|  |  |
| --- | --- |
| A purple neon sign on a brick wall  Description automatically generated  PURPLE LEAGUE  Project Plan | ABSTRACT  Thomas, Gilton G.J.  S2-CB08 |

Document Version History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Changes** | **Author** |
| **1.0** | 19/09/2024 | Initial document. | Thomas, Gilton |

Table of Contents

[Document Version History 0](#_bookmark0)

* 1. [Project Assignment 3](#_bookmark1)
  2. [Context 3](#_bookmark2)
  3. [Project Goal 3](#_bookmark3)
  4. [Deliverables 3](#_bookmark4)
  5. [Project Organization 3](#_bookmark5)
  6. [Client 3](#_bookmark6)
  7. [Team 4](#_bookmark7)
  8. [Communication 4](#_bookmark8)
  9. [Time Management 4](#_bookmark9)
  10. [Phasing 4](#_bookmark10)
  11. [Time Plan 4](#_bookmark11)
  12. [Gantt Chart 5](#_bookmark12)
  13. [Testing Strategies 6](#_bookmark13)
  14. [Testing Strategy 6](#_bookmark14)
  15. [Development Strategy 6](#_bookmark15)
  16. [Tools 6](#_bookmark16)
  17. [Constraints 6](#_bookmark17)
      1. [Technical Constraints 6](#_bookmark18)
      2. [Time Constraints 6](#_bookmark19)
  18. [Risk and Mitigation 7](#_bookmark20)

# Project Assignment

## Context With the increasing demand for interactive AI-like systems, this project aims to build a chatbot that offers real-time, rule-based conversation handling. Unlike AI-driven solutions, this chatbot will use predefined rules to simulate interaction, like an FAQ system, but with dynamic capabilities

## Project Goal

1. The chatbot solution involves two parts:
2. 1. A web-based chat interface: Users will interact with a responsive, real-time chat system using rules to trigger responses.
3. 2. A backend API: The API will manage user authentication, conversation handling, and CRUD operations on conversations. The system will use real-time notifications to inform users of responses.

## Deliverables

* + - A project plan document
    - A user stories specifications document
    - A web application
    - A database to store all the data .(sprint boot)

# Project Organization

## Client

The advisors on this project could be considered the clients since the advisors not only guide me through the individual assignment but also provide me with requirements like a client on a project. The contact information of the advisors can be found below:

|  |  |
| --- | --- |
| **Name** | **Position** |
|  |  |
|  |  |

|  |  |
| --- | --- |
|  |  |
|  |  |

## Team

#### The team working on this project contains a single member because it is an individual project. However, collaborators on this project consists of advisors, whom could be considered stakeholders. The contact information can be found below:

#### Name: Gilton Thomas Student ID: I493163

#### University e-mail: [I493163@fhict.nl](mailto:I493163@fhict.nl)

## Communication

As part of the individual assignment, I am required to hold a feedback session after each submission. Nonetheless, as part of the learning outcomes, it is expected that I will request for feedback once a week. These feedback sessions often take place on campus, but there is the possibility of holding these sessions online. We would make use of Microsoft Teams whenever these sessions take place online.

# Time Management

## Phasing

I have divided the individual assignment within four phases.

1. Phase 1: Foundations (Starting Week 1) – Set up project structure, authentication, and backend API.  
2. Phase 2: Core Development (Starting Week 3) – Develop rule-based chat logic and implement CRUD operations.  
3. Phase 3: Real-time Functionality (Starting Week 5) – Add WebSocket functionality for real-time notifications and response handling.  
4. Phase 4: Testing and Polishing (Starting Week 7) – Conduct tests, gather feedback, and refine the project.

## Time Plan

|  |  |
| --- | --- |
| **Duration** | **Activities** |
| **Laying the Foundations (September 12)** | |
| **Week 1** | Ideation document |
| **Week 2** | Project plan |
| **Week 3** | User stories |
| **Week 4 & 5** | Back end api /react front end |
| **Building the Framework (October 10)** | |
| **Week 6** | Testing and final adjustments |

## 

# Testing Strategies

## Testing Strategy

The testing will occur throughout the phases:  
- Unit tests: Each feature will have automated unit tests to verify functionality.  
- Functional testing: Ensure each component works together, especially the chatbot's rule-based responses and real-time notifications.  
- User acceptance testing: Validate with real users to simulate interaction and gather feedback

# Development Strategy

The development process will follow an agile methodology with regular sprint reviews to refine features and implement feedback.

### Tools

- Frontend: React.js  
- Backend: Spring Boot  
 - Database: MySQL   
- Real-time Notifications: WebSockets   
 - Authentication: JWT tokens with Spring Security

# Constraints

### Technical Constraints

The project will use rule-based responses, without natural language processing (NLP).  
The chat system will be limited to predefined rules for interaction

### Time Constraints

The project has a period of 16 weeks , to document, design, integrate, test the entire project, and deliver it to the client.

|  |  |  |
| --- | --- | --- |
| **Risk** | **Mitigation** | **Chance of Occurrence** |
| Development delays due to unforeseen technical challenges. | Allocating additional time for complex technical tasks. | Low |
| Inadequate resources for testing and quality assurance. | Regular checkpoints to ensure the project remains on schedule. | Low |
| Scope creep due to expanding project requirements. | Clearly defined project boundaries and confirmed with stakeholders. | Mid |