CEBU INSTITUTE OF TECHNOLOGY
UNIVERSITY

IT342-Section
SYSTEMS INTEGRATION AND ARCHITECTURE 1

FUNCTIONAL REQUIREMENTS SPECIFICATION (FRS)

Project Title: TBA

Prepared By: Russjie G. Hopista

Date of Submission: Feb 6, 2025

Version: 1

Table of Contents

1. Introduction

1.1. Purpose

The purpose of this system is to provide a robust authentication and authorization framework. The intended audience includes software developers maintaining the codebase and project stakeholders interested in the security lifecycle of a user session.

- This application aims to Register, Login, and Logout a user as well as provide a dashboard page and profile page.

1.2. Scope

The system focuses on secure identity management. It handles the transition from an anonymous "Guest" to an "Authenticated User" through a dual-token (JWT + Refresh Token) system.

- The system focuses on the basic features of an application: login, register, dashboard, profile, and logout.

1.3. Definitions, Acronyms, and Abbreviations

- **JWT (JSON Web Token):** The token used for stateless authentication.
- **Refresh Token:** A database-backed "Backup Token" used to renew sessions.
- **DTO (Data Transfer Object):** A simple object used to pass data between layers.

2. Overall Description

2.1. System Perspective

The system is a standalone authentication module integrated into a simple web or mobile application.

It follows a layered architecture consisting of:

- Controller Layer (handles user requests)
- Service Layer (business logic for authentication)
- Repository Layer (database access)
- Security Components (JWT Token Provider and Password Encoder)

The system communicates with a database to store user information and refresh tokens while using JWT for secure stateless authentication.

2.2. User Classes and Characteristics

Guest User

- Users who have not registered or logged in
- Can only access the home screen and registration/login pages

Registered User (Authenticated User)

- Users who have successfully logged in
- Can access dashboard and profile pages
- Can logout and manage their session

System Administrator (Optional / Future)

- Manages users and system security (if extended later)

2.3. Operating Environment

- The system will operate in the following environment:
- **Software:**
  - Backend: Java (Spring Boot or similar framework)
  - Database: MongoDB
  - Frontend: ReactJS
- **Tools:**
  - IDE (IntelliJ, VS Code, Eclipse)
  - API testing tools (Postman)

2.4. Assumptions and Dependencies

Assumptions:

- Users have internet access
- Users provide valid credentials during registration
- The system is hosted on a secure server

Dependencies:

- Database server availability
- JWT library for token generation
- Password encryption library
- Stable internet connection

3. System Features and Functional Requirements

Describe each major feature of the system and its functional requirements.

3.1. Feature 1: Register

Description:
Allows new users to create an account by providing personal and login information.

Functional Requirements:

- The system shall allow users to enter username, email, and password
- The system shall validate user input
- The system shall securely store encrypted passwords in the database
- The system shall prevent duplicate user accounts

3.2. Feature 2: Login

Description:
Allows registered users to log in and receive authentication tokens.

Functional Requirements:

- The system shall authenticate users using username/email and password
- The system shall generate a JWT access token upon successful login
- The system shall generate and store a refresh token
- The system shall deny access for invalid credentials

3.3. Feature 2: Dashboard Access

Description:
Provides authenticated users access to the main application screen.

Functional Requirements:

- The system shall verify JWT token before granting access
- The system shall display user-specific information
- The system shall block unauthorized users

3.4. Feature 2: Profile Access

Description:
Allows users to view their profile information.

Functional Requirements:

- The system shall retrieve user data from the database
- The system shall display profile details securely
- The system shall require authentication to access profile

3.5. Feature 2: Logout

Description:
Ends the user session securely.

