

Darshan University

A Project Report on

**“Exam Management system”**

Under the subject

**Software Engineering (2101CS503)**

B. Tech, Semester – V

Computer Science & Engineering Department

|  |  |
| --- | --- |
| Submitted By | |
| Student Name:Shreyansh Patel | Enrollment No.:22010101147 |
| Academic Year  (2024-2025) | |
| Internal Guide  Prof. Rajkumar Gondaliya  Darshan University | Dean-DIET  Dr. Gopi Sanghani  Darshan University |

|  |  |
| --- | --- |
|  | **Computer Science & Engineering Department**  **Darshan University** |

**DECLARATION**

We hereby declare that the SRS, submitted along with the **Software Engineering** **(2101CS503)** for entitled **“Exam Management system”** submitted in partial fulfillment for the Semester-5 of **Bachelor Technology (B. Tech)** in **Computer Science and Engineering (CSE)** Departmentto Darshan University, Rajkot, is a record of the work carried out at **Darshan University, Rajkot** under the supervision of Prof. Rajkumar Gondaliyaand that no part of any of report has been directly copied from any students’ reports, without providing due reference.

Shreyansh Patel

Student’s Signature

Date: \_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
|  | **Computer Science & Engineering Department**  **Darshan University** |

**CERTIFICATE**

This is to certify that the SRS on **“Exam Management system” has** been satisfactorily prepared by **Shreyansh Patel (22010101147)** under my guidance in the fulfillment of the course **Software Engineering (2101CS503)** work during the academic year 2024-2025.

|  |  |  |
| --- | --- | --- |
| Internal Guide  Prof. Rajkumar Gondaliya  Darshan University |  | Dean-DIET  Dr. Gopi Sanghani  Darshan University |

**Acknowledgement**

I wish to express my sincere gratitude to my project guide **Prof. Rajkumar Gondaliya** and all the faculty members for helping me through my project by giving me the necessary suggestions and advices along with their valuable co- ordination in completing this work.

I also thank my parents, friends and all the members of the family for their precious support and encouragement which they had provided in completion of my work. In addition to that, I would also like to mention the Darshan University personals who gave me the permission to use and experience the valuable resources required for the project from the University premises.

Thus, in conclusion to the above said, I once again thank the faculties and members of **Darshan University** for their valuable support in completion of the project.

Thanking You

**Shreyansh Patel**

**ABSTRACT**

Exam management system is a system which aims in developing a computerized system to take exams. This system will act as a tool to transfer traditional Exam into digital Exam. This project has many features which are generally not available in offline exam system like facility of user login, submission of test on one click etc. It also has a facility of admin login through which the admin can monitor the whole system.  This system will be designed with the basic features such as Examiner can add/view/update/delete tests and question details in it. It has also a facility where student after logging in their accounts can see list of all tests and its result and also they can see upcoming tests. The examiner after logging into his account i.e. admin account can generate various reports such as student report, subject report, teacher report.

Main purpose of this system is to reduce human efforts as much as possible.

**Table of Contents**

[List of Figures I](#_Toc108)

[List of Tables II](#_Toc14109)

[1 Introduction 1](#_Toc25242)

[1.1 Product perspective 1](#_Toc21347)

[1.2 Product features 1](#_Toc16759)

[1.2.1 There are three different users who will be using this product: 1](#_Toc10083)

[1.2.2 The features that are required for the Admin are: 1](#_Toc20697)

[1.2.3 The features that are required for the Student are: 1](#_Toc16312)

[1.2.4 The features that are required for the Faculty Members are: 1](#_Toc31369)

[1.3 Functional Requirement 2](#_Toc6202)

[1.3.1 Admin : 2](#_Toc26474)

[1.3.2 Student : 2](#_Toc2309)

[1.3.3 Faculty Members : 3](#_Toc7903)

[1.4 Non-Functional Requirement 4](#_Toc13504)

[1.4.1 Usability: 4](#_Toc22145)

[1.4.2 Accuracy: 4](#_Toc18947)

[1.4.3 Availability: 4](#_Toc7263)

[1.4.4 Maintainability: 4](#_Toc14609)

[2 Design and Implementation Constraints 5](#_Toc6237)

[2.1 Use case diagram 5](#_Toc18496)

[2.2 Activity diagram and Swimlane diagram 6](#_Toc17776)

[2.3 Sequence diagram 8](#_Toc1222)

