomer to web development, I acknowledge that this work may contain imperfections. Throughout this project, I focused on applying fundamental concepts to create a functional demonstration. I'm grateful for the support and guidance provided by my friends and teachers. While this project might not meet the demands of a large-scale organization, I hope it serves as a helpful learning resource for other aspiring web developers.

**TABLE OF CONTENT**

[1.1 Introduction 10](#_Toc197302368)

[1.2 Purpose 10](#_Toc197302369)

[1.3 Scope 10](#_Toc197302370)

[1.5 Project requirements 12](#_Toc197302371)

[1.5.1 Functional Requirements 12](#_Toc197302372)

[1.5.2 Non-Functional Requirements 13](#_Toc197302373)

[1.6 use cases and usage scenarios 14](#_Toc197302374)

[1.6.1 Use Case Diagrams 14](#_Toc197302375)

[1.6.2 Usage Scenarios 15](#_Toc197302376)

[1.7.3 Work Plan (Gantt chart) 21](#_Toc197302377)

[1.7.4 Project Schedule (Submission Calendar) 21](#_Toc197302378)

[2.1 Introduction 23](#_Toc197302379)

[2.5 Architectural Representation (Architecture Diagram) 24](#_Toc197302380)

[2.6 Dynamic Model: Sequence Diagrams 25](#_Toc197302381)

[2.7 Object Model/Logical Model: Class Diagram 26](#_Toc197302382)

[2.8 Database Model (Database Diagram) 27](#_Toc197302383)

[2.9 Graphical User Interfaces 28](#_Toc197302384)

[3.1 Development plan (Architecture Diagram) 32](#_Toc197302385)

<<Dear Students, you have to take special care of page numbers, which are written against each Chapter and sub headings. You have to mention your own page numbers and correct one against each document heading. Make proper table of contents. Remember that if there are any new headings or you have included more headings in your documents then you also have to write these in table of contents and modify table of contents according to your documents material>>

**<<Dear Students, before starting each chapter the following would be the title page for each chapter on a separate page>>**

**CHAPTER 1**

Gathering & Analyzing Info

# Introduction

Chatbots are increasingly becoming integral to industries such as customer service, e-commerce, healthcare, and education. In this project, students will develop a chatbot using Google Dialogflow, an NLP-powered conversational interface. The chatbot should address a specific business need by automating customer interaction, providing relevant responses, and enhancing user experience. Students can select a case study from one of three industries: **Training Company**, **Pharmacy Store**, or **Restaurant**

# 1.2 Purpose

The scope of the project includes the development of intelligent NLP-based chatbots using Dialogflow to meet specific business needs in industries such as customer service, e-commerce, healthcare, and education.

The chatbots will automate important tasks such as answering frequently asked questions, providing product recommendations, scheduling appointments, and providing educational guidance.

It will use the dialogflow's natural language understanding function to accurately process the user's input, identify intention, pull out entity and generate related answers.

Chat robots will support the placement of platforms such as sites, mobile applications and social media many channels to ensure smooth integration with external systems, such as API or databases for dynamic and personalized interactions.

# 1.3 Scope

The scope of the project includes the development of intelligent NLP-based chatbots using Dialogflow to meet specific business needs in industries such as customer service, e-commerce, healthcare, and education.

The chatbots will automate important tasks such as answering frequently asked questions, providing product recommendations, scheduling appointments, and providing educational guidance.

It will use the dialogflow's natural language understanding function to accurately process the user's input, identify intention, pull out entity and generate related answers.

Chat robots will support the placement of platforms such as sites, mobile applications and social media many channels to ensure smooth integration with external systems, such as API or databases for dynamic and personalized interactions.

* **Intended Functionalities:**
* Table Reservation
* Menu Navigation
* FAQS
* Feedback
* **Specific Task the System Will Accomplish:**
* Easy Customer Interaction
* Better Operations
* Personalized Experience
* **What the System Will Not Address:**
* Food Preparation
* Delivery
* Payment Handling
* Complex Quires
* **Project Overview:**

The project developed a Google Dialogflow chatbot that can automate tasks such as FAQs, recommendations, and scheduling in industries such as customer service and education.

Using NLP for accurate, context-aware responses provides user-friendly, scalable, and effective solutions across platforms.

* **Scope Boundaries:**

**In scope:** NLP chatbot development.

**Out scope:** complex conversation, advanced AI.

# 1.5 Project requirements

## 1.5.1 Functional Requirements

* **Table Reservations:**
* **Activity:** Enable the customer to check table available or not for desired date, time and party size.
* **Action:** The user provides detail about date, time and no. of guests. The chatbot will check availability and then process confirmation.
* **Order Placements:**
* **Activity:** Customers should be able to place order from the chatbot for takeaway or dining in.
* **Action:** After reviewing the menu, users can select items to order and the chatbot should collect their selections, confirm the order.
* **Menu Navigation:**
* **Activity:** A complete menu with item descriptions, prices, and exclusive deals should be presented to users by the chatbot.
* **Action:** The user has the option to ask the chatbot questions regarding particular menu items. The requested information, including any current sales or exclusive offers, is provided by the chatbot in response.
* **Customers Support:**
* **Activity:** Inquiries about the restaurant's location, hours of operation, frequently asked questions, and feedback should all be handled by the chatbot.
* **Action:** Users can ask questions about the restaurant's location or operating hours, as well as ask questions and provide feedback. The chatbot will respond with pertinent and factual information.

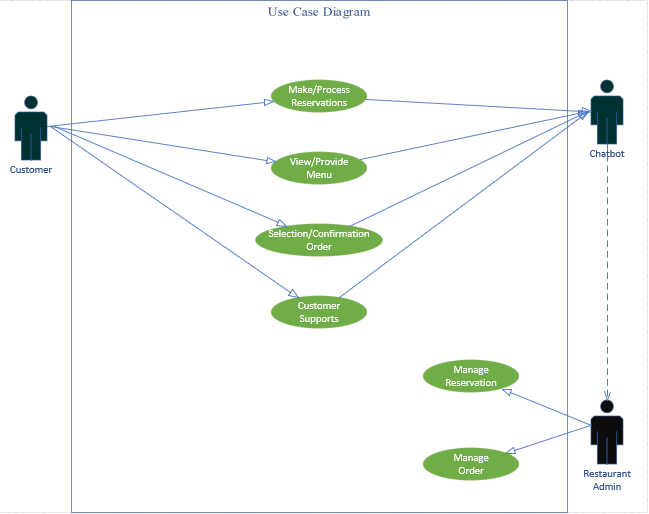
## 1.5.2 Non-Functional Requirements

* **Scalability:**
* **Activity:** The system should be able to handle increased traffic, particularly on weekends or holidays when demand is at its peak.
* **Action:** Make sure the foundational system is scalable to handle unexpected spikes in user activity. Use cloud-based hosting with autoscaling features, for instance.
* **Security:**
* **Activity:** The chatbot should protect user data, particularly private data like payment preferences and contact information
* **Action:** Encrypt communication channels, abide by data protection regulations and make sure user data is handled and stored securely.
* **Usability:**
* **Activity:** The chatbot interface should be user-friendly and easy to navigate.
* **Action:** Create a conversational flow that is easy to follow so that the chatbot can comprehend user inquiries and reply in a way that is engaging and natural.
* **Reliability:**
* **Activity:** The chatbot should be available 24/7 with minimal downtime.
* **Action:** Use monitoring tools and a dependable hosting service to identify and address issues quickly. The system's continued operation will be guaranteed by routine backups and recovery procedures.
* **Performance:**
* **Activities:** The chatbot should respond to user queries within 2-3 seconds.
* **Actions:** Optimization of the backend processing**,** response caching where necessary, and making sure the server or cloud architecture can support numerous users at once without experiencing noticeable delay

