**Software Requirements Specification (SRS) for Time Tracking App**

**1. Introduction**

**1.1 Project Overview**

A **Time Tracking App** is a crucial tool designed to improve time management, boost productivity, and enhance project tracking. This application helps **admins, project managers, and developers** monitor task durations, identify productivity trends, and optimize workflow efficiency.

**1.2 Background**

Time management is essential for **software development teams** to meet deadlines, allocate resources effectively, and ensure timely project completion. This system streamlines task assignments, records time spent, and generates insightful reports to facilitate better decision-making.

**1.3 Purpose**

**1.3.1 Problem Statement**

Managing time and tracking productivity manually can be inefficient and error-prone. A centralized system is needed to automate tracking and reporting.

**1.3.2 Project Aim**

The aim of this project is to develop an efficient time-tracking system that provides detailed reports on time usage and assists in identifying bottlenecks in the development process.

**1.3.3 Project Objectives**

* Enable **project managers** to assign tasks and track progress.
* Allow **developers** to log their working hours.
* Provide **admins** with analytical reports and insights.
* Offer **visual charts** to monitor deadlines and productivity levels.

**1.4 Project Scope**

The system will:

* Allow project managers to **assign tasks** and monitor completion.
* Enable developers to **log their working hours**.
* Generate reports to analyze **time spent per project/task**.
* Provide **role-based access** for different users.

**1.5 Impact, Significance, and Contributions**

* Helps teams **manage workloads effectively**.
* Improves **transparency** in project timelines.
* Enhances **productivity tracking** and **performance evaluation**.

**1.6 Organization of Project Report**

This document outlines the **functional, non-functional requirements, user roles, system design, implementation strategies, and testing methodologies**.

**2. Literature Review**

*(To be added if necessary)*

**3. Software Requirements Specifications**

**3.1 Functional Requirements**

**3.1.1 User Management**

* Role-based access for **Admin, Project Manager, Developer**.
* User registration, login, and profile management.

**3.1.2 Task and Project Management**

* Project creation and task assignment.
* Task status updates and deadline management.

**3.1.3 Time Tracking**

* Automatic/manual time logging for assigned tasks.
* Task completion time calculation.

**3.1.4 Reports and Analytics**

* Generate **project-wise reports**.
* Provide **charts and graphs** for productivity insights.

**3.2 Non-Functional Requirements**

**3.2.1 Performance Requirements**

* The system should handle **simultaneous tracking** for multiple users.

**3.2.2 Security Requirements**

* Secure authentication and role-based access control.

**3.2.3 Usability Requirements**

* Intuitive UI with easy navigation.

**3.2.4 Reliability Requirements**

* Ensure **accurate time logs** and prevent data loss.

**3.2.5 Maintainability Requirements**

* Modular architecture for **scalability and updates**.

