Software Design Document

ISO Audit Software

Version 1.0

Prepared by:

1. Abhignya Kotha (PES1UG21CS018)
2. Adithya B (PES1UG21CS036)
3. Aditi Prabhu A (PES1UG21CS039)
4. Ambati Revanth Sreeram (PES1UG21CS070)

Submitted to: Anand M S

Lecturer

October 27, 2023

Revision History

Version	Primary Author	Description of Version	Date Completed
1.0	The Distinct Anomalies	Initial Release	27/10/2023

Table of Contents

1. Introduction

1.1 Purpose

This Sample Design Document (SDD) is created to outline the architectural and design considerations for the ISO Audit Management Software. The document provides a comprehensive overview of the software's design principles, system components, and the rationale behind design decisions. It serves as a reference for developers, architects, and project stakeholders, offering insights into the software's design aspects.

1.2 Scope

The SDD covers the design aspects of the ISO Audit Management Software, including architectural design, data models, user interface design, and the design of core components and modules. It presents an organised framework for understanding the software's design choices, ensuring that the development process aligns with the intended design principles.

1.3 Definitions, Acronyms and Abbreviations

- **SDD**: Sample Design Document
- **API**: Application Programming Interface
- **UI**: User Interface
- **ISO**: International Organization for Standardization
- **RBCA:** Role Based Access Control
- **RTM:** Requirements Traceability Matrix

1.4 References

- ISO 19011:2018 Guidelines Document
  https://www.iso.org/standard/70017.html
- Software Requirements Specification

https://drive.google.com/file/d/1w_Wrof4vltDEybw34DKnvOpVOMF6e-GO/view?usp=drive_link

- Sample Software Design Document

https://drive.google.com/file/d/18sOWBsQTUh05TdcdXATE3gMHos3iXjkx/view?usp=drive_link

2. System Overview

The ISO Audit Management System employs MySQL as the backend database and HTML/CSS for the frontend user interface. One of its key advantages is scalability; it can handle a growing volume of audit data and users, making it suitable for organisations of all sizes. The web-based nature of the system ensures portability, allowing auditors and administrators to access and use it from various devices, including smartphones and tablets, regardless of their physical location.

The system improves efficiency by automating audit processes, such as planning, execution, and reporting, reducing manual paperwork and administrative burdens. It promotes data accuracy through the centralised data storage in MySQL, decreasing the likelihood of errors and duplication. Real-time insights provided by the system enable organisations to monitor their compliance status and track progress towards ISO certification

3. System Components

3.1 Decomposition, Dependency and Interface Description

System components:

1. Audit planning component
    a. Description:
        a. Component for the planning, creation and organising of audits and resource, permissions management for efficient and effective audit process.
    b. Dependencies:
        a. User Management and Access Control Module for user authentication and authorization.
        b. Data and Evidence Repository for storing audit plans and historical data.
    c. Interfaces:
        a. Audit Creation:
            CreateAudit()
        b. Audit Scheduling:
            ScheduleAudit()
            UpdateSchedule()
        c. Audit Checklist:
            CreateChecklist()
            AddItem()
            RemoveItem()
        d. Resource Allocation:
            AddResource()
            CheckAvailability()

2. Audit execution component
    a. Description:
        a. Component which supports the execution of the audits.
        b. Auditors collect the data, record observations and store it in the database.
    b. Dependencies:
        a. Data and evidence repository for data storage and retrieval
        b. Audit Planning module
    c. Interfaces:
        a. Audit Data Collection:
            CreateDocument()

EditDocument()
DeleteDocument()

b. Audit Evidence and Findings Management:
   UploadDocument()
   GetDocument()
c. Progress Tracking:
   CheckProgress()
d. Communication:
   CreateRoom()
   AddMember()
   CreateChat()
   DeleteRoom()
e. Auditor Training and Competence:
   CreateCourse()
   JoinCourse()
   CreateTest()
   TakeTest()

3. Audit reporting and analysis component
   a. Description:
      a. Supports post-audit activities and generates the audit-report along with the analysis using the data collected during the audit.
   b. Dependencies:
      a. Audit Execution module
      b. Database Manager
      c. Audit Planning module
   c. Interfaces:
      a. Automated Report Generation:
         UploadData()
         GenerateReport()
      b. Report Distribution:
         SendReportTo()
      c. Data Visualisation:
         UploadReport()
         ShowReportStats()
         CreatePDF()
         ShareStats()