# 1.6 use cases and usage scenarios

## 1.6.1 Use Case Diagrams

**Use Case Diagram(s):**



## 1.6.2 Usage Scenarios

**Usage Scenarios**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Use Case**  **Title** | **Use case Id** | **Description** | **Alternative**  **Paths** | **Pre-**  **Condition** | **Post-**  **Condition** | **Actors** | **Exception** |
| **Make Reservations** | UC-1 | Customer books a table using the chatbot. | Suggests alternate slots for unavailable times. | Customer provides valid details. | Reservation confirmed, notification sent. | Customer | System fails to confirm availability |
| **View Menu** | UC-2 | Customer browses the menu through the chatbot. | Notify customer if menu is under maintenance. | Menu data is available. | Menu displayed successfully. | Customer | Menu fails to load. |
| **Selection/**  **Confirmation Order** | UC-3 | Customer places an order for selected menu items. | Suggest alternative items if some are unavailable. | Customer is logged in. | Order confirmed, details recorded. | Customer | Invalid payment or network failure. |
| **Customer Support** | UC- 4 | Customer contacts chatbot for support regarding orders/issues. | None. | Chatbot is functional | Issue resolved or escalated to admin. | Customer | Chatbot fails to understand query. |
| **Process Reservations** | UC-5 | Chatbot processes the reservation request made by the customer. | Suggests alternative slots for unavailable reservations | Customer details are valid. | Reservation confirmed. | Chatbot | System fails to check availability |
| **Provide Menu** | UC-6 | Chatbot provides menu categories and items. | Notify the customer if menu data is temporarily unavailable | Menu data exists in the system. | Menu provided successfully. | Chatbot | Database error or missing records. |
| **Confirm Orders** | UC-7 | Chatbot processes the customer's order. | Suggest substitutions for unavailable items. | Items are in stock. | Order processed and saved in system. | Chatbot | Inventory fails to sync. |
| **Provide Customer Support** | UC-8 | Chatbot handles customer support requests. | Forward unresolved queries to the restaurant admin. | Chatbot is online. | Support query resolved or escalated. | Chatbot | Query misinterpretation. |
| **Manage Reservations** | UC-9 | Admin oversees and modifies reservations made by customers. | Notify customer of any updates. | Admin is logged into the system | Reservation details updated. | Restaurant  Admin | Reservation record not found. |
| **Manage Orders** | UC-10 | Admin manages order processing and ensures accurate updates. | Notify customers about order status changes. | Admin is authenticated | Order details updated in the system. | Restaurant  Admin | Order record missing or system crash. |
|  | | | | **Author** | | **F24PROJECT8DE28** | |

1.7 Development Methodology

**Adopted Methodology: -**

Methodology for software development is a set of workflow techniques and methods that are harnessed to design I.T. software solutions. The aim of the various methodologies is to organize teams so that these can develop the program features in the best possible way.

Vu Process Model is combination of the Water Fall and Spiral Model.

* **Waterfall Model:**

The five stages are as follows:

1. **Requirement Analysis and Definition:**

What The systems services, constraints and goals are established by consultation with system users. They are then defined in detail and serve as a system specification.

1. **System and Software Design:**

How the system design process partitions the requirements to either hardware of software systems. It establishes and overall system

Architecture. Software design involves fundamental system abstractions and their relationships.

1. **Implementation and Unit Testing:**

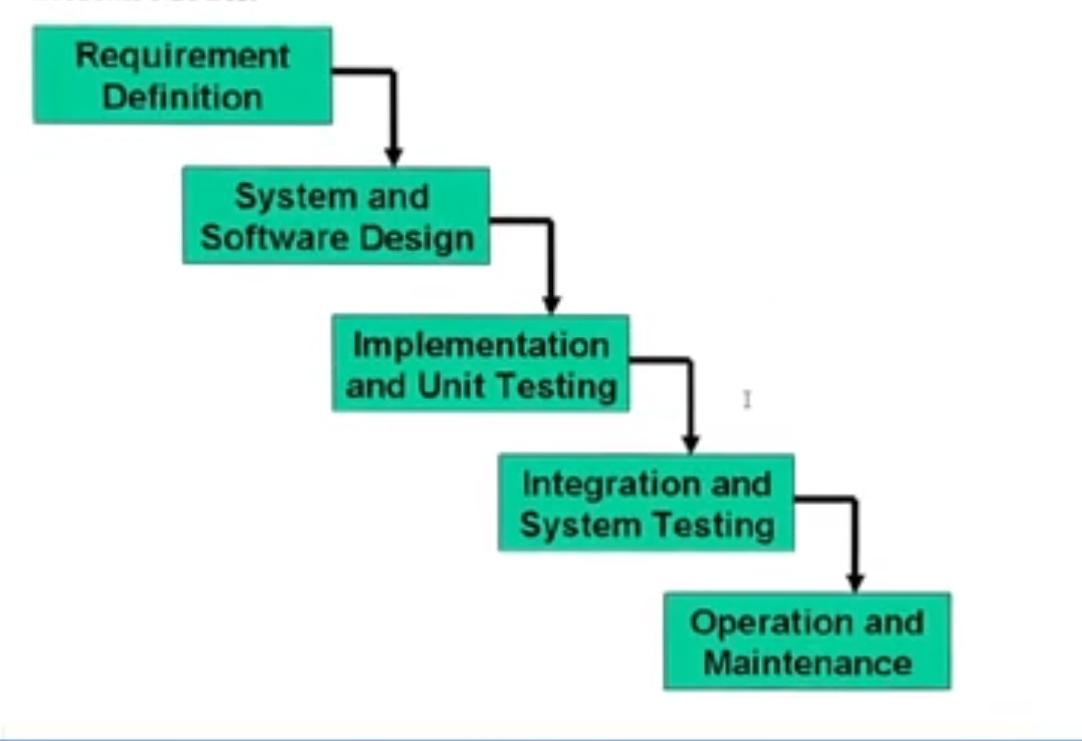
How – During this stage the software design is realized as a set of programs or program units. Unit testing involves verifying that each unit meets its specifications.

1. **Integration and system testing:**

The individual program unit or programs are integrated and tested as a complete system to ensure that the software requirements have been met. After testing, the software system is delivered to the customer.

1. **Operation and Maintenance:**

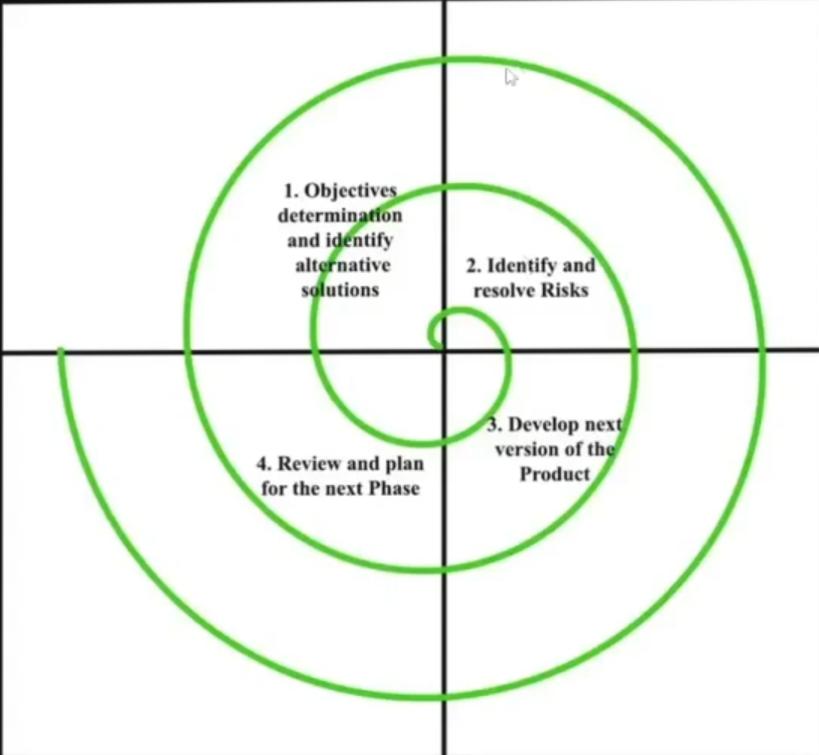
Normally this is the longest phase of the software life cycle. The system is installed and put into practical use. Maintenance involves correcting errors which were not discovered in earlier stages of the life-cycle, Improving the implementation of system units and enhancing the system’s services as new requirements are discovered.

