# Report 1: Project Proposal Report (Initiation Phase)

## Project Title

* Equipment Rental Management System

## Team Members and Roles

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Student ID | Student Name | Role |
|  | HE194350 | Phan Quyết Chiến | ALL |

## Objectives and Expected Outcomes

* **Objectives:**

 Develop a desktop application to manage equipment rentals efficiently.

 Provide role-based access for Admin and Staff.

 Support core operations such as rental contracts, equipment returns, stock tracking, and invoice generation.

 Ensure data consistency and scalability through a 3-layer architecture.

* **Expected Outcomes**:   
   **Admin:** Manage inventory, view reports/statistics, manage accounts.

 **Staff:** Create rental contracts, return equipment, update rental status.

 A fully functional Windows Desktop App with a clean and intuitive interface.

## Scope and Technical Requirements

* **Include Scope:**

 Authentication with role-based access (Admin, Staff).

 Equipment inventory management.

 Rental contract creation and return processing.

 Invoice generation and statistics reporting.

 Data persistence using SQL Server.

* **Exclude Scope:**

 No mobile/web interface in the current phase.

 No payment integration.

* **Technical Requirements:**

 **Language:** C#

 **IDE:** Visual Studio 2022

 **Database:** SQL Server Management Studio

 **Architecture:** 3-layer

 **Framework:** .NET 8 (or .NET Core if more suitable)

 **Tools:** Git, GitHub

## Initial Implementation Plan

|  |  |  |
| --- | --- | --- |
| **Phase** | **Main Activities** | **Estimated Time** |
| |  | | --- | | Requirement Analysis |  |  | | --- | |  | | |  | | --- | | Gather and define system requirements |  |  | | --- | |  | | |  | | --- | | 3-5 days |  |  | | --- | |  | |
| |  | | --- | | Database Design |  |  | | --- | |  | | |  | | --- | | Design ERD and create necessary tables |  |  | | --- | |  | | |  | | --- | | 3 days |  |  | | --- | |  | |
| |  | | --- | | Project Setup |  |  | | --- | |  | | |  | | --- | | Create 3-layer project and login screen |  |  | | --- | |  | | |  | | --- | | 1 day |  |  | | --- | |  | |
| |  | | --- | | Owner Functions |  |  | | --- | |  | | |  | | --- | | CRUD operations for equipment, users, categories |  |  | | --- | |  | | |  | | --- | | 2 days |  |  | | --- | |  | |
| |  | | --- | | Staff Functions |  |  | | --- | |  | | |  | | --- | | Create/return rental contracts, manage invoices |  |  | | --- | |  | | |  | | --- | | 2 days |  |  | | --- | |  | |
| |  | | --- | | Report Module |  |  | | --- | |  | | |  | | --- | | Generate statistical reports |  |  | | --- | |  | | |  | | --- | | 2 days |  |  | | --- | |  | |
| |  | | --- | | Testing & Debugging |  |  | | --- | |  | | |  | | --- | | System test and bug fixes |  |  | | --- | |  | | |  | | --- | | 3 days |  |  | | --- | |  | |
| |  | | --- | | Finalization |  |  | | --- | |  | | |  | | --- | | Prepare report, UI polish, deployment |  |  | | --- | |  | | |  | | --- | | 5 days |  |  | | --- | |  | |

**Responsibilities:**

* All members are responsible for code implementation, UI design, and documentation.

## Resources and Tools

* **Tool & Frameworks:**

 IDE: Visual Studio 2022

 Framework: .NET (WPF)

 Database: SQL Server

 Version Control: Git, GitHub

**Initial Setup:**

* Create a GitHub repo named **EquipmentRentalManagementApp**
* SQL database EquipmentRentalManagement with table creation scripts.

Install necessary libraries and tools (Entity Framework, etc.).

## Risk Assessment

|  |  |  |
| --- | --- | --- |
| **Risk** | **Level** | **Mitigation measures** |
| |  | | --- | | Incomplete requirement specification |  |  | | --- | |  | | Medium | |  | | --- | | Collect use cases early and validate with mock data |  |  | | --- | |  | |
| |  | | --- | | Poor database design |  |  | | --- | |  | | Medium | |  | | --- | | Design ERD before implementation, get feedback early |  |  | | --- | |  | |
| |  | | --- | | Time constraints |  |  | | --- | |  | | Medium | |  | | --- | | Split tasks with short deadlines and integrate frequently |  |  | | --- | |  | |

## References (Optional)

* Microsoft Learn (.NET): <https://learn.microsoft.com/en-us/dotnet/>
* GitHub Actions Guide: <https://docs.github.com/en/actions>
* SQL Server Docs: <https://docs.microsoft.com/en-us/sql/sql-server/>