**4. User Classes and Characteristics**

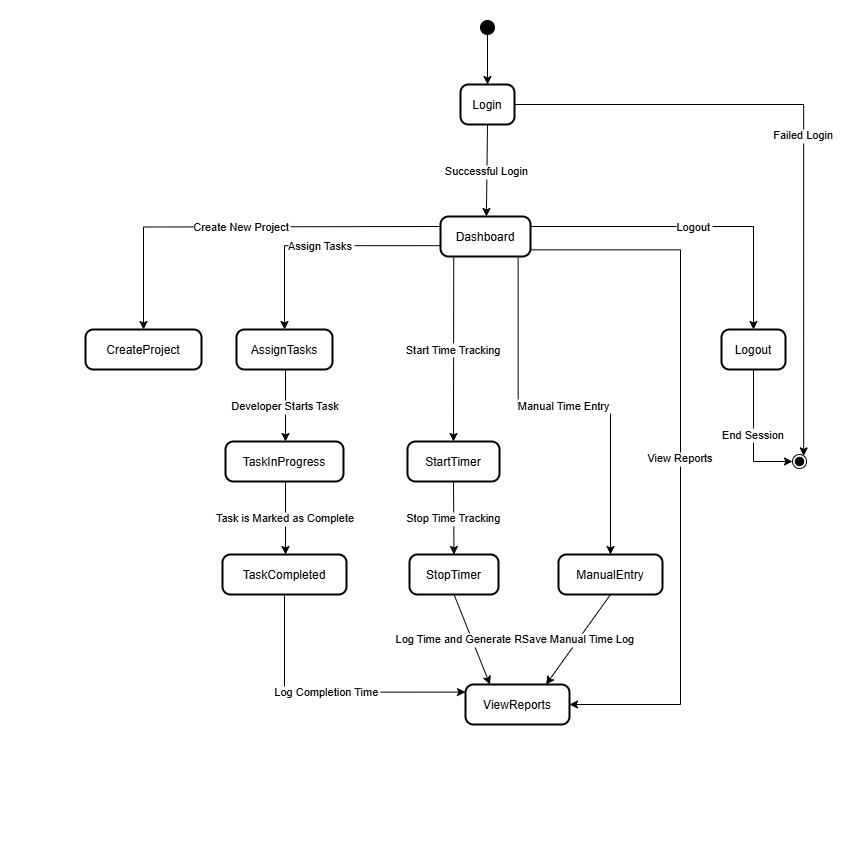
**4.1 Users**

* **Admin** – Oversees entire system and generates reports.
* **Project Manager** – Assigns tasks and tracks developer progress.
* **Developer** – Logs work hours and updates task status.

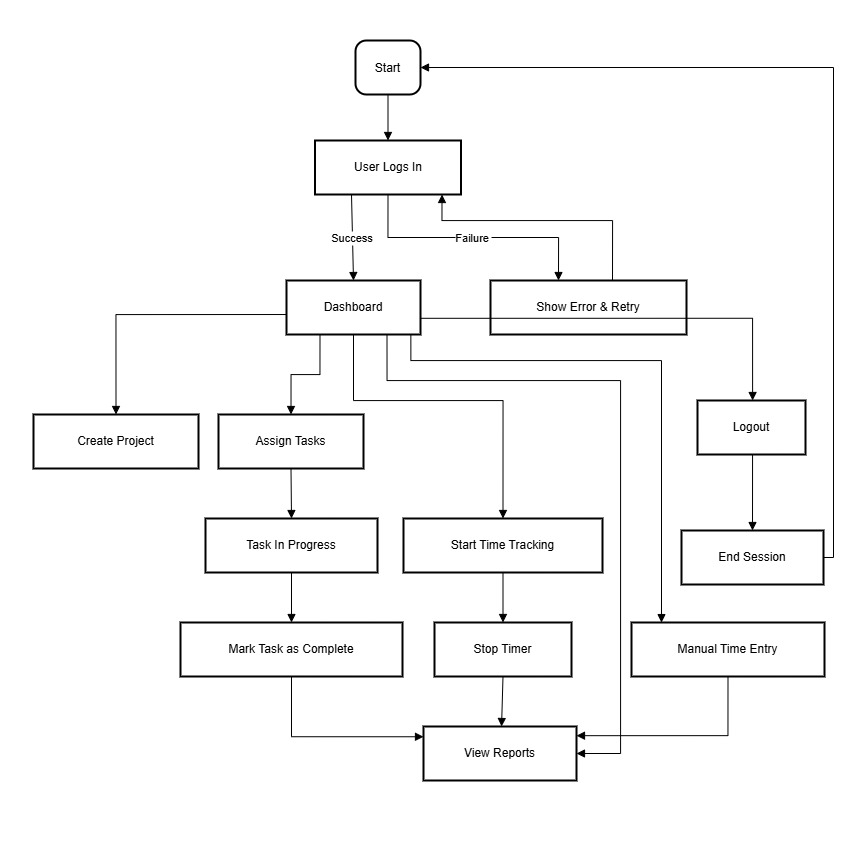
**5. Technology Stack**

* **Frontend**: React.js, Vite
* **Backend**: Python FastAPI
* **Database**: MongoDB
* **Server**: Django Server