****

* **Spiral Model:**

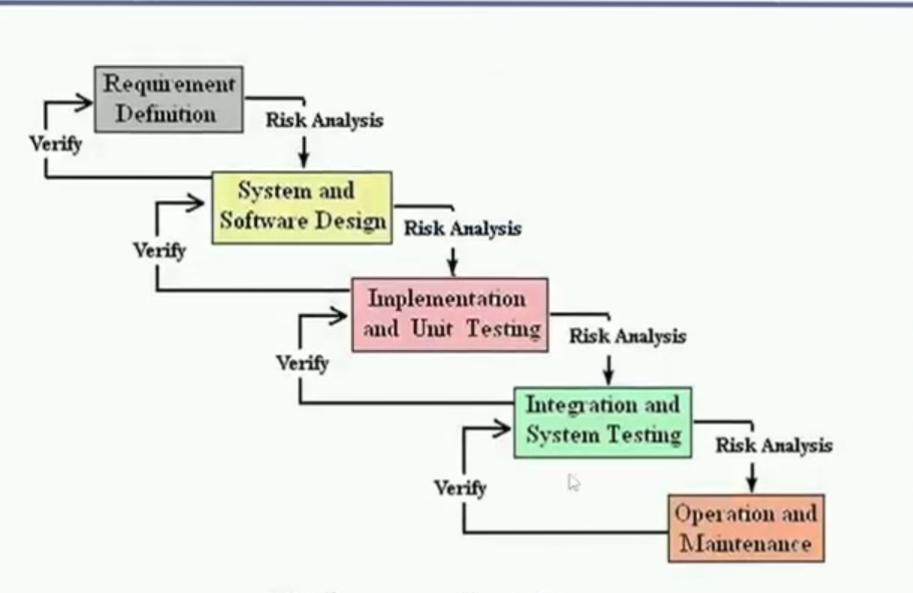
The Spiral Model is a **Software Development Life Cycle (SDLC)** model that provides a systematic and iterative approach to software development. In its diagrammatic representation, looks like a spiral with many loops. The exact number of loops of the spiral is unknown and can vary from project to project. Each loop of the spiral is called a **Phase of the software development process.**

The four stages are as follows:

1. **Planning**: This phase involves establishing objectives, identifying constraints, and defining requirements. During this phase, the project scope, schedule, and cost estimates are determined. Risk analysis is also performed to identify potential risks and devise strategies to mitigate them.
2. **Risk Analysis:** In this phase, potential risks are identified and analyzed. This includes identifying technical risks, resource risks, schedule risks, and other project-specific risks. Risk analysis helps in prioritizing risks based on their likelihood and impact on the project.
3. **Engineering**: This phase involves the development of the product incrementally. Each iteration of the spiral represents a cycle of development. Requirements are gathered, design is created, code is written, and the system is tested. This phase focuses on building and refining the product through iterative development cycles.
4. **Evaluation**: In this phase, the product is evaluated to determine if it meets the specified requirements and quality standards. Feedback from stakeholders and end-users is collected and incorporated into the next iteration of the spiral. This phase also includes reviewing the project status, identifying lessons learned, and updating project plans based on the feedback received.

* **VU Process Model:**

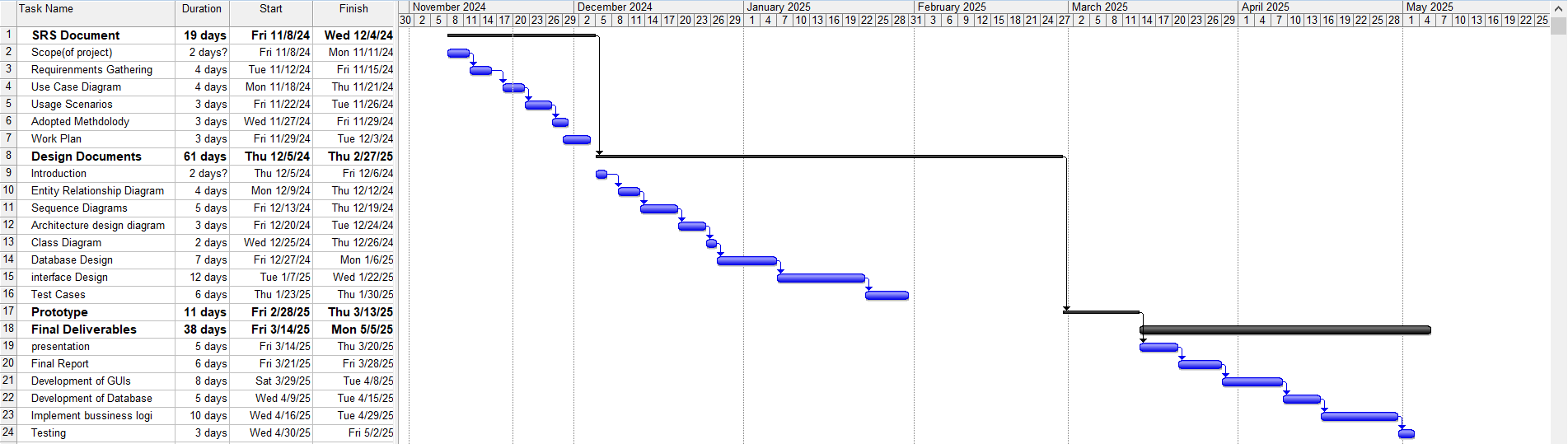
This is the combination of Waterfall and Spiral model. In this model each stage of waterfall is preceded by identification of alternatives and risk analysis and followed by evaluation and planning for next phases



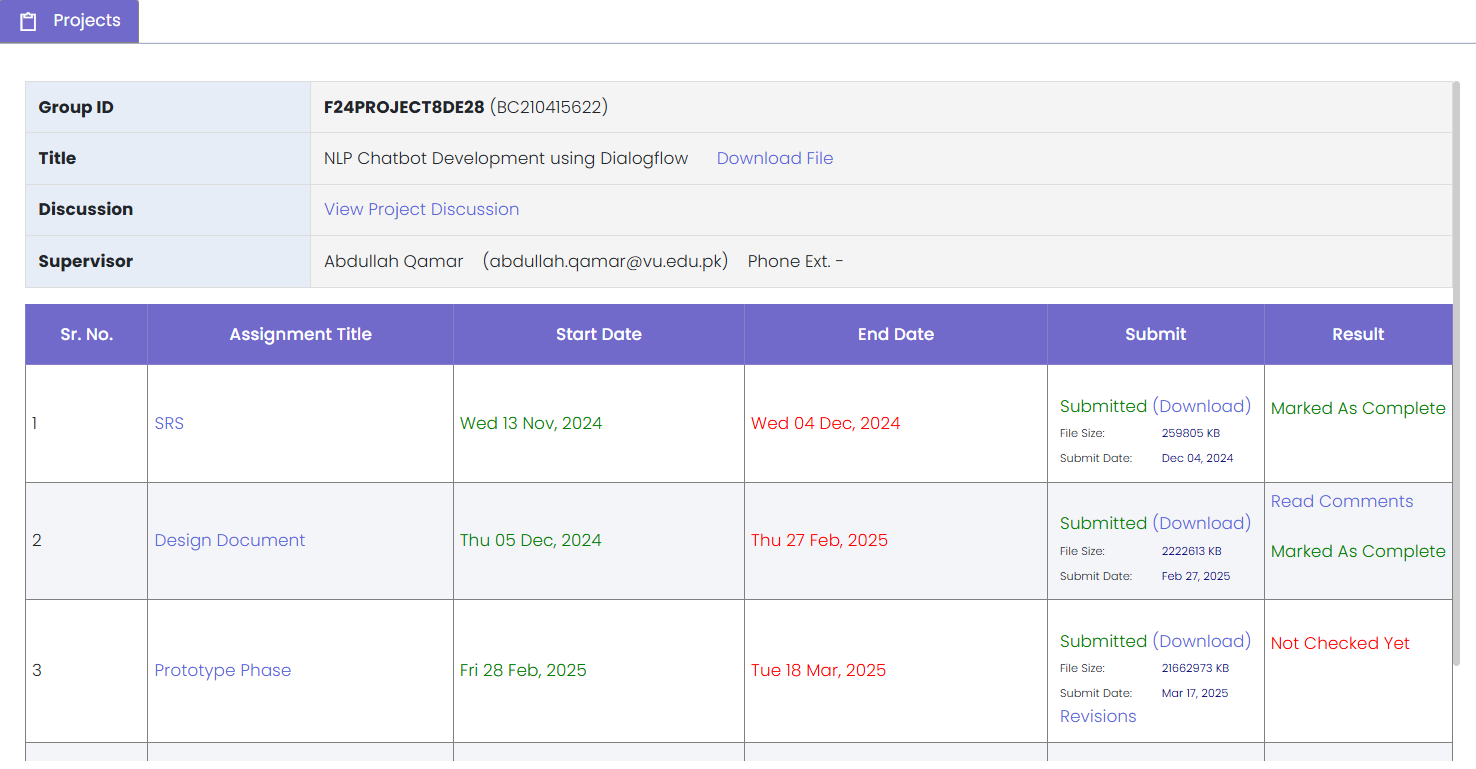
**Reason for choosing VU PROCESS MODEL:**

* Our project is divided into different phases which we have to complete in sequence and submit to our supervisor.
* If some mistakes are found than he suggest us to correct that mistakes and improve our project. This process will be adopted due to its spiral nature.
* When the phase is well worked-out, and also accepted by our Supervisor then we will proceed to next phase. This will be done due to the waterfall nature of the VU Process model.
* Hence both models are suppressed in single form and that will be our VU process model. It will provide error free result as each step is done in a sequence.