4. Dashboard component (include audit and compliance, risk and opportunities together)
   a. Description:
      a. Main Dashboard with all functionalities in a unified place for ease of access, user can access all details pertaining to the audits and other details and also receives communications.
   b. Dependencies:

a. User management and access control
b. Audit reporting and analysis module

c. Interfaces:
    a. Compliance Management:
        VerifyDocument()
        CheckStandards()
    b. Risks and Opportunities Management:
        IdentifyRisk()
        RiskRank()
    c. Notifications and Updates:
        PushNotification()
        ShareUpdates()
    d. General Tracking :
        CallAuditFunctions()

5. Data and evidence repository
    a. Description:
        a. Supports the organisation, storage and retrieval of documents, statements pertaining to audits, auditors and auditees.
    b. Interfaces:
        a. Data Ingestion and validation:
            UploadData()
            UploadDocument()
            RemoveDocument()
            RemoveData()
            VerifyDocument()
        b. Data Retrieval (based on authorization, permissions):
            CheckRole()
            CheckAuth()
            GetDocument()
        c. Audit Trail:
            EventLog()
        d. Data Management (organisation, categorisation):
            CreateFolder()
            CreateFile()
            UploadFile()
            GetFile()

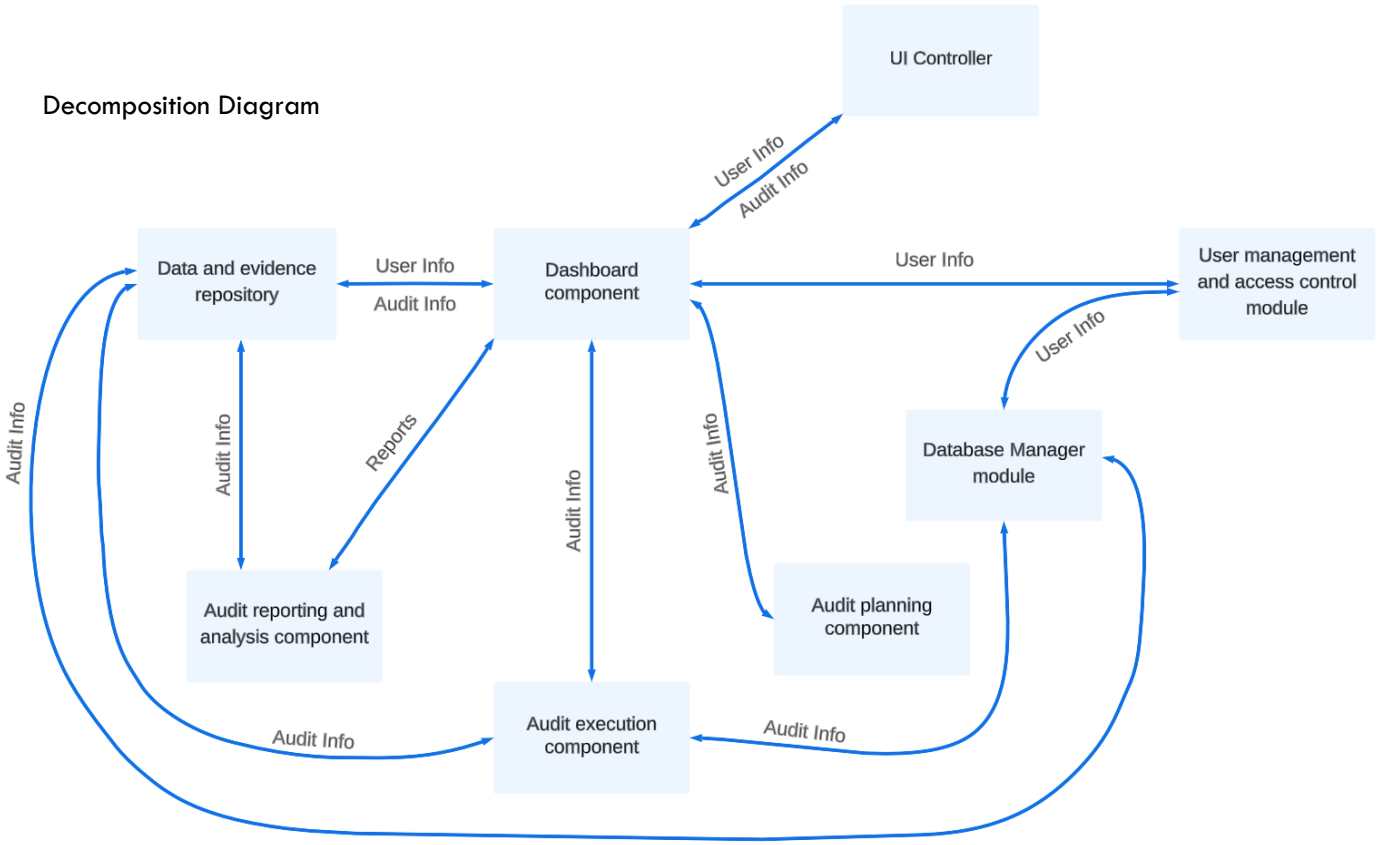
        e. Data export and sharing:
            ExportDocument()
            ShareDocument()