[2.4 State diagram 9](#_Toc18883)

[2.5 Class diagram 10](#_Toc15042)

[2.5.1 Context diagram (level-0) 11](#_Toc3343)

[2.5.2 DFD Level-1 11](#_Toc14011)

[3 External interface requirement (Screens) 12](#_Toc26284)

[3.1 Screen-1: Registration Form 12](#_Toc25581)

[3.2 Screen-3: Invigilator page 14](#_Toc18720)

[3.3 Screen-2: Exam Registration Form 15](#_Toc13846)

[3.4 Screen-3: Upload Question paper 16](#_Toc16187)

[3.5 Screen-3: Add Invigilator details 17](#_Toc679)

[4 Database design 18](#_Toc28417)

[4.1 List of Tables 18](#_Toc19588)

[5 Stories and Scenario 20](#_Toc14577)

[5.1.1 Scenario# S1.1 20](#_Toc24249)

[5.1.2 Scenario# S1.2 20](#_Toc32357)

[5.1.3 Scenario# S1.3 21](#_Toc13766)

[5.1.4 Scenario# S1.4 21](#_Toc12324)

[5.1.5 Scenario# S1.5 22](#_Toc29816)

[6 Test cases 23](#_Toc3680)

[7 References 29](#_Toc18541)

# List of Figures

[Figure 2.1‑1 Use case diagram for Exam management system 5](#_Toc9352)

[Figure 2.3‑1 Sequence diagram for taking attendance 8](#_Toc11588)

[Figure 2.4‑1 State diagram of taking attendance 9](#_Toc15089)

[Figure 2.5‑1 Class diagram for Exam management system 10](#_Toc5102)

[Figure 2.6‑1 Context diagram for Exam management system 11](#_Toc23716)

[Figure 2.6‑2 DFD level-1 for Exam management system 11](#_Toc32740)

[Figure 3.1‑1 Screen-1: Student screen 12](#_Toc6991)

[Figure 3.2‑1 Screen-2: Faculty screen 13](#_Toc6116)

[Figure 3.3‑1 Screen-3: Invigilator page 14](#_Toc6189)

# List of Tables

[Table 3.1‑1 Screen element of Registration form 12](#_Toc934)

[Table 3.2‑1 Screen element of Login form 13](#_Toc7137)

[Table 3.3‑1 Screen element of Add borrower book 14](#_Toc1427)

[Table 4.1‑1 Table: Exam 18](#_Toc20392)

[Table 4.1‑2 Table: Question Paper 18](#_Toc20103)

[Table 4.1‑3 Table: Student 18](#_Toc22501)

[Table 4.1‑4 Table: Question 19](#_Toc24164)

[Table 4.1‑4 Table: Faculty 19](#_Toc29471)

# Introduction

## Product perspective

This project is basically updating the manual library system into an internet-based application so that the users can know the details of their account, availability of books etc. It is a multi-user version and can take care of all the fundamental functions of a Library like Cataloguing, Circulation, Accessioning and Housekeeping. It can satisfactorily cater to all the basic functions of a small library.

## Product features

### There are three different users who will be using this product:

* Admin who will be acting as the administrator.
* Student who will be accessing the library.
* Faculty Members who will request for membership.

### The features that are required for the Admin are:

· **User Management:** Admins are responsible for adding, removing, and updating user details for

students, instructors, and proctors.

· **Exam Creation and Management:** Admins can create, update, and delete exam templates and

schedules. They also publish exam schedules and manage exam-related rules and regulations.

· **System Configuration:** Admins configure system settings, including access controls, system

preferences, and security protocols.

· **Monitoring and Reporting:** They monitor ongoing exams, generate detailed reports on exam

performance, and ensure the integrity and security of the exam process.

· **Data Management:** Admins are responsible for backing up and restoring system data, as well as

archiving past exam data.

· **Notifications and Support:** They send notifications to users, handle student complaints, and provide technical support.

### The features that are required for the Student are:

· **Exam Registration:** Students register for upcoming exams and download necessary materials such as admit cards and exam instructions.

· **Exam Participation:** They take online exams as per the schedule and follow the provided guidelines and rules.

· **Results and Feedback:** Students view their exam results, request re-evaluation if needed, and provide feedback on exams.

· **Study and Preparation:** They access study materials, join study groups, and use practice exams to prepare for upcoming tests.