## 1.7.3 Work Plan (Gantt chart)



## 1.7.4 Project Schedule (Submission Calendar)



**CHAPTER 2**

Designing the Project

# 2.1 Introduction

The design document plays a crucial role in the software development lifecycle, serving as a blueprint that guides developers in implementing the envisioned system. This document outlines the structure, architecture, and key functionalities of our project. It provides a comprehensive overview of the proposed solution, ensuring that all stakeholders have a clear understanding of the system's design before development begins.

**Purpose of design phase:**

* **Clarity and Consistency:**

The design phase aims to establish a clear and consistent vision for the web application. By detailing the system architecture, data flow, and user interfaces, the design document ensures that developers, designers, and other stakeholders are on the same page regarding the project's objectives.

* **Blueprint for Development:**

The design document serves as a blueprint that developers can follow during the implementation phase. It provides a structured approach to coding, helping to minimize errors and ensuring that the final product aligns with the client's requirements.

* **Risk Mitigation:**

Identifying potential challenges and risks early in the design phase allows for proactive planning and mitigation strategies. This reduces the likelihood of unexpected issues arising during development and helps maintain project timelines.

* **Communication Tool:**

The design document acts as a communication tool between different teams and stakeholders. It facilitates discussions on architecture, functionalities, and user experience, fostering collaboration and ensuring everyone involved is well-informed.

**Benefits of the design phase:**

* **Reduced Development Costs:**

Early identification of design flaws and improvements helps prevent costly rework during the development phase.

* **Improved Collaboration:**

Clear documentation fosters better communication between development teams, designers, and stakeholders, leading to a more cohesive and successful project.

* **Enhanced User Experience:**

A well-thought-out design document ensures that user interfaces are intuitive and aligned with user expectations, resulting in a positive user experience.

* **Risk Management:**

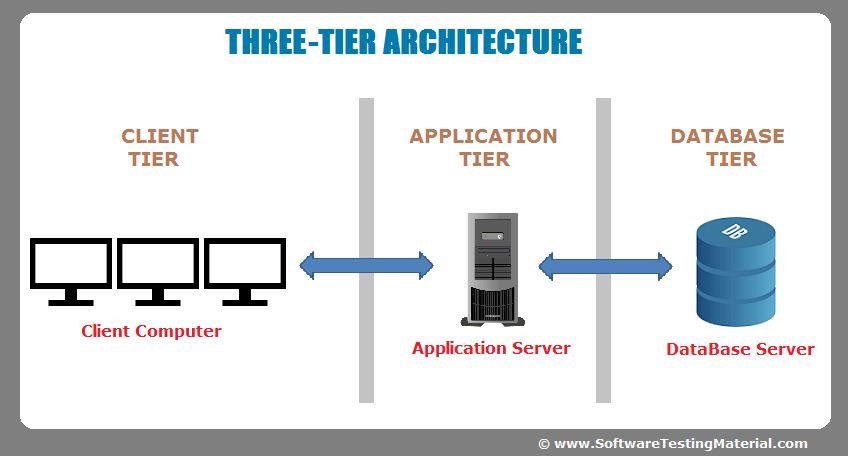
Anticipating and addressing potential risks at the design stage allows for proactive risk mitigation, reducing the likelihood of issues affecting the project timeline and budget.

* **Client Confidence:**

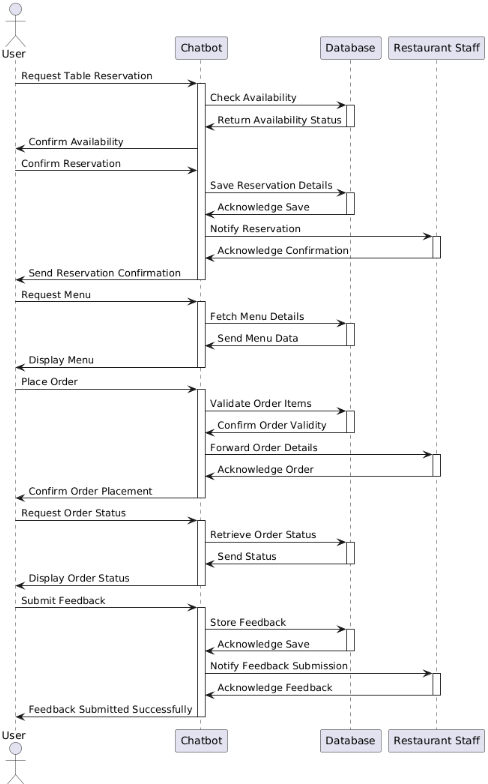
A detailed design document instills confidence in clients by providing a tangible representation of the proposed solution, allowing them to visualize the final product before development begins.