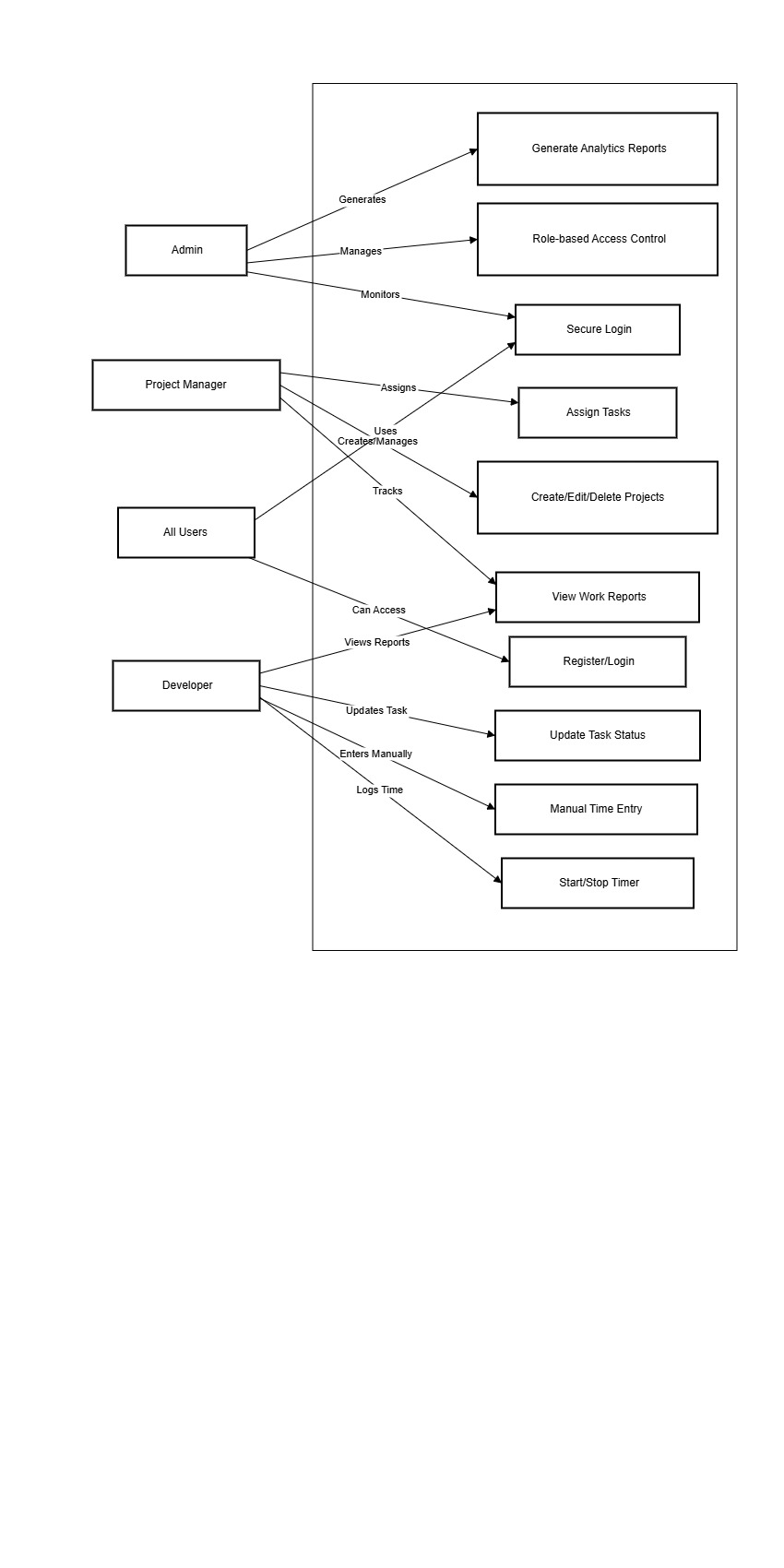
**6. System Design**

**6.1 State Diagram **

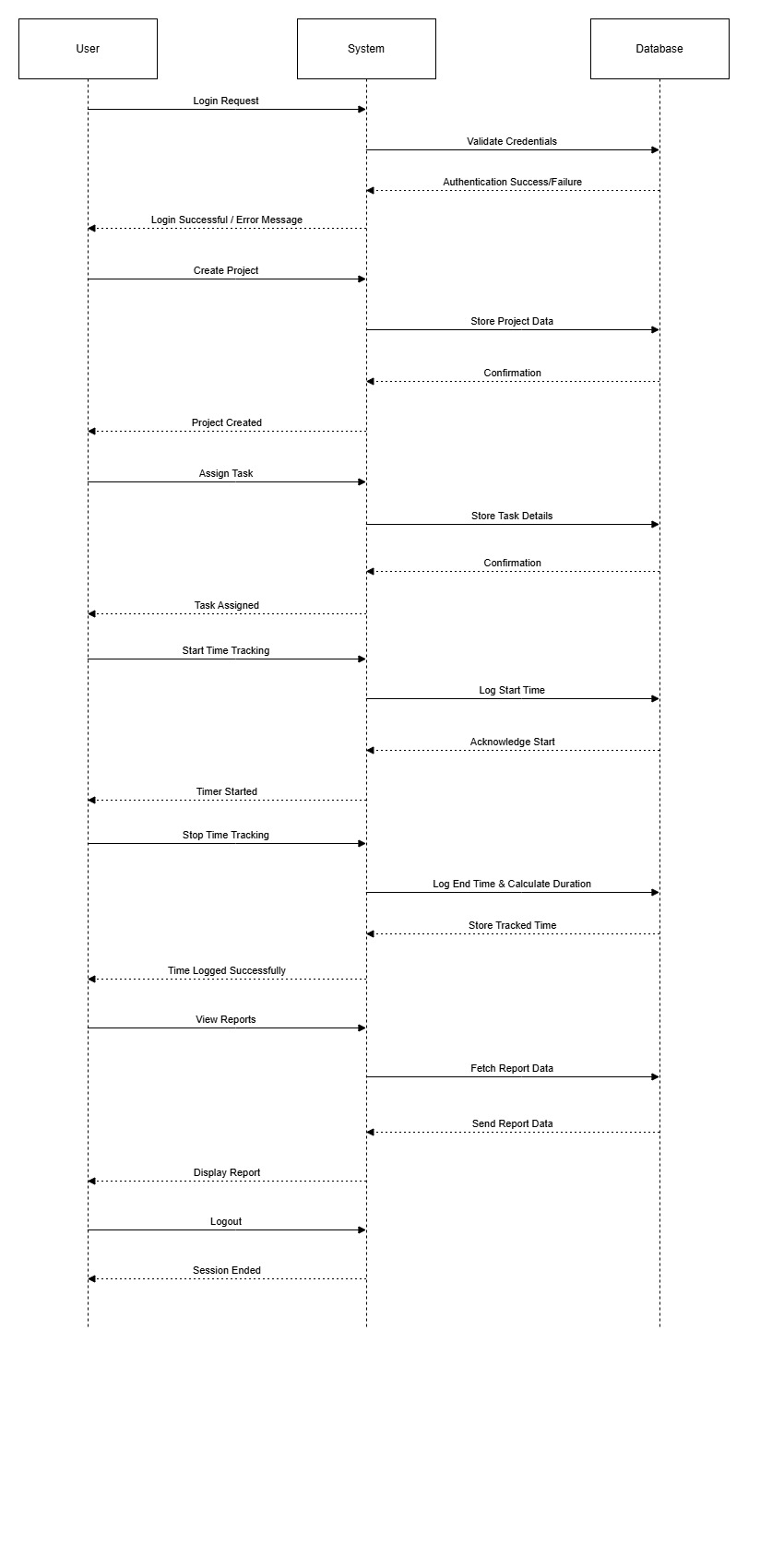
**6.2 Activity Diagram**

****

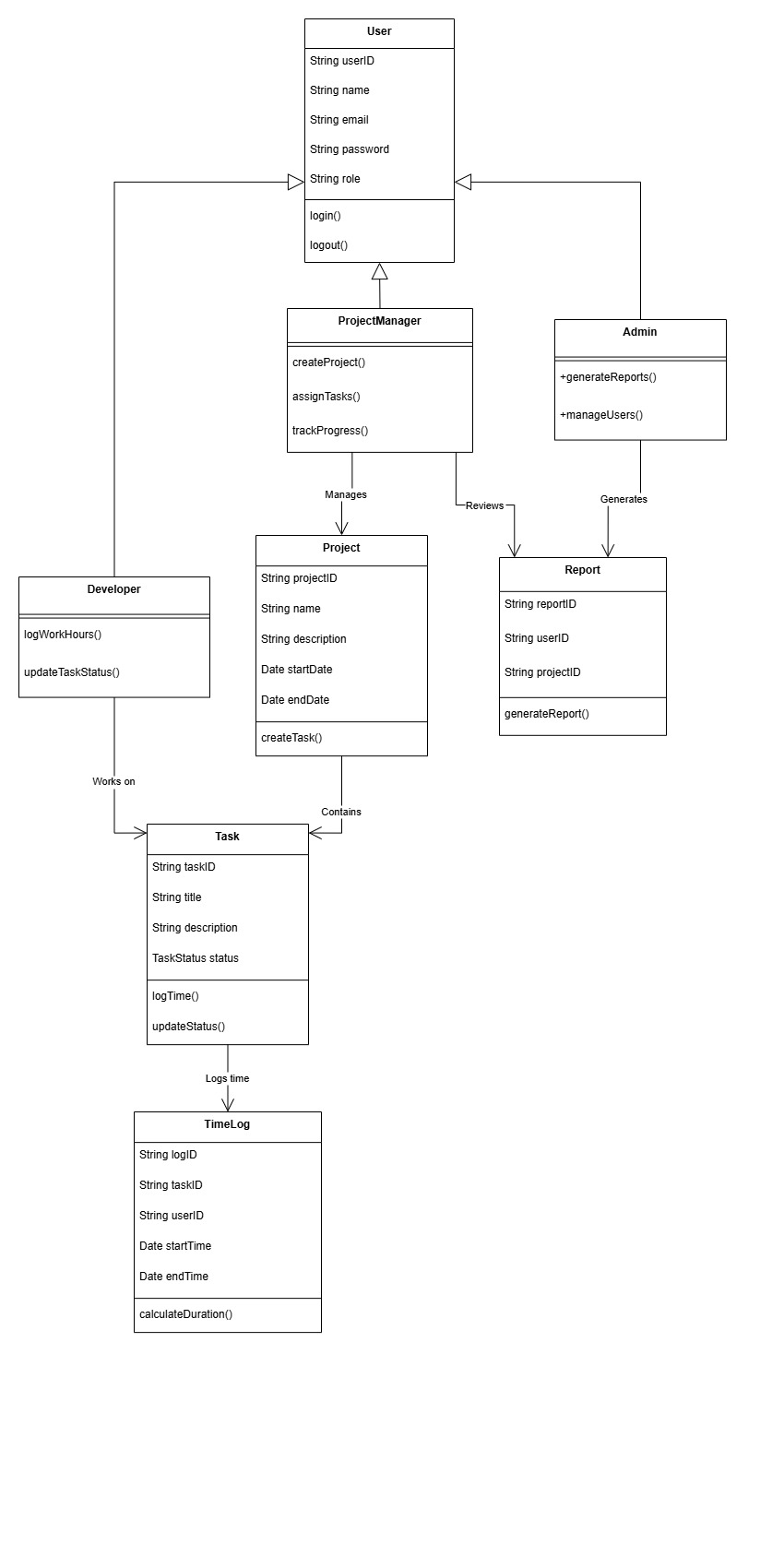
**6.3 Use Case Diagram**

****

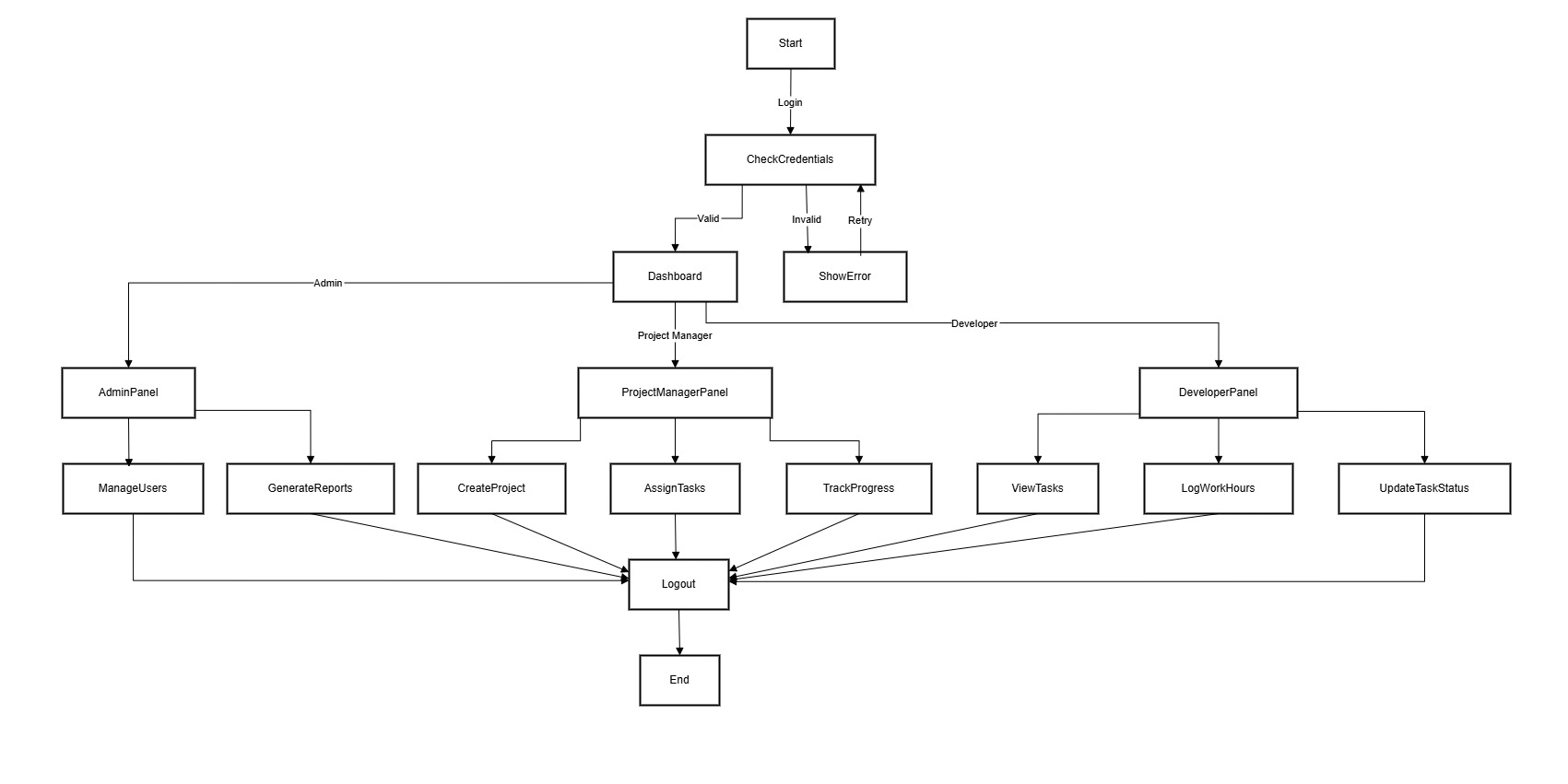
**6.4 Sequence Diagram**

****

**6.5 Class Diagram**

****

**6.6 Flowchart Diagram**

****

**7. Prototype *(UI Screens To Be Designed)***

**8. Process Model**

**8.1 Iterative Model *(To be added)***

**9. Project Plan**

**9.1 Objective *(To be added)***

**9.2 Project Overview and Duration *(To be added)***

**9.3 Scope of the Project *(To be added)***

**9.4 Milestones and Timeline *(To be added)***

**10. Implementation *(To be added)***

**11. Testing *(To be added)***

**11.1 Unit Testing**

**11.2 Integration Testing**

**11.3 Regression Testing**

**11.4 Performance Testing**

**11.5 Security Testing**

**11.6 User Acceptance Testing (UAT)**

**11.7 Continuous Testing & Monitoring**

**12. User Manual *(To be added)***

**13. Conclusion and Future Work**

**13.1 Conclusion *(To be added)***

**13.2 Future Enhancements *(To be added)***

**14. Annexure *(To be added)***