6. UI controller module
    . Description:
        a. Module which provides an interface for the users to interact with the software efficiently.
        b. This module is designed to provide a user-friendly and intuitive interface, enabling users to navigate, input data, and access system functionality efficiently.
    a. Interfaces:
        a. Navigation interface
        b. UI interaction

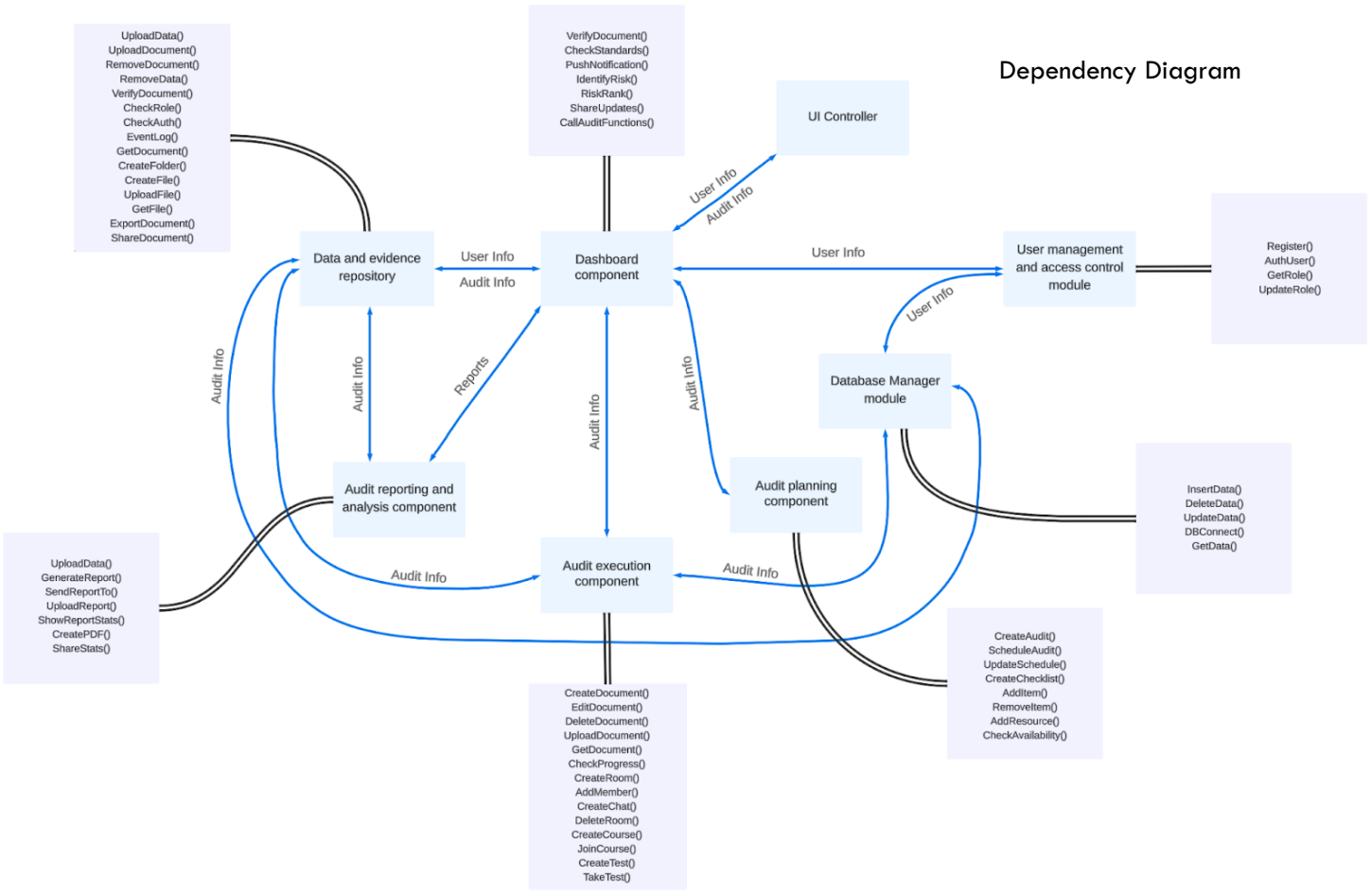
7. Database Manager module
    1. Description:
        1. The module serves the purpose of a central repository for data storage, retrieval, and management while ensuring data integrity, performance and reliability.
        2. Handles transaction management, query optimization and schema design.
    2. Interfaces:
        1. Data Insertion and Modification:
            InsertData()
            DeleteData()
            UpdateData()
        2. Database connection and transaction management:
            DBConnect()
        3. SQL query execution and optimisation:
            GetData()