· **Profile Management:** Students update their personal profile information and view their grades history.

### The features that are required for the Faculty Members are:

· **Question Management:** Faculty Members create, update, and delete questions for the exam question bank.

· **Exam Creation:** They create and update exams, including setting the difficulty level and structure.

· **Grading and Feedback:** Faculty Members grade exams, provide feedback to students, and review exam results.

· **Student Interaction:** They communicate with students regarding exam preparation, provide study materials, and organize study sessions.

· **Performance Analysis:** Faculty Members analyze exam performance, track student progress, and review student feedback to improve the exam process.

## Functional Requirement

### Admin :

· **User Management**

* Ability to add, edit, and delete users (students and instructors).
* Granting permissions based on roles (admin, student, instructor).

· **Exam Creation**

* Creating exams with various question types (multiple-choice, essay, etc.).
* Setting time limits and exam availability.

· **Monitoring and Reporting**

* Viewing exam results and analytics.
* Generating reports on student performance and course statistics.

· **System Configuration**

* Configuring system settings such as exam timings, grading schemes, etc.
* Customizing email notifications and alerts.

· **Security Management**

* Ensuring data security and integrity.
* Managing access control and permissions.

· **Payment Management**

* Handling fee collection for exams or courses.
* Integration with payment gateways for transactions.

### Student :

· **Exam Enrollment**

* Viewing available exams and enrolling in them.
* Checking exam schedules and timings.

· **Exam Attempt**

* Taking exams within the specified time limits.
* Submitting answers and reviewing responses where applicable.

· **Performance Tracking**

* Viewing personal exam results and grades.

· **Profile Management**

* Updating personal information.
* Viewing personal information.

· **Notification Alerts**

* Receiving alerts for upcoming exams and deadlines.
* Notifying about changes in exam schedules.

· **Feedback Submission**

* Providing feedback on exams.

· **Support Access**

* Contacting support for technical assistance.

· **Mobile Access**

* Accessing the platform via mobile devices.
* Taking exams using mobile-compatible interfaces.

### Faculty Members :

· **Exam Creation and Management**

* Creating exams.

· **Grading and Feedback**

* Grading student submissions for exams.
* Providing detailed feedback and annotations.

· **Communication**

* Sending announcements and notifications to students.

· **Attendance Tracking**

* Recording student attendance for classes or exams.
* Monitoring participation and engagement.

· **Collaboration Tools**

* Integrating with tools for virtual classrooms (video conferencing, etc.).
* Facilitating group projects and discussions.

· **Assessment Security**

* Ensuring exam integrity and preventing cheating.
* Monitoring suspicious activity during exams.

· **Course Evaluation**

* Gathering student feedback on courses.
* Implementing improvements based on feedback.

· **Certification Management**

* Issuing certificates for course completion.
* Verifying student credentials for certification.

### 

## Non-Functional Requirement

### Usability:

* The UI should be simple enough for everyone to understand and get the relevant information without any special training. Different languages can be provided based on the requirements.

### Accuracy:

* The data stored about the books and the fines calculated should be correct, consistent, and reliable.

### Availability:

* The System should be available for the duration when the library operates and must be recovered within an hour or less if it fails. The system should respond to the requests within two seconds or less.

### Maintainability:

* The software should be easily maintainable and adding new features and making changes to the software must be as simple as possible. In addition to this, the software must also be portable.