Top of Form

# 2.5 Architectural Representation (Architecture Diagram)



# 2.6 Dynamic Model: Sequence Diagrams



# 2.7 Object Model/Logical Model: Class Diagram

1. **Class Diagram**



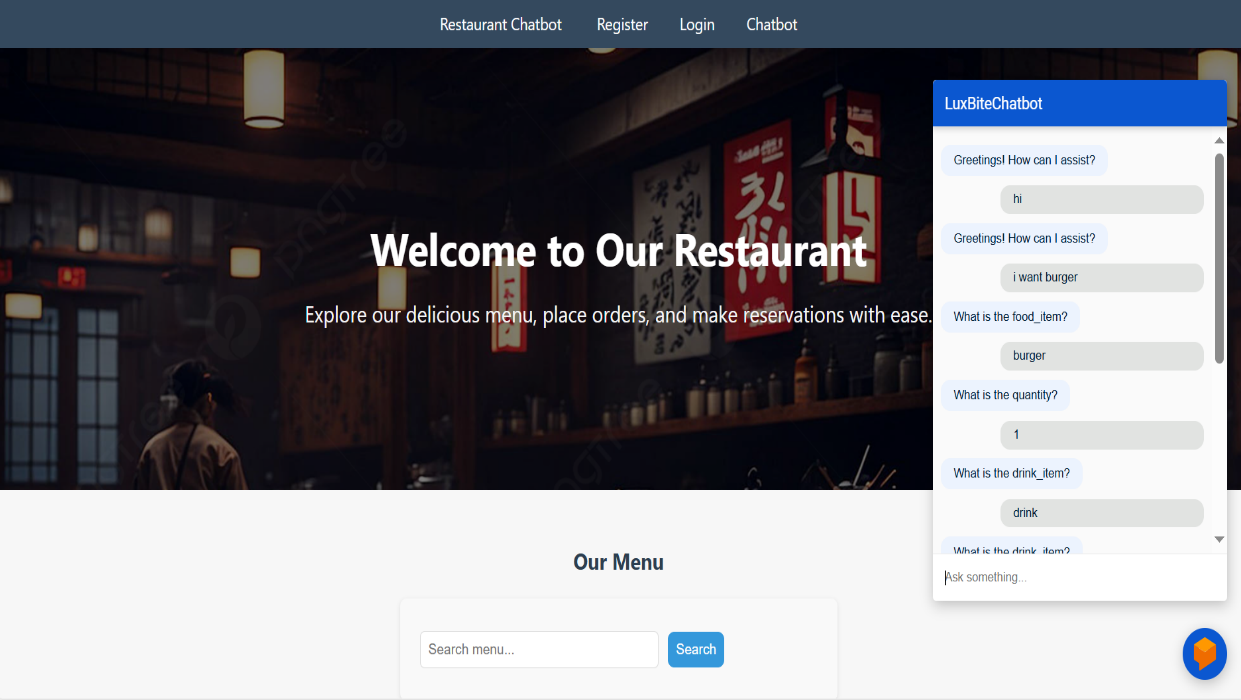
# 

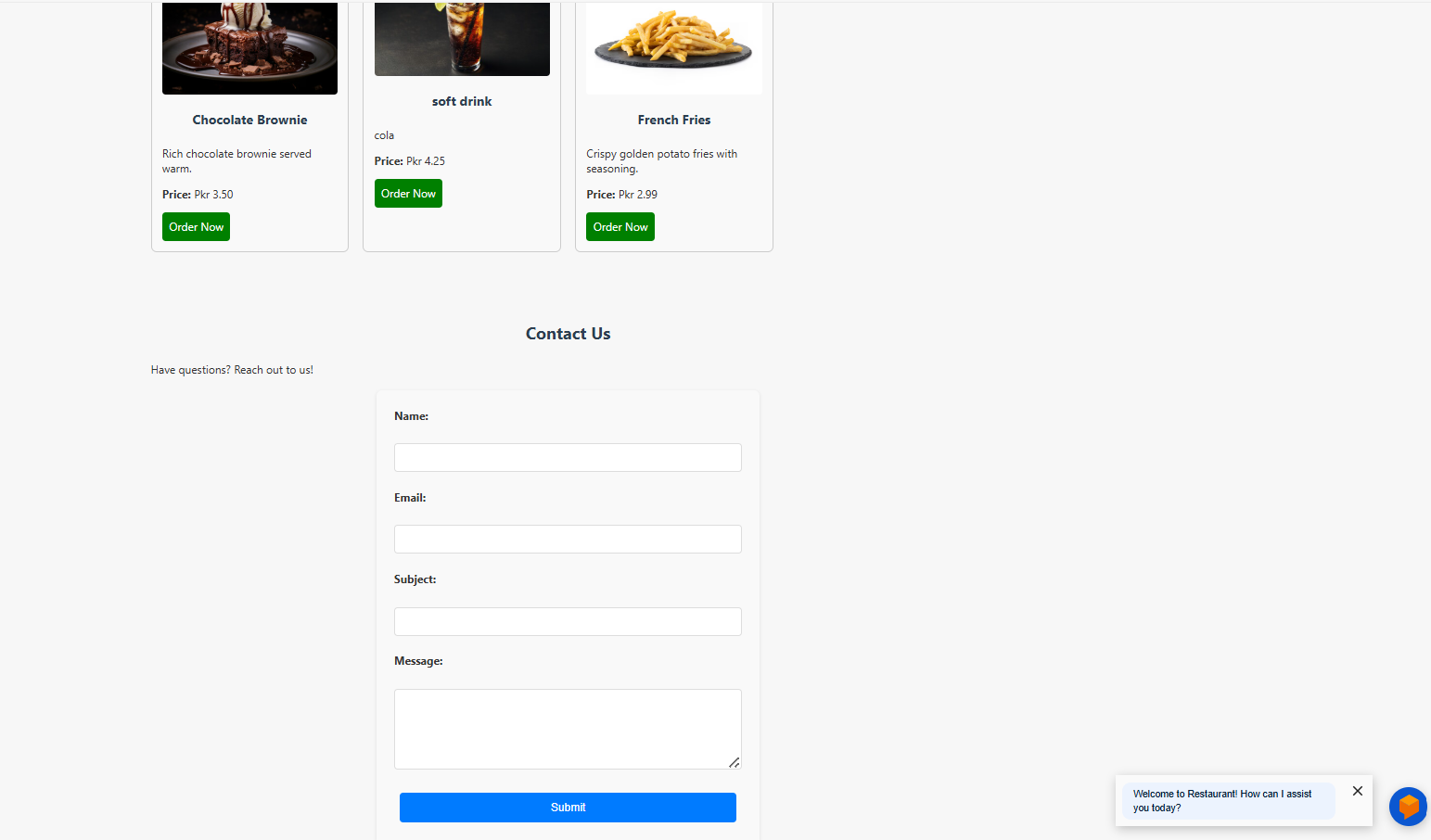
# 2.8 Database Model (Database Diagram)