8. User management and access control module
    a. Description:
        1. User registration, permissions and authentication module that ensures security and data privacy.
        2. Gives roles to individuals to ensure security and organisation.
    b. Dependencies:
        1. Database Manager
    c. Interfaces:
        1. User registration
            Register()
        2. User authentication
            AuthUser()
        3. RBCA
            GetRole()
            UpdateRole()

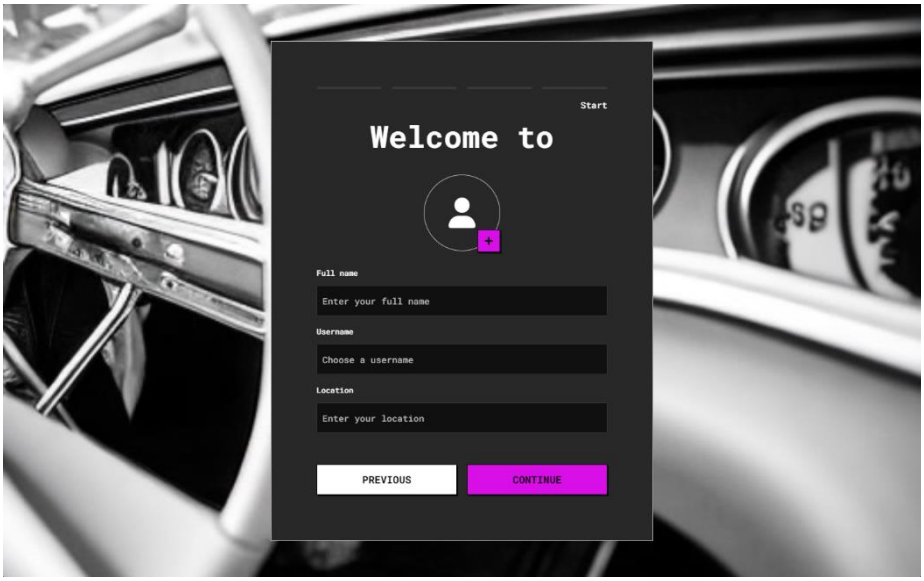
Decomposition Diagram



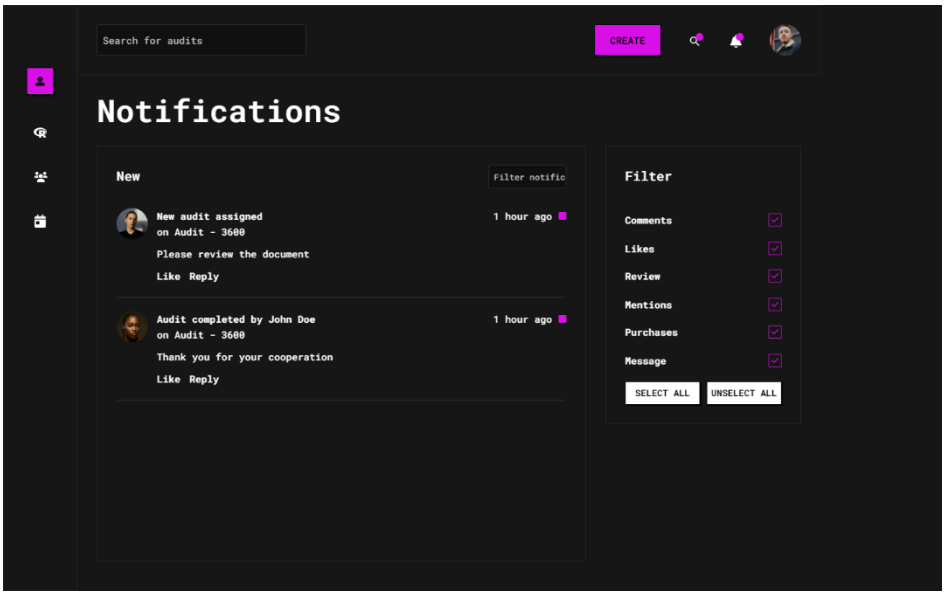
Dependency Diagram



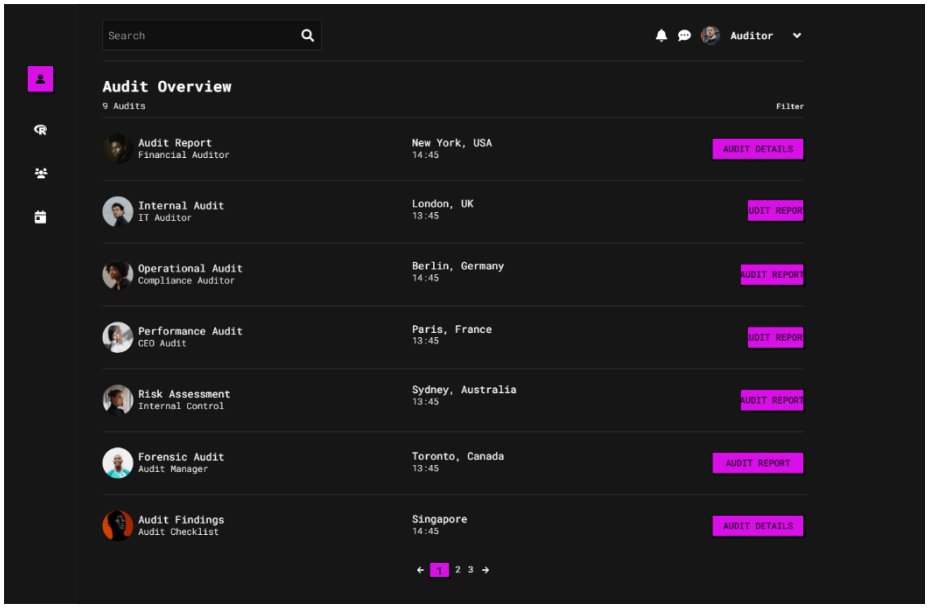
3.2 User Interfaces



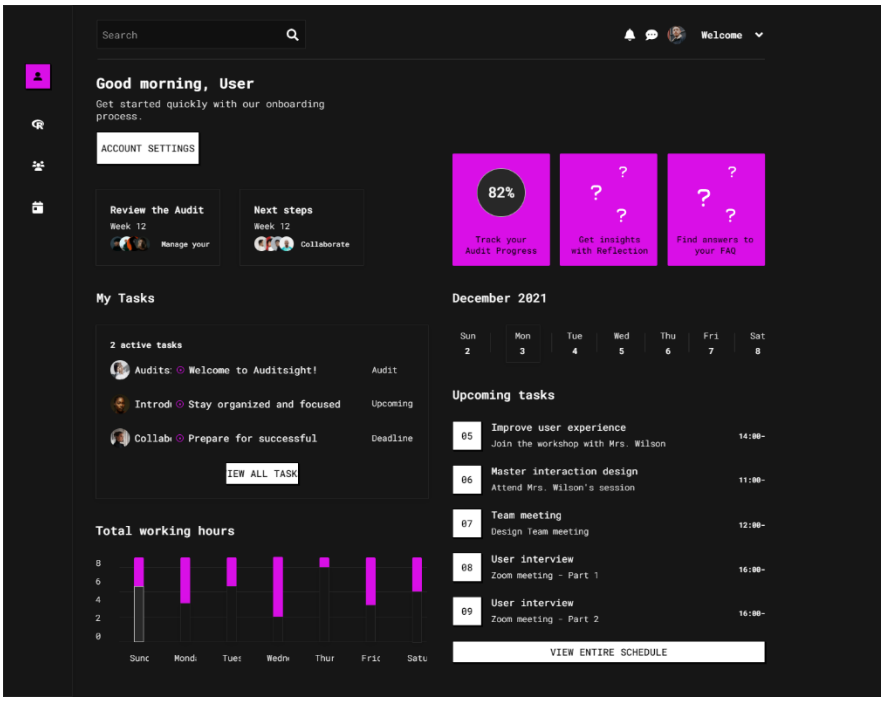
Login Page for users to be authenticated