# Design and Implementation Constraints

## Use case diagram

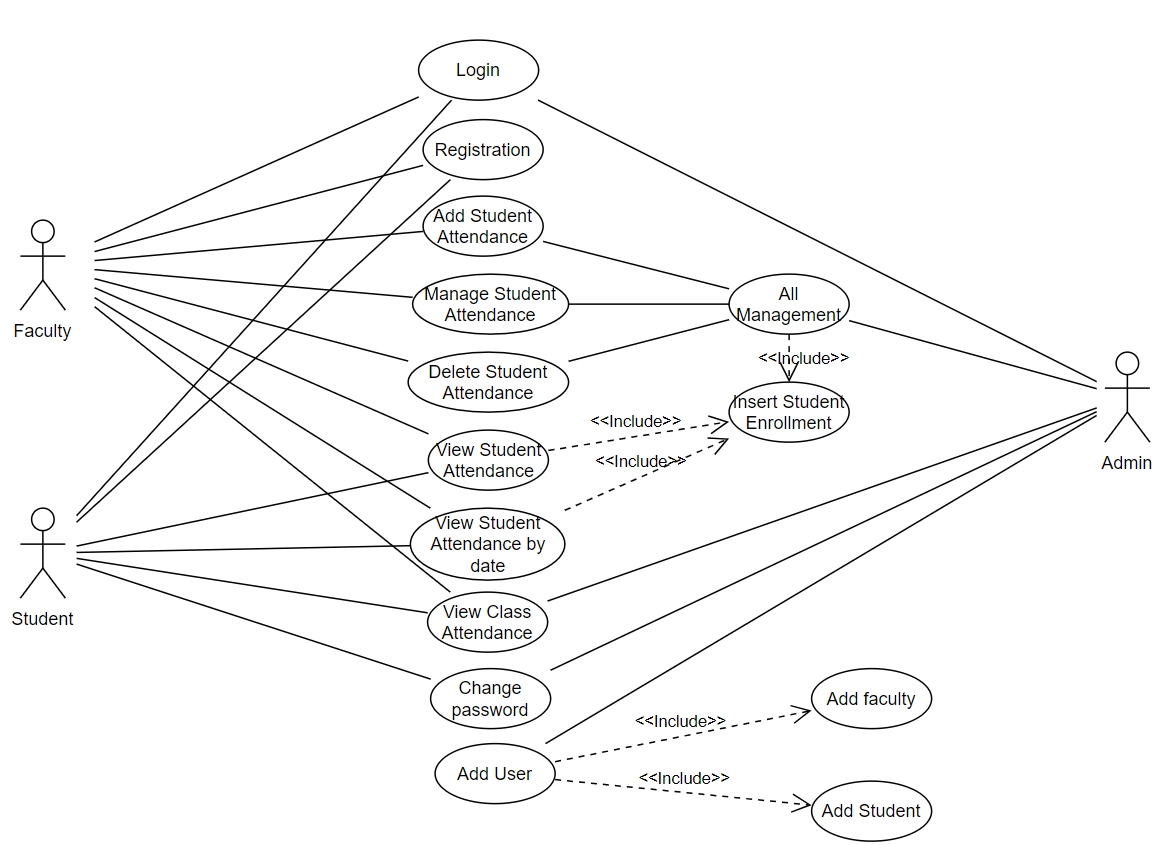