1. **Database Design**

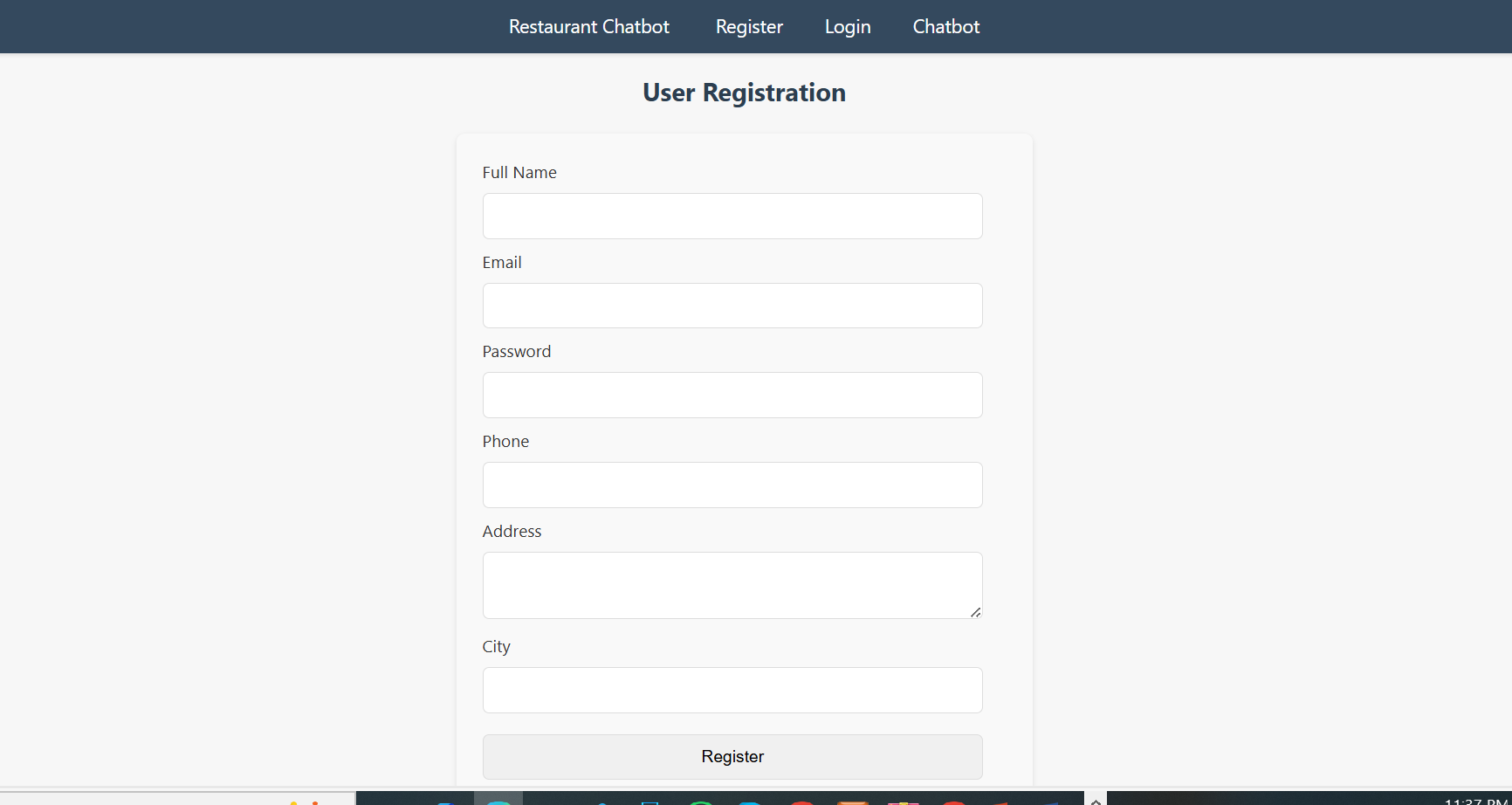


# 2.9 Graphical User Interfaces

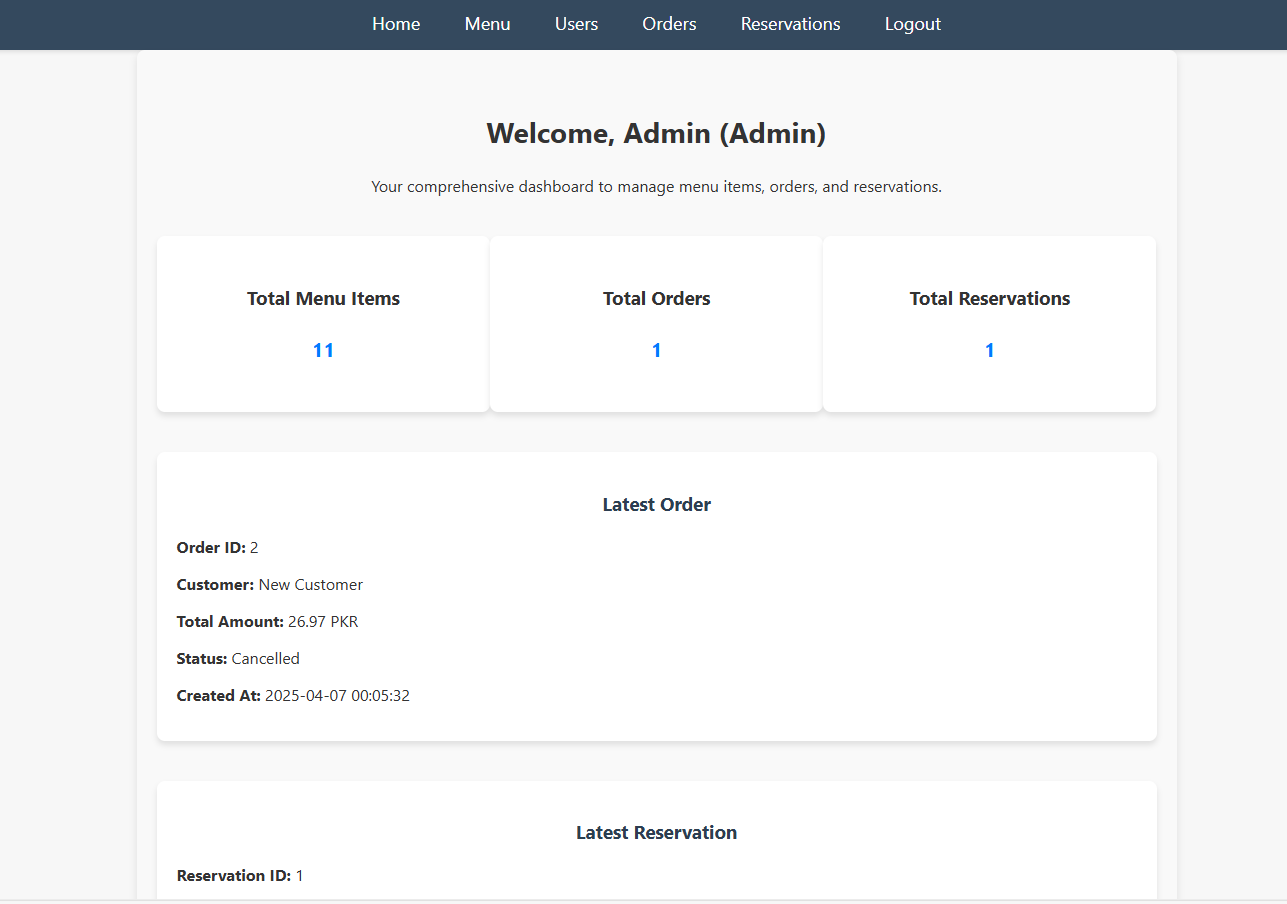
* **Home Page Interface: **
* **Contact Us form:**