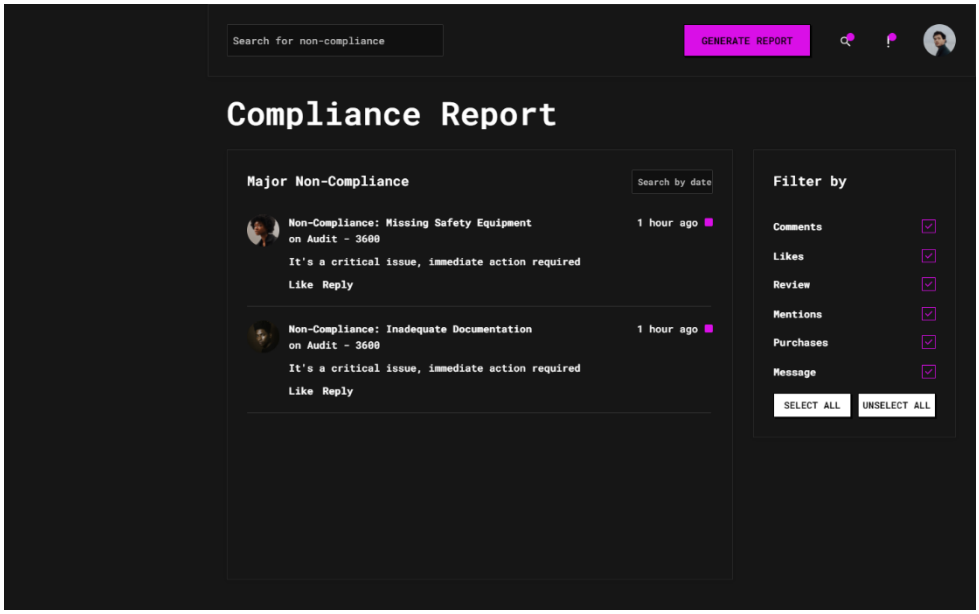
Notifications and Updates Page



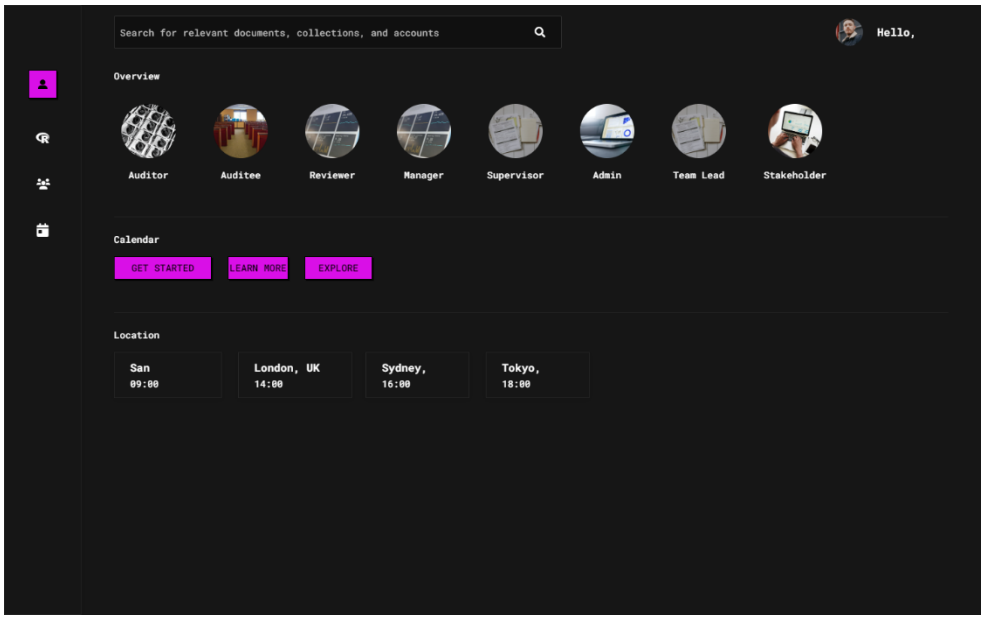
Audit Details Board that gives all details regarding auditors, locations etc.