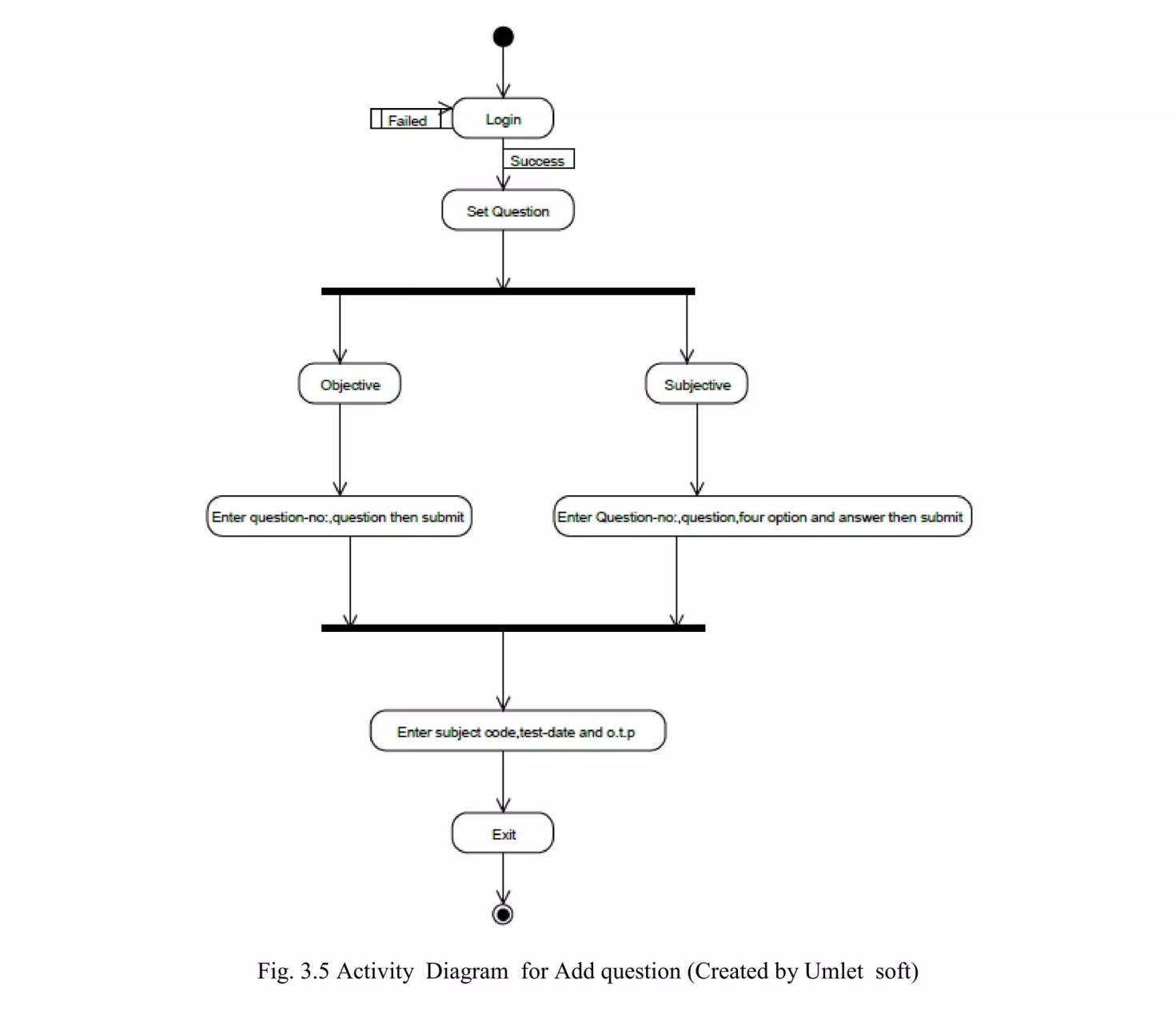
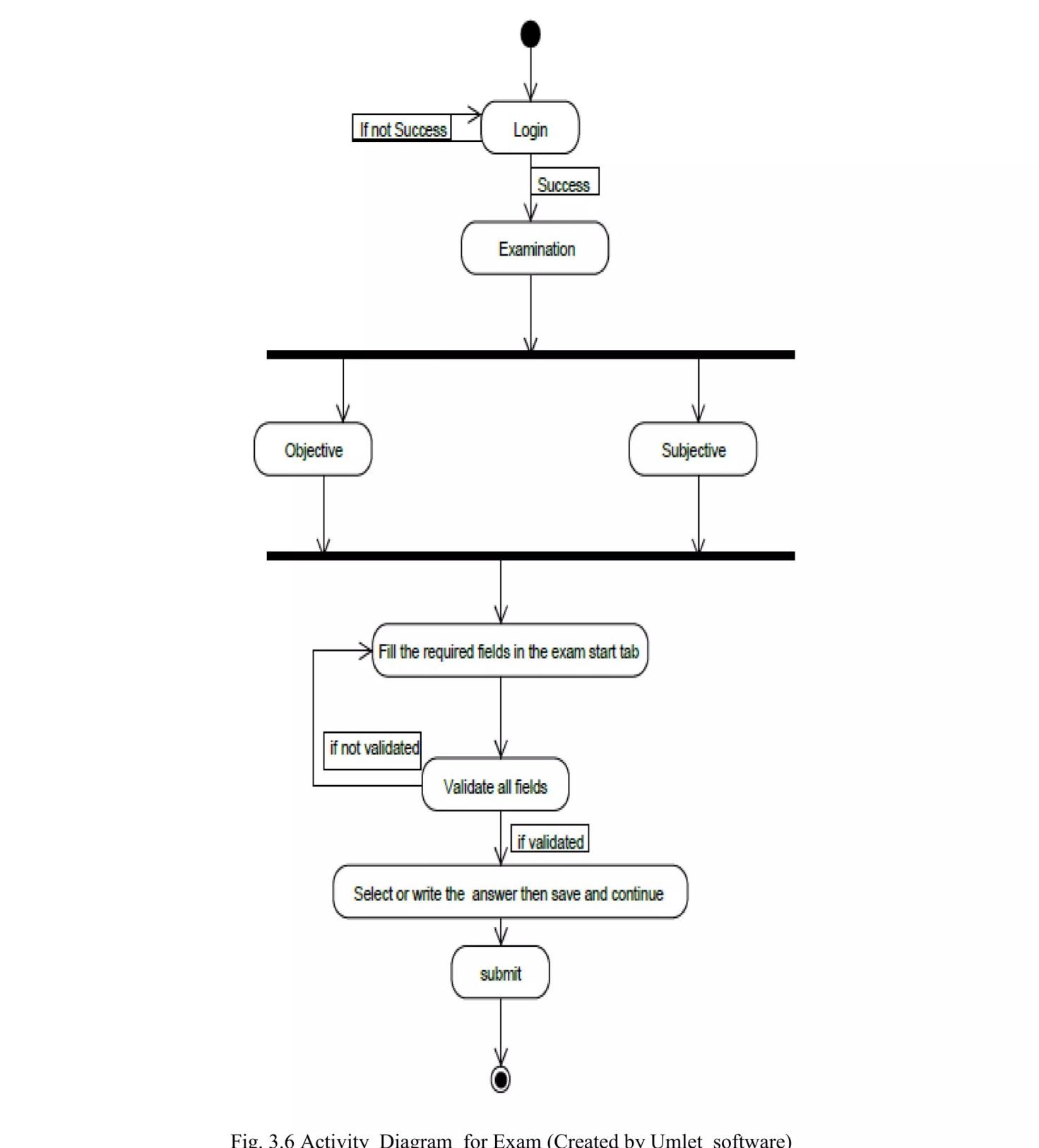