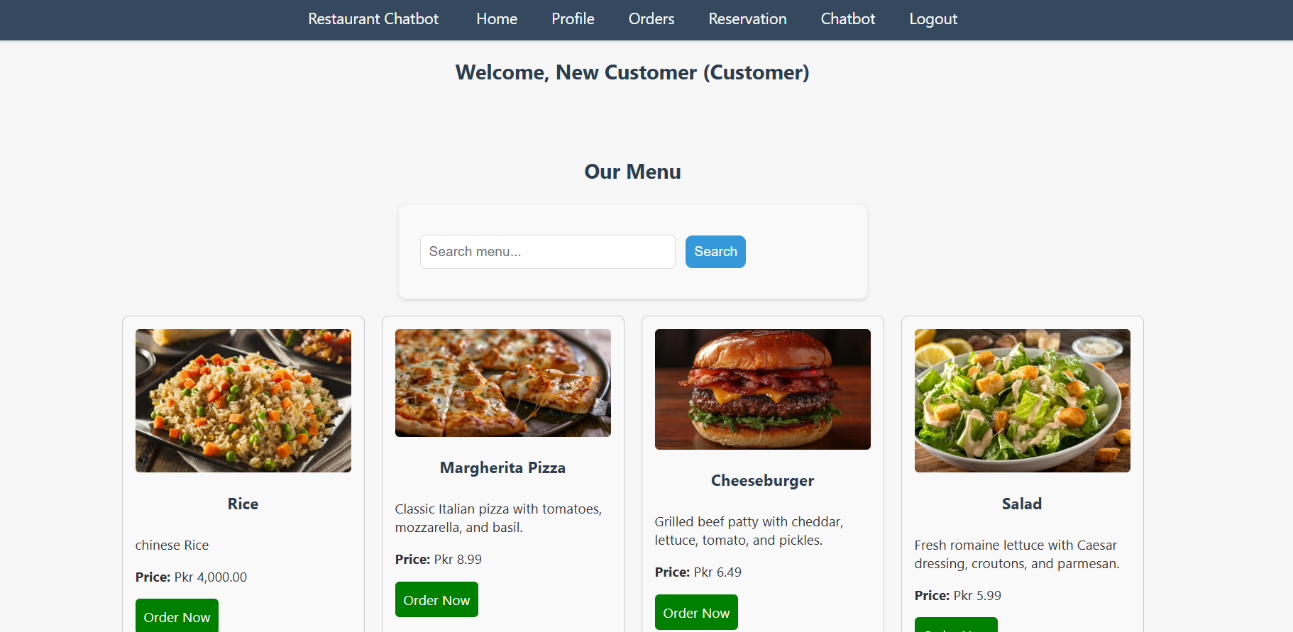
* **Customer Registration form:**

****

* **Admin Dashboard:**

****

* **Customer login interface:**

****

**CHAPTER 3**

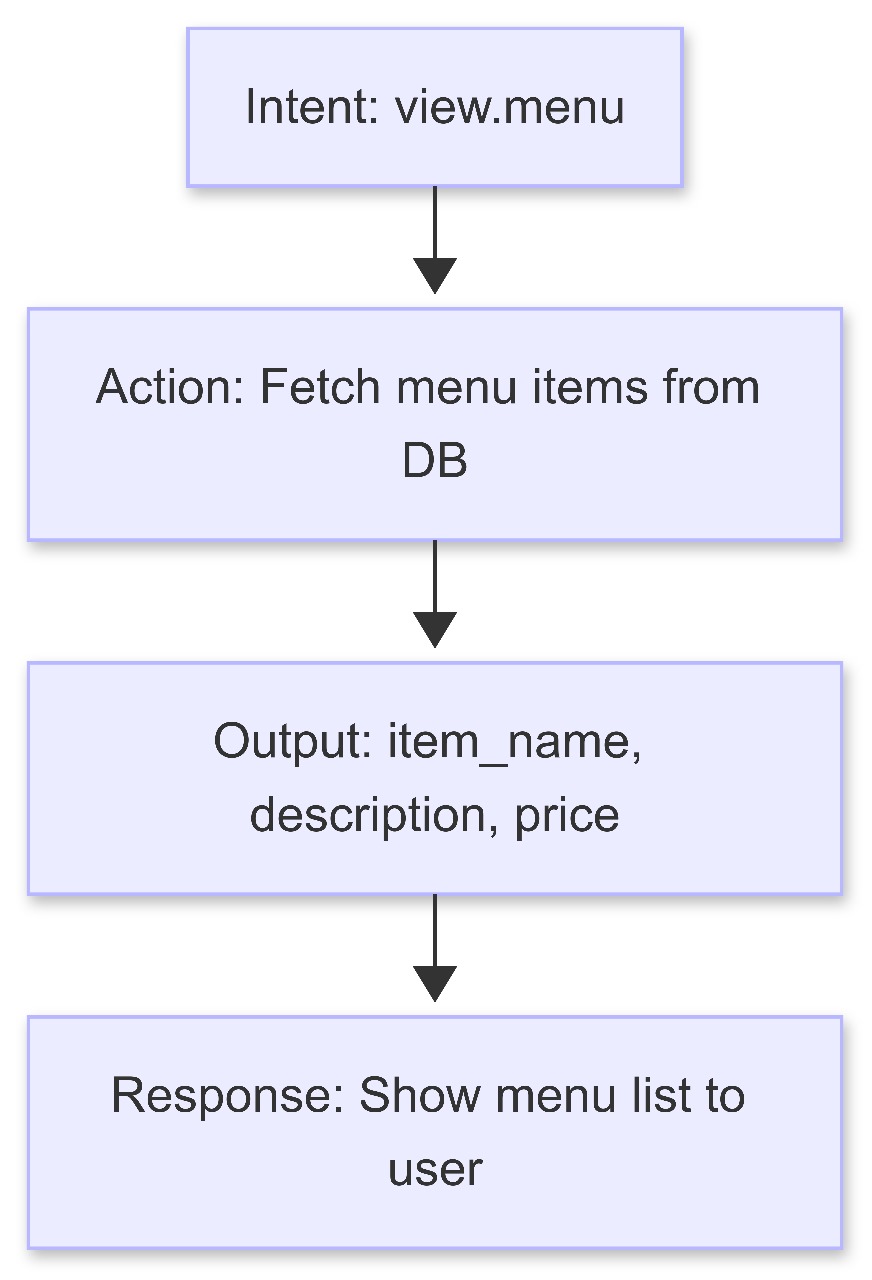
Development

# 3.1 Development plan (Architecture Diagram)

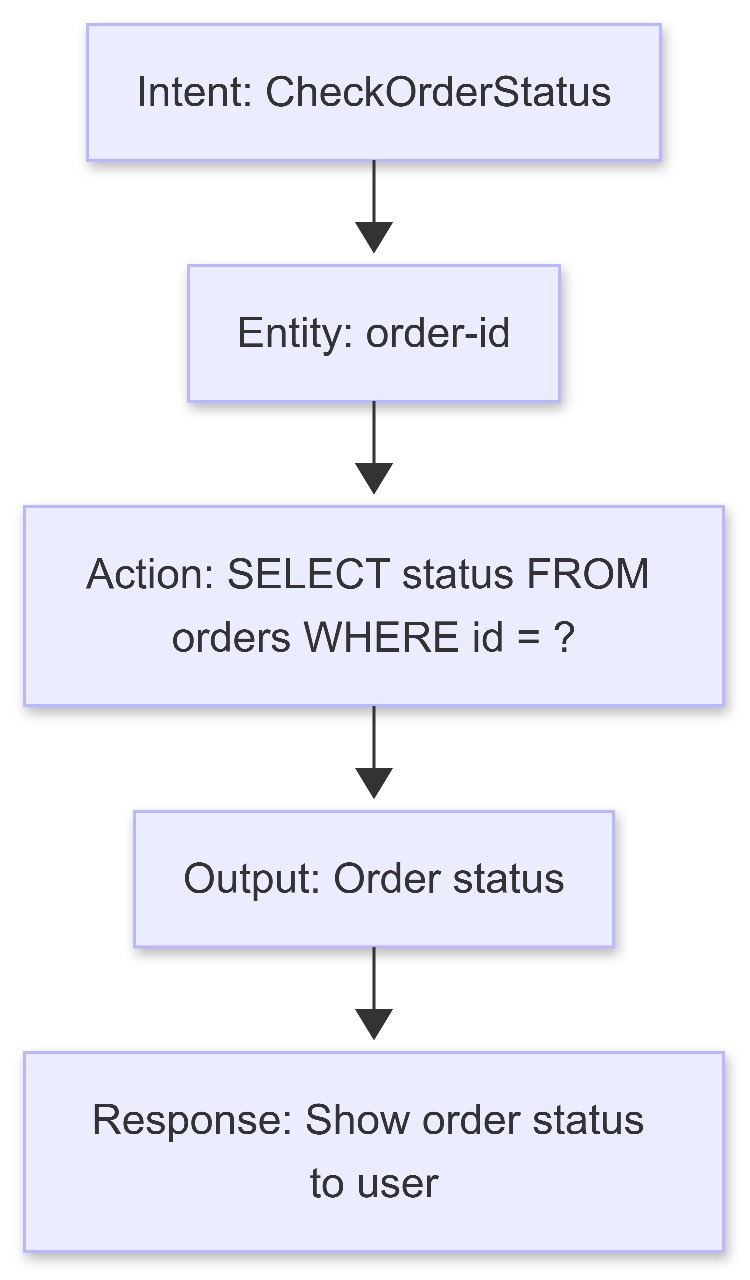


**Intent Relation diagram**

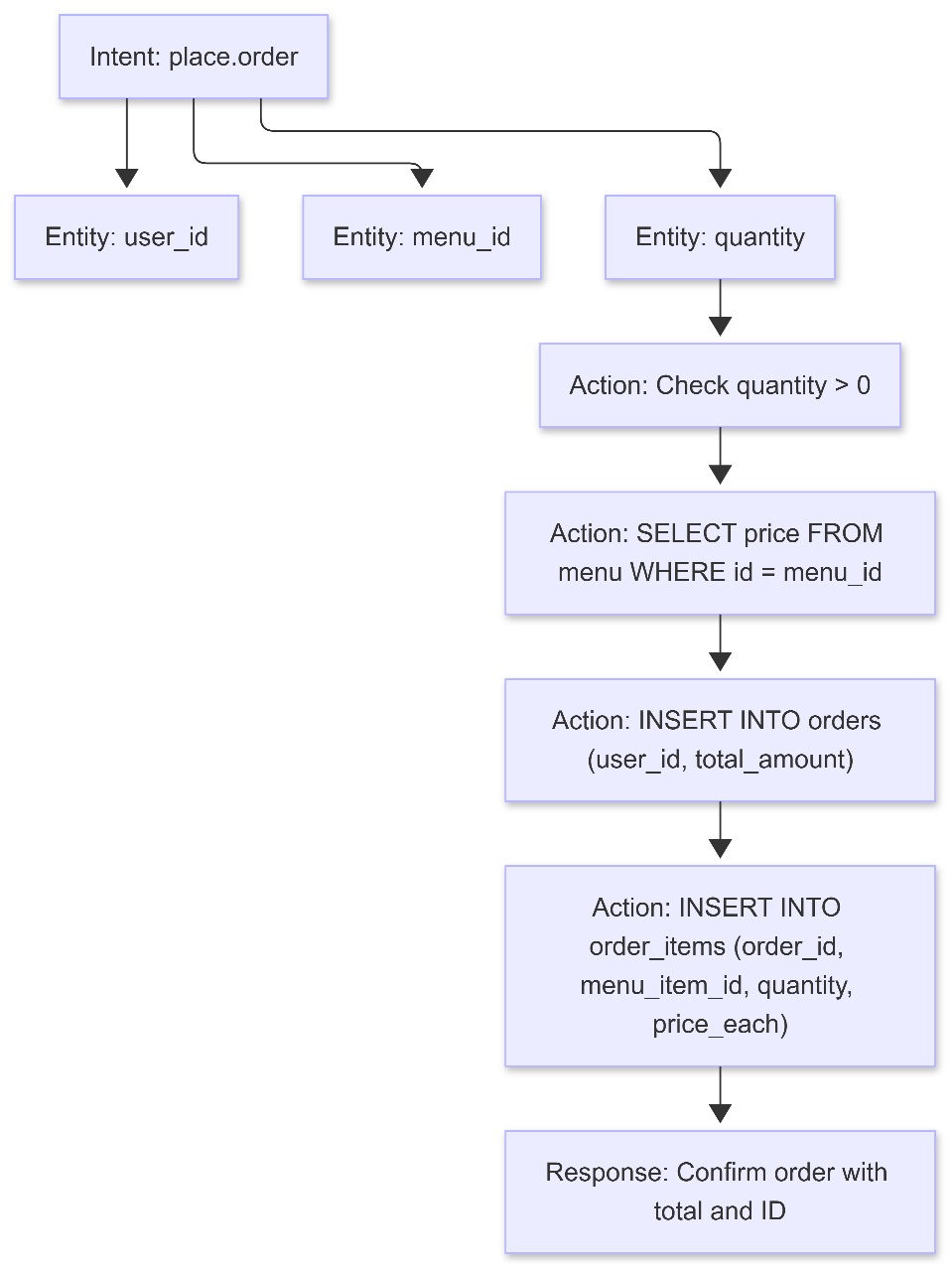
**Intent : view.menu**



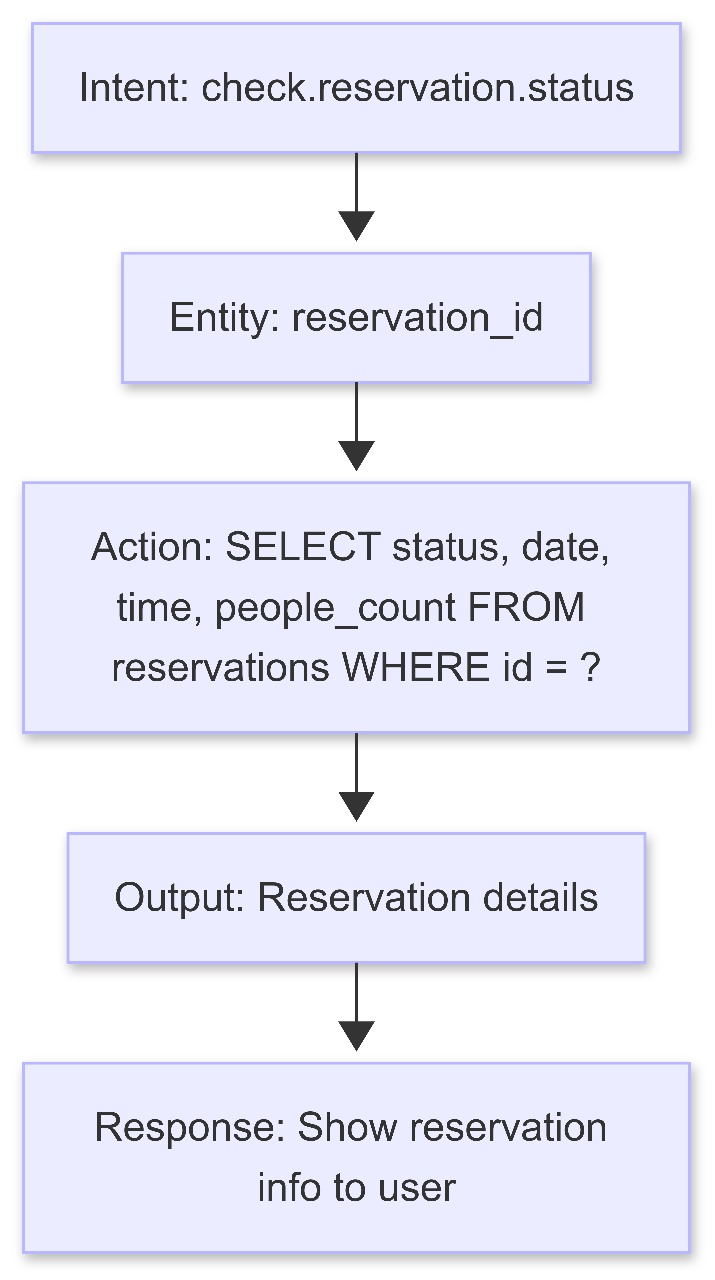
**Intent: CheckOrderStatus:**



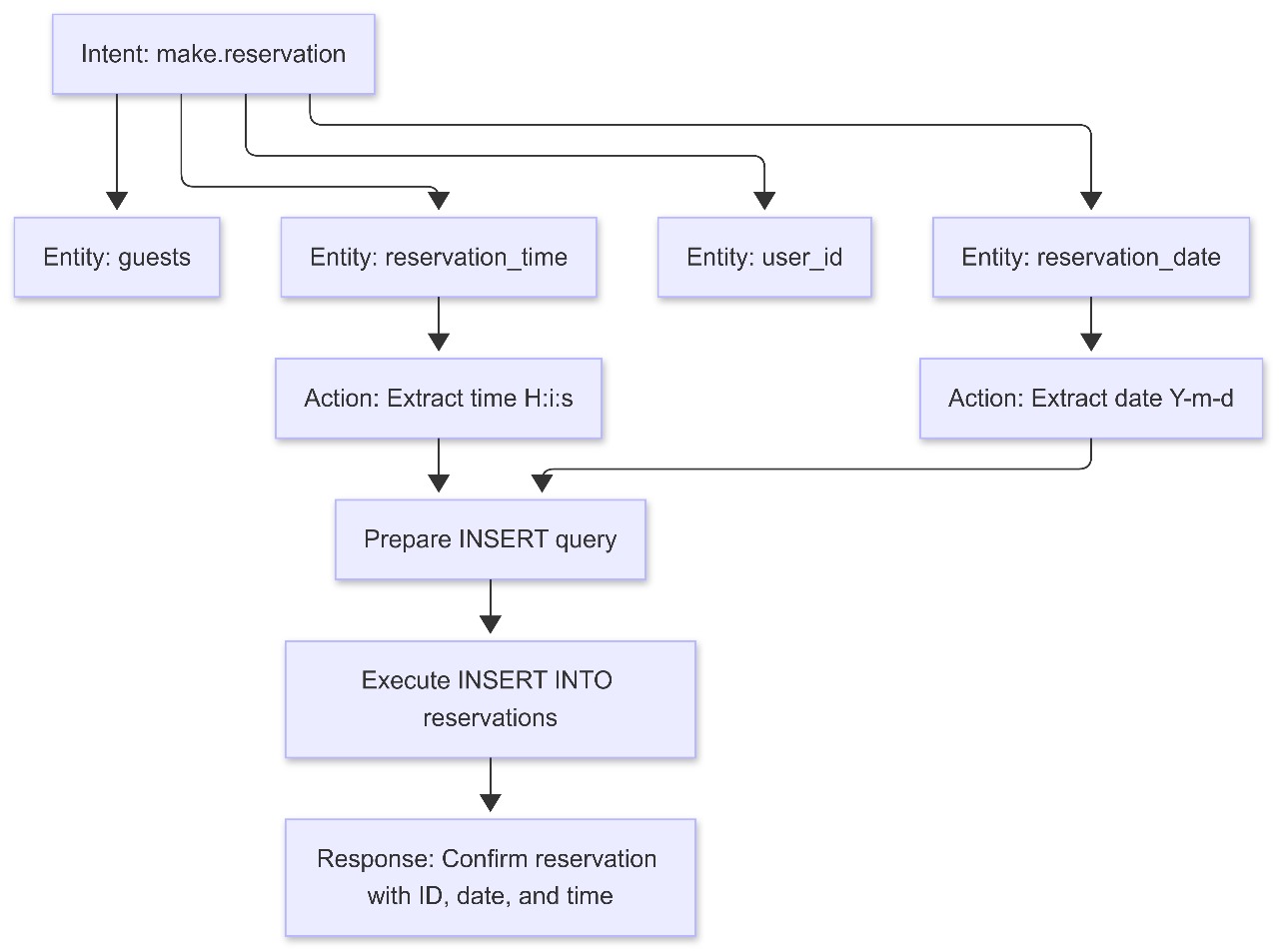
**Intent: Order.place:**



**Intent: Check.Reservation.Status**

****

**Intent: make.reservation**

****

**REFERENCES**

1. **Mozilla Developer Network (MDN) Web Docs:** developer.mozilla.org
2. **W3Schools:** w3schools.com
3. **Stack Overflow:** stackoverflow.com
4. **GitHub:** github.com
5. **"Clean Code" by Robert C. Martin**
6. **"HTML and CSS" by Jon Duckett**
7. **"JavaScript: The Good Parts" by Douglas Crockford**
8. **"Don't Make Me Think" by Steve Krug** (for usability)
9. **Google Developers:** developers.google.com
10. **CSS-Tricks:** css-tricks.com

**APPENDIX**

|  |  |
| --- | --- |
| **Section 1** | Gathering & Analyzing Information |
| **Section 2** | planning the Project |
| **Section 3** | Designing the Project |
| **Section 4** | Development |