Functional Requirements:

- The system shall invalidate refresh tokens
- The system shall remove session access
- The system shall redirect user to home or login page

4. Non-Functional Requirements

Performance

- The system shall process login requests within 2 seconds
- The system shall support multiple users simultaneously

Security

- Passwords shall be encrypted
- JWT tokens shall have expiration time
- Refresh tokens shall be stored securely

Usability

- The interface shall be simple and easy to navigate
- Users shall receive clear error messages

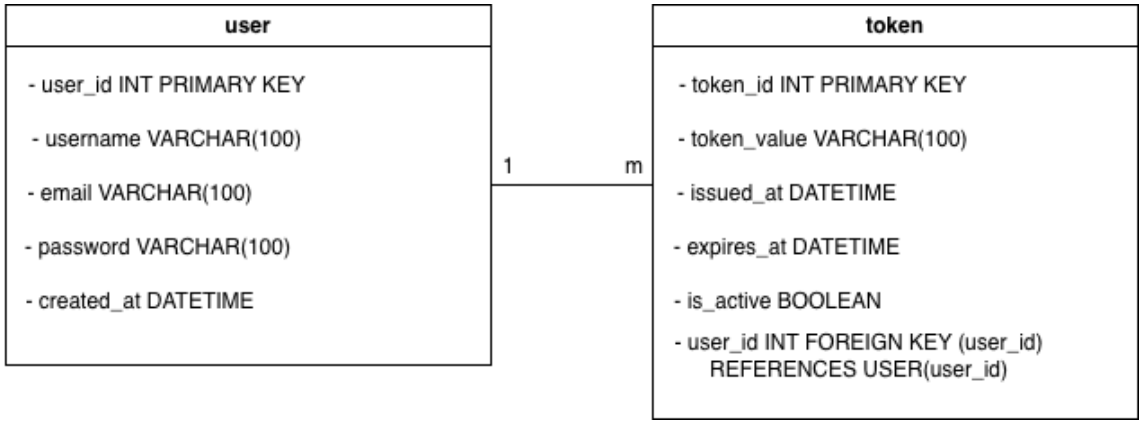
Reliability

- The system shall maintain session consistency
- The system shall handle token expiration properly