Figure 2.1‑1 Use case diagram for Exam management system

x

## Activity diagram and Swimlane diagram





## Sequence diagram

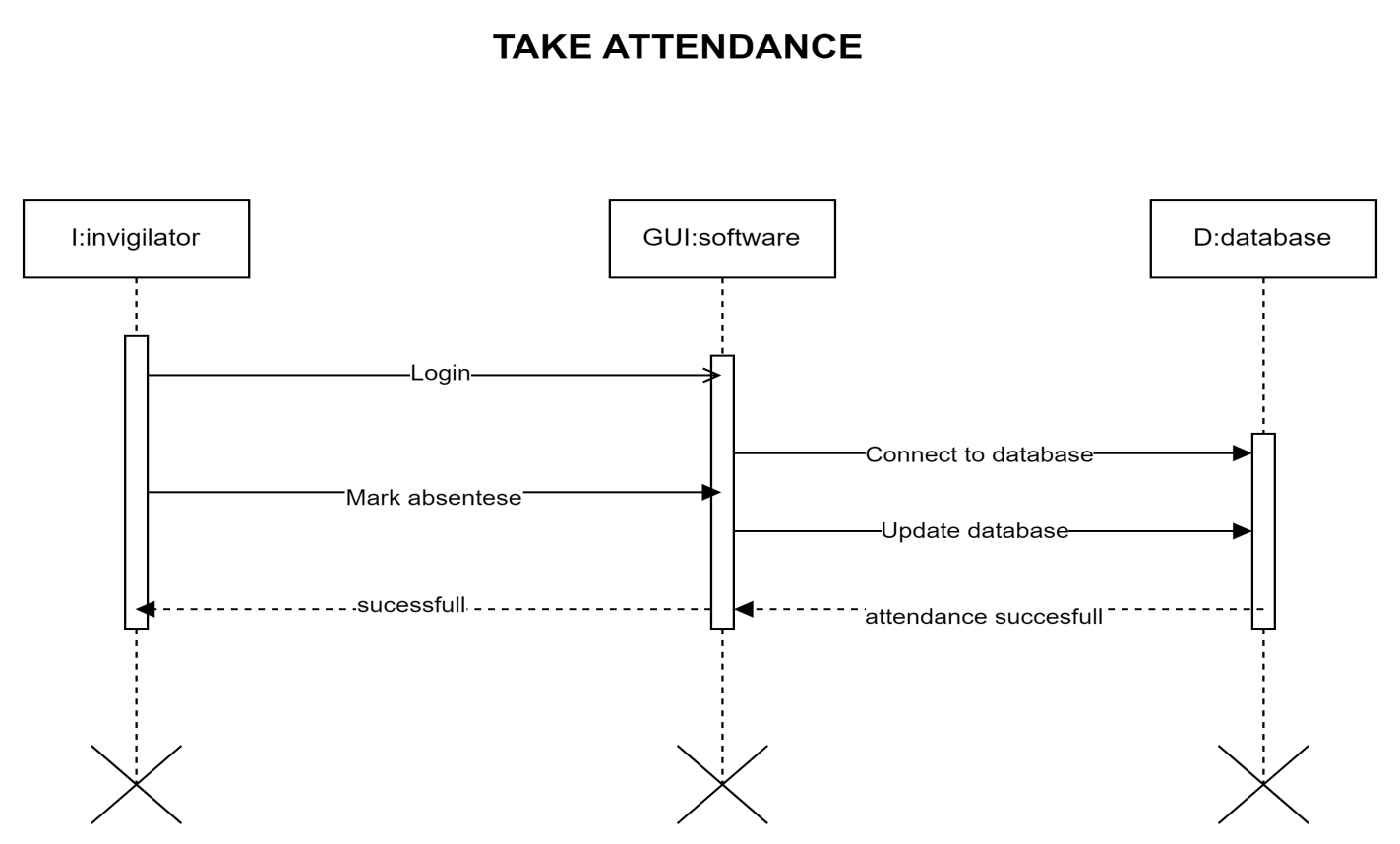


Figure 2.3‑1 Sequence diagram for taking attendance