Dashboard that gives an overview of all details pertaining to the audit



Generated Compliance report based on findings by the audit team

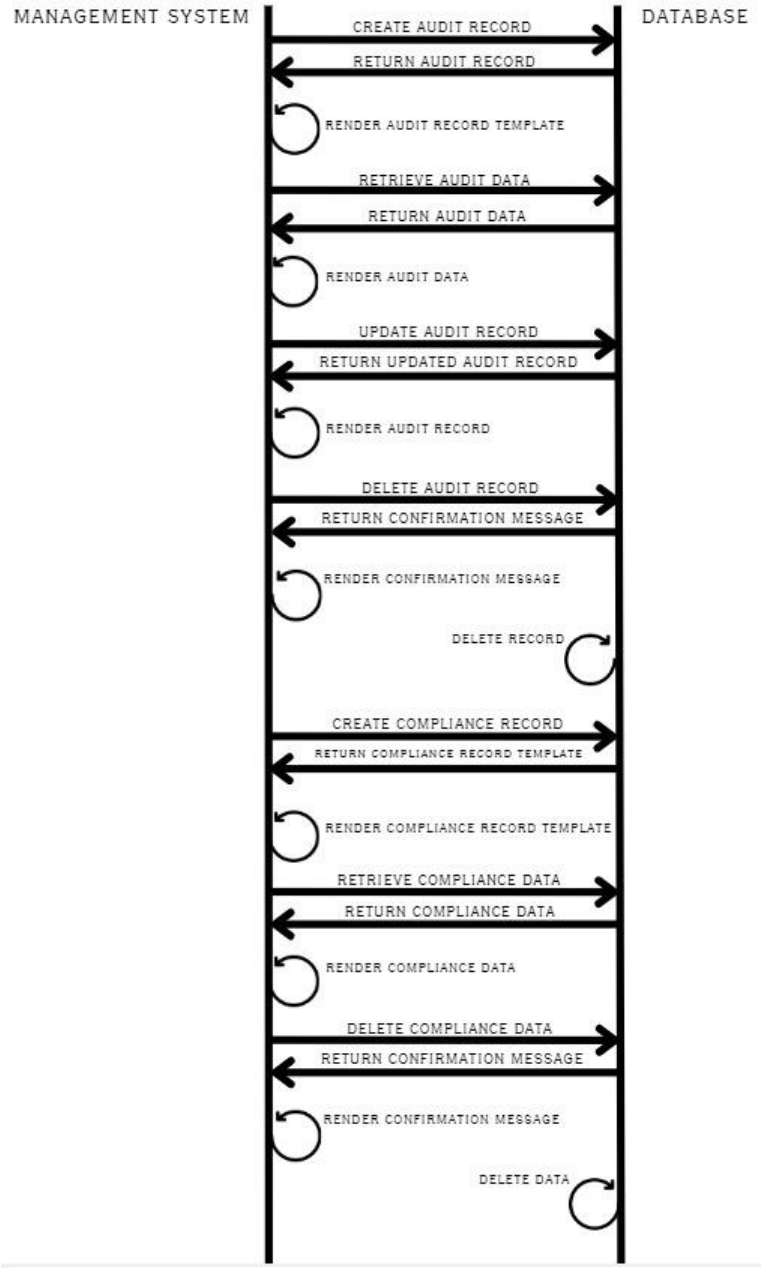


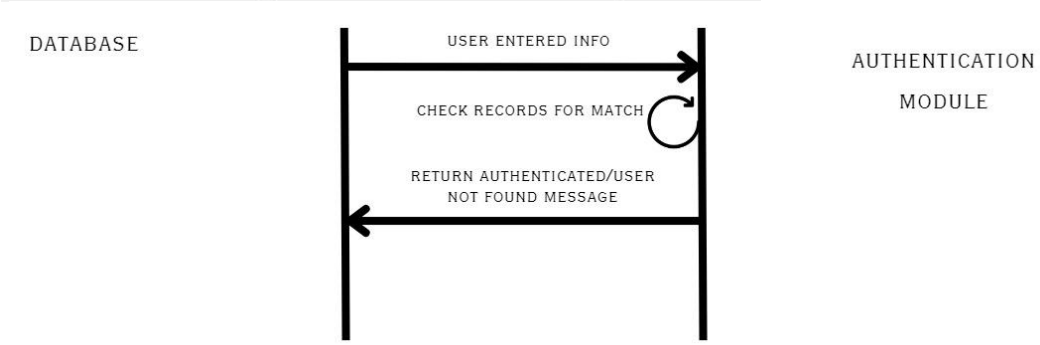
Overview Tab with other details and filters