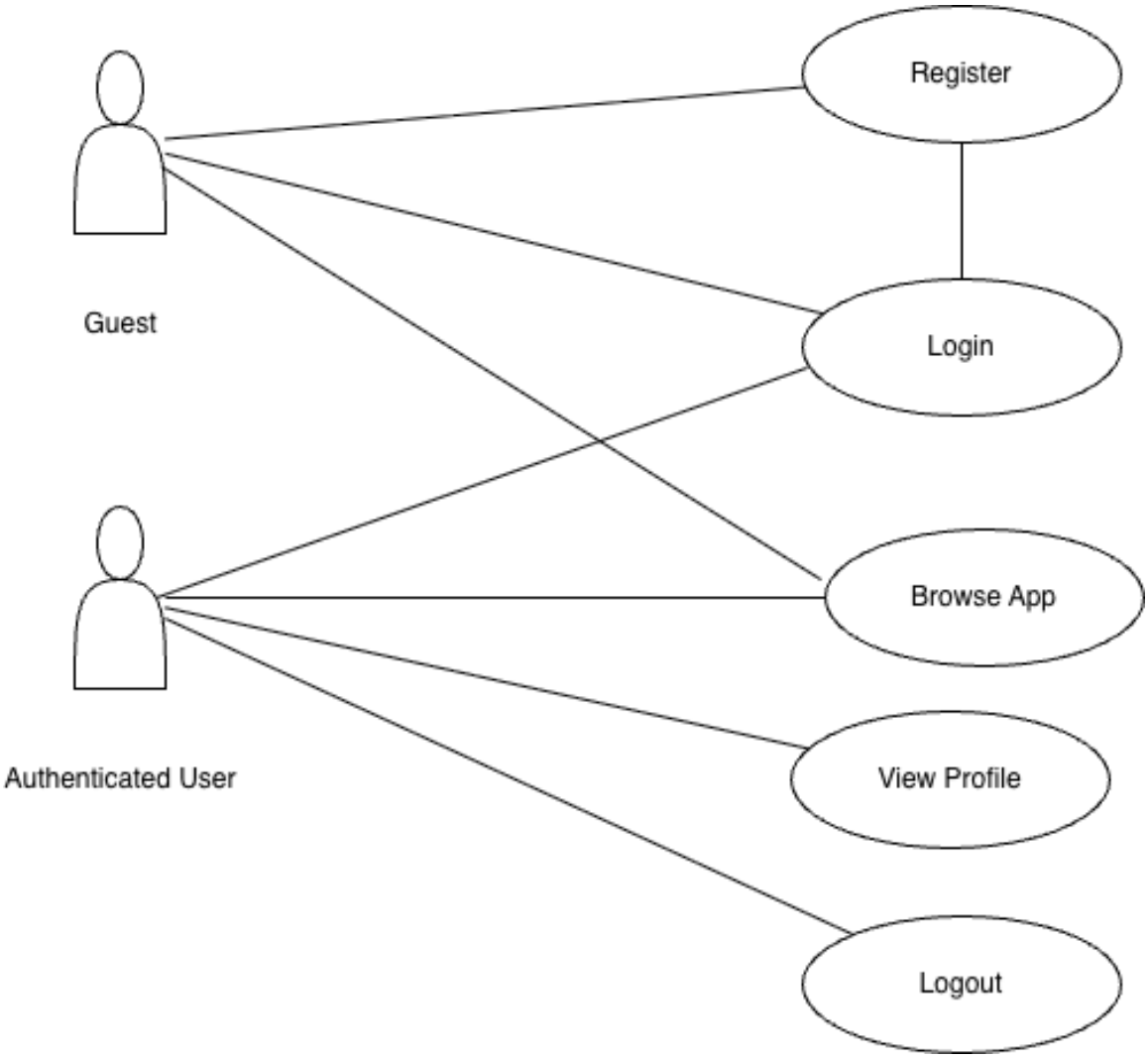
5. System Models (Diagrams)

Insert the necessary diagrams for the system:

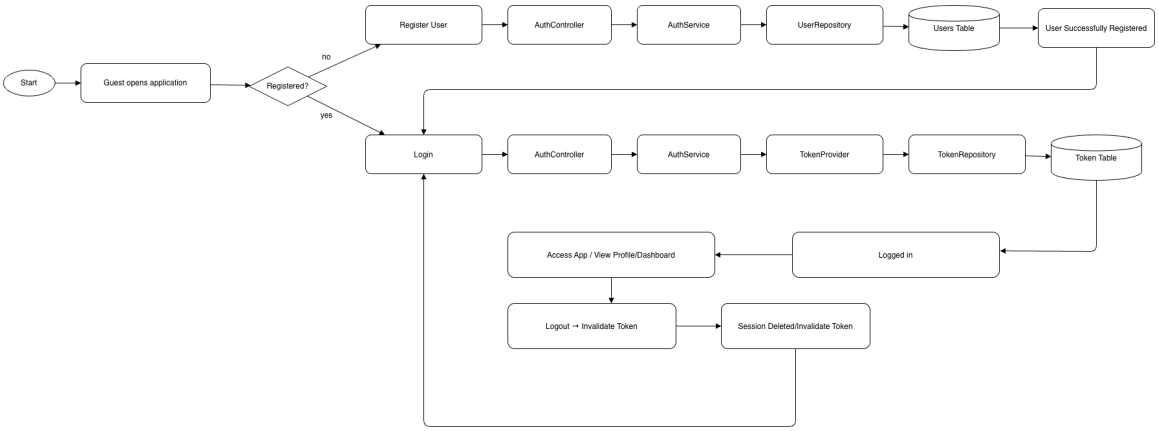
5.1. ERD



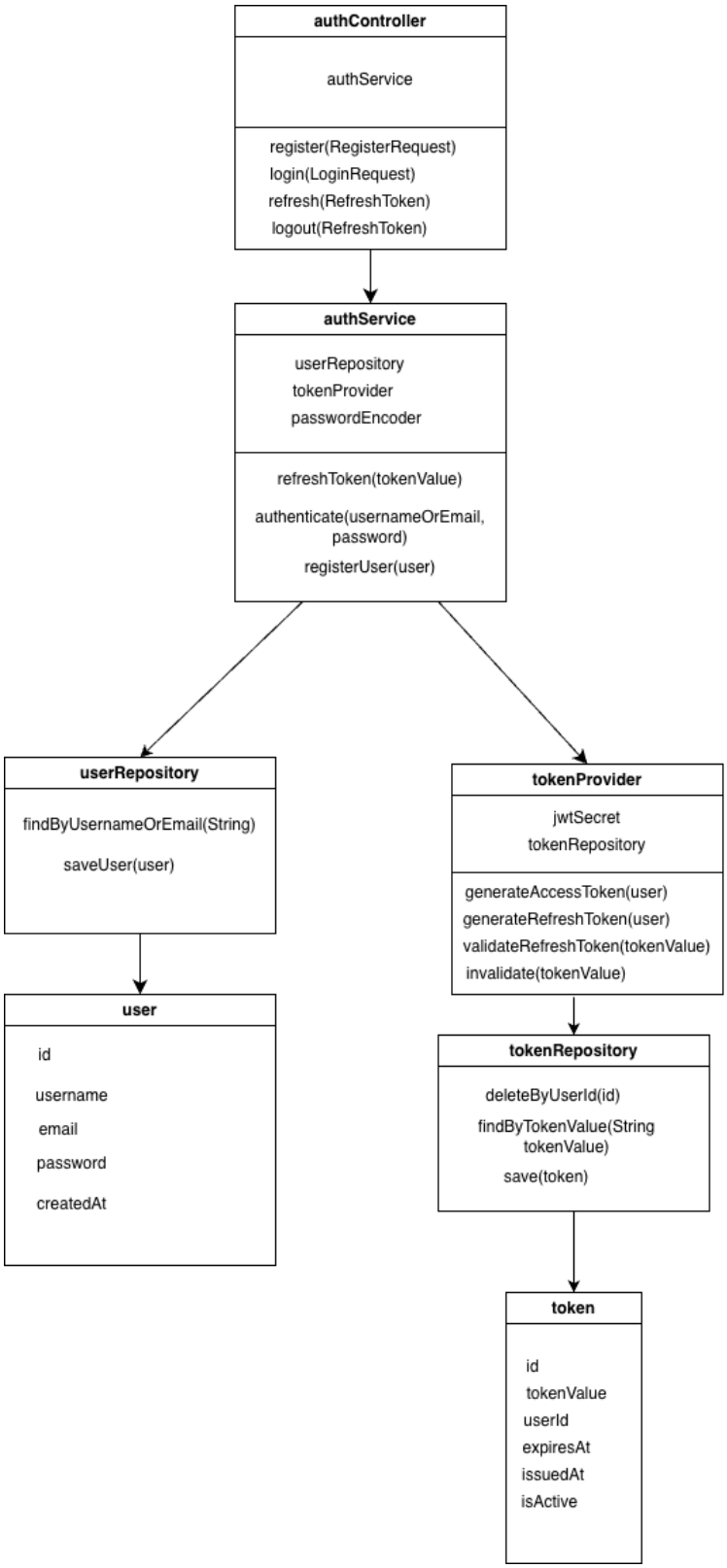
5.2. Use Case Diagram



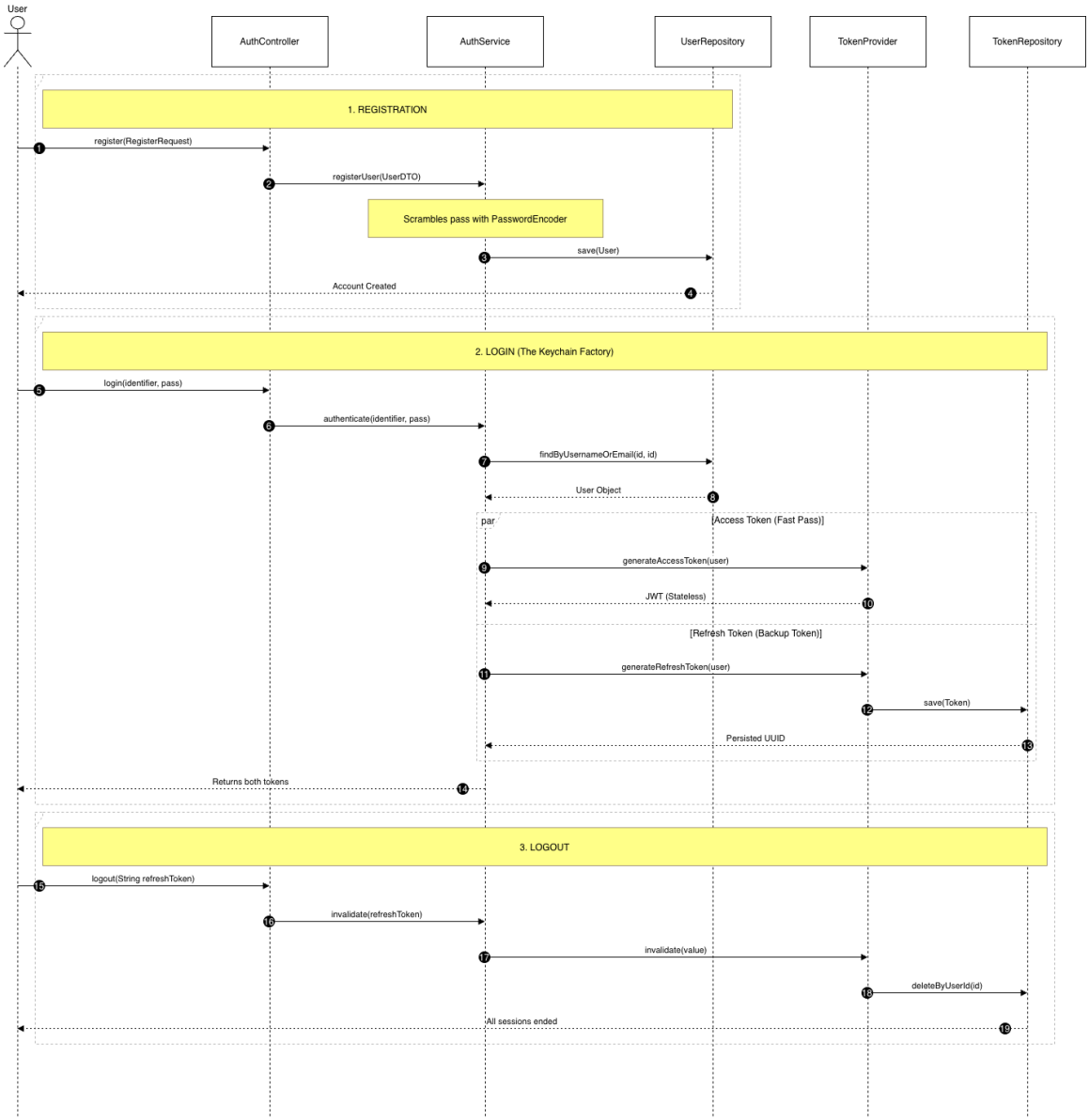
5.3. Activity Diagram



5.4. Class Diagram



5.5. Sequence Diagram



6. Appendices

Include any additional information, references, or support materials.