## State diagram

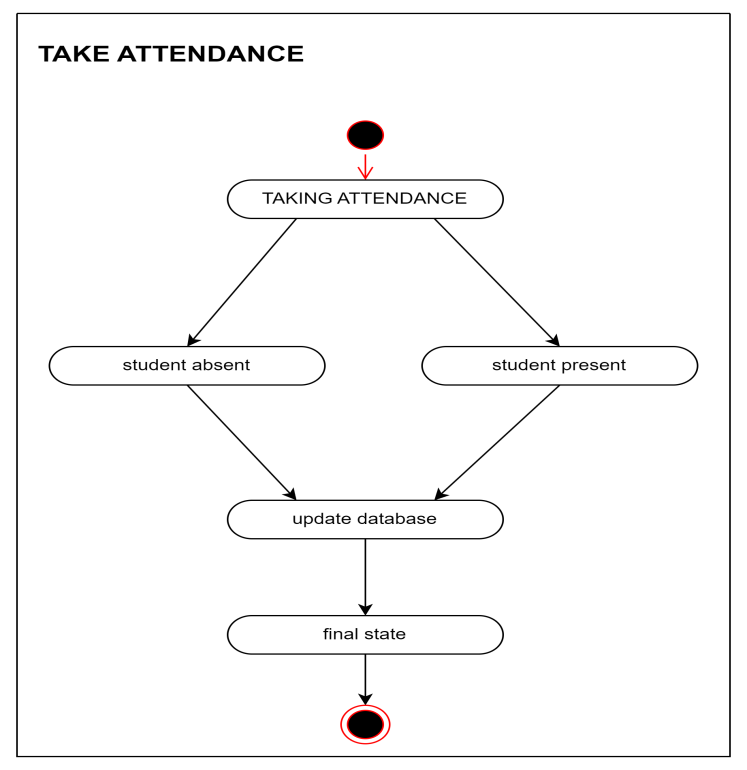


Figure 2.4‑1 State diagram of taking attendance

## Class diagram

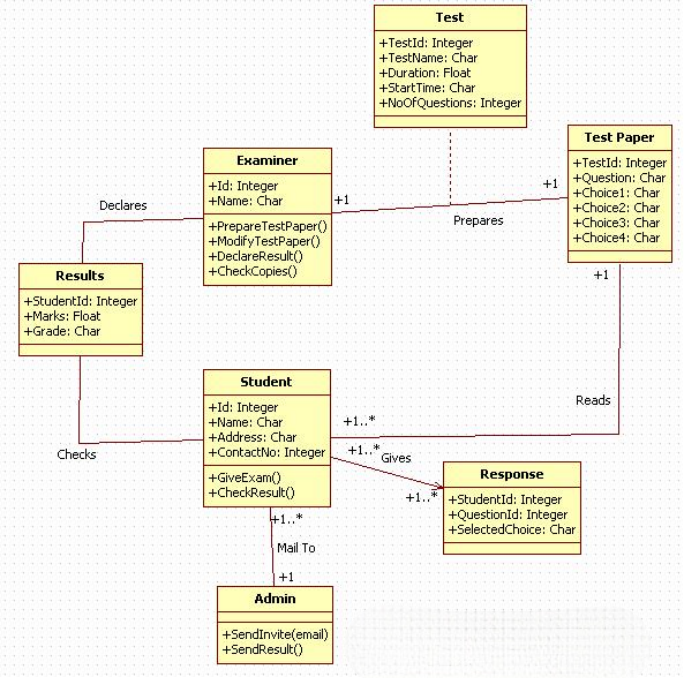