4. Detailed Design

4.1 Module Detailed Design

4.1.1 Data Retrieval



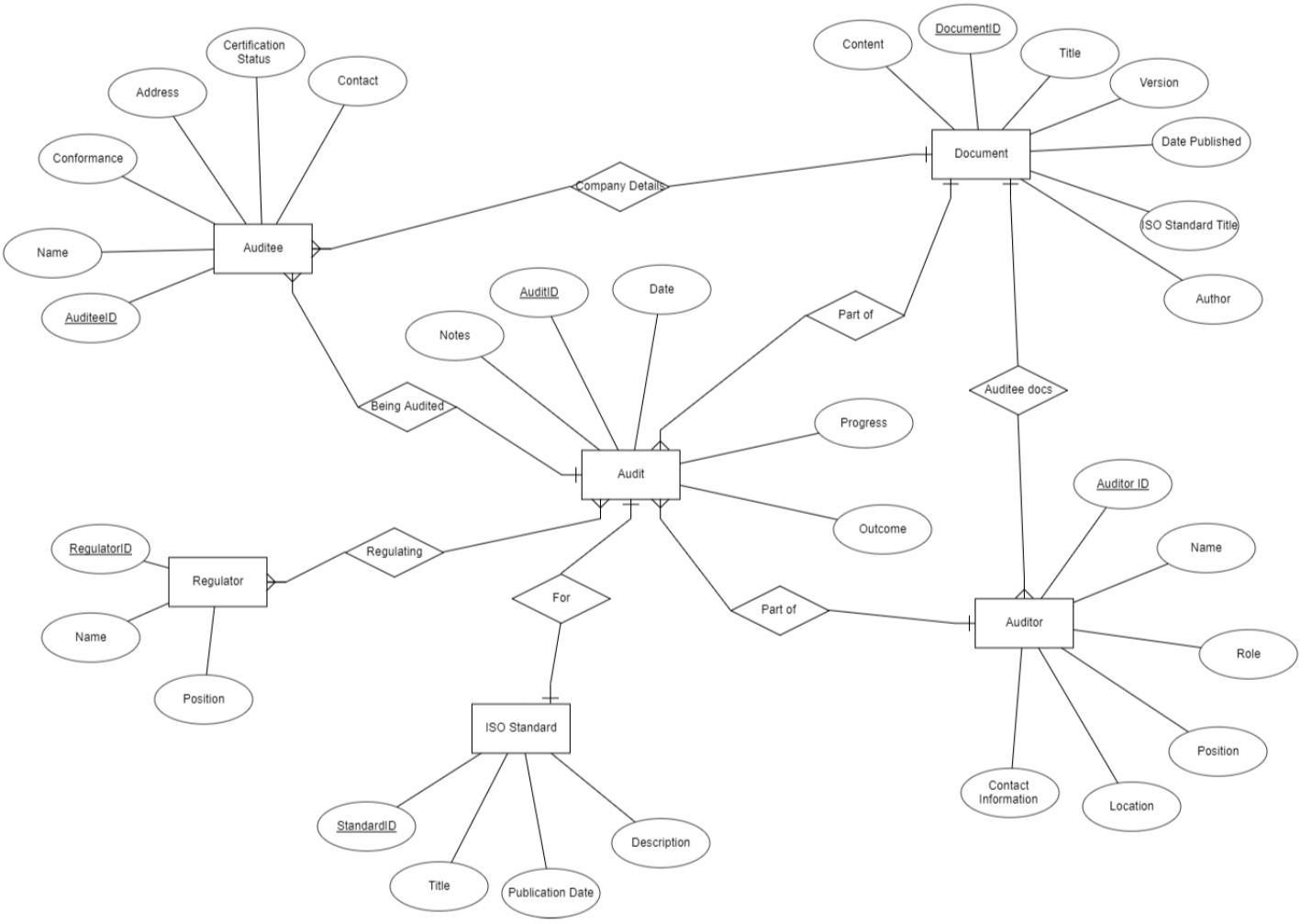


Pseudocode:

```
User initiates CRUD operation
    Database call auth module
        If(Auth module verification() == success)
            Perform CRUD operation
        Else
            Failed operation

Auth module verification
    If user credential is valid and has permission
        Return success
    Else
        Return failure
```

4.2 Data Detailed Design



4.3 RTM

Requirement ID	Requirement Description	Design Component	Test Case #
3.1.1	Login Page (login, remember-me, forgot password)	Dashboard Component	
3.1.2	Dashboard	Dashboard Component	
3.1.3	Audit Planning Interface (create and manage audit)	Audit Planning Component	
3.1.4	Audit Execution Interface (checklist, communication tools)	Audit Execution Component	
3.1.5	Audit Reporting and Analysis (generating, reviewing and analysing audit reports)	Audit Reporting and Analysis Component	
3.1.6	Document Management Interface (searching and filtering)	Data and Evidence Repository	
3.1.7	User Profile and Settings (edit profile)	User Management and Access Control Module	