Figure 2.5‑1 Class diagram for Exam management system

### Context diagram (level-0)

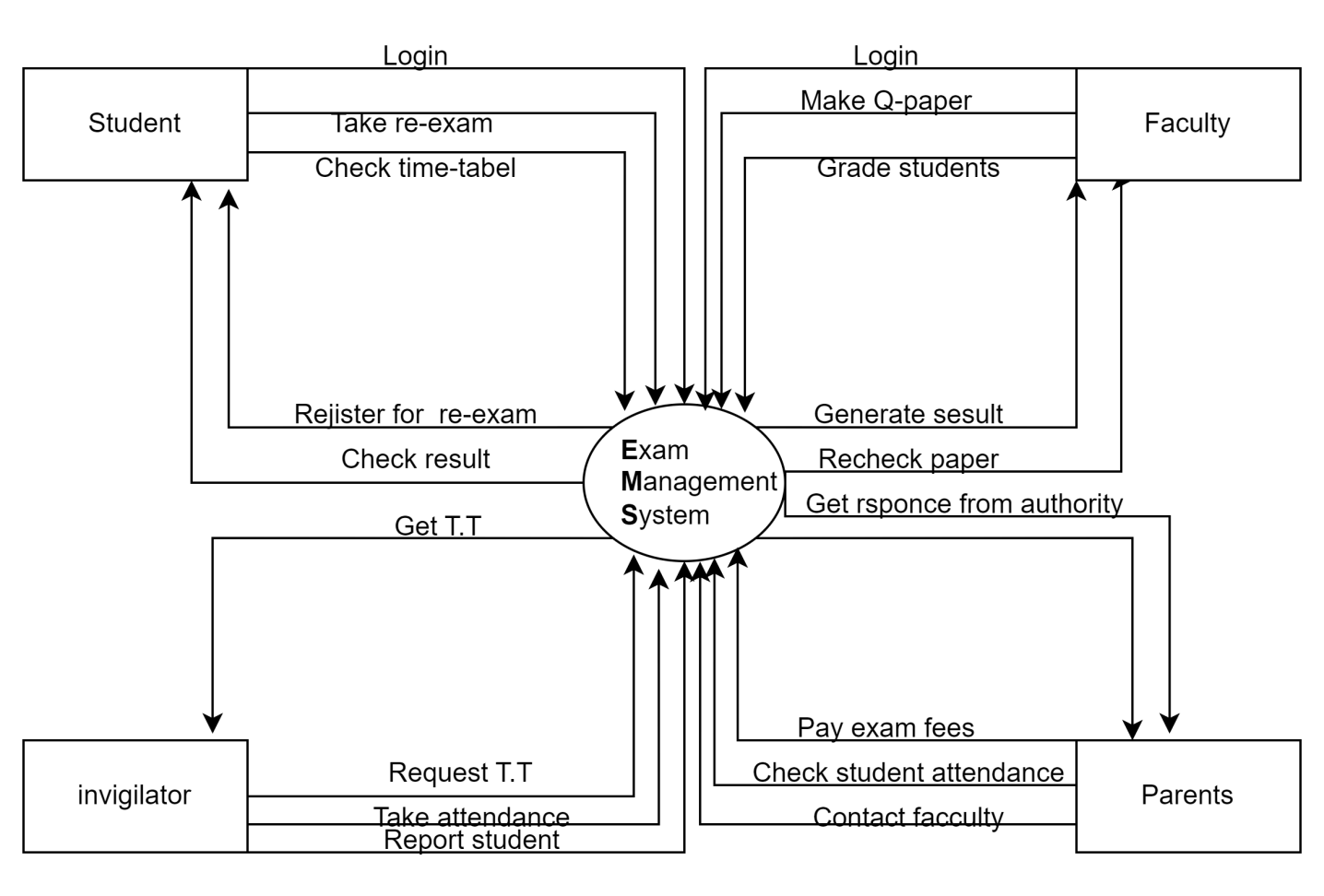


Figure 2.6‑1 Context diagram for Exam management system

### DFD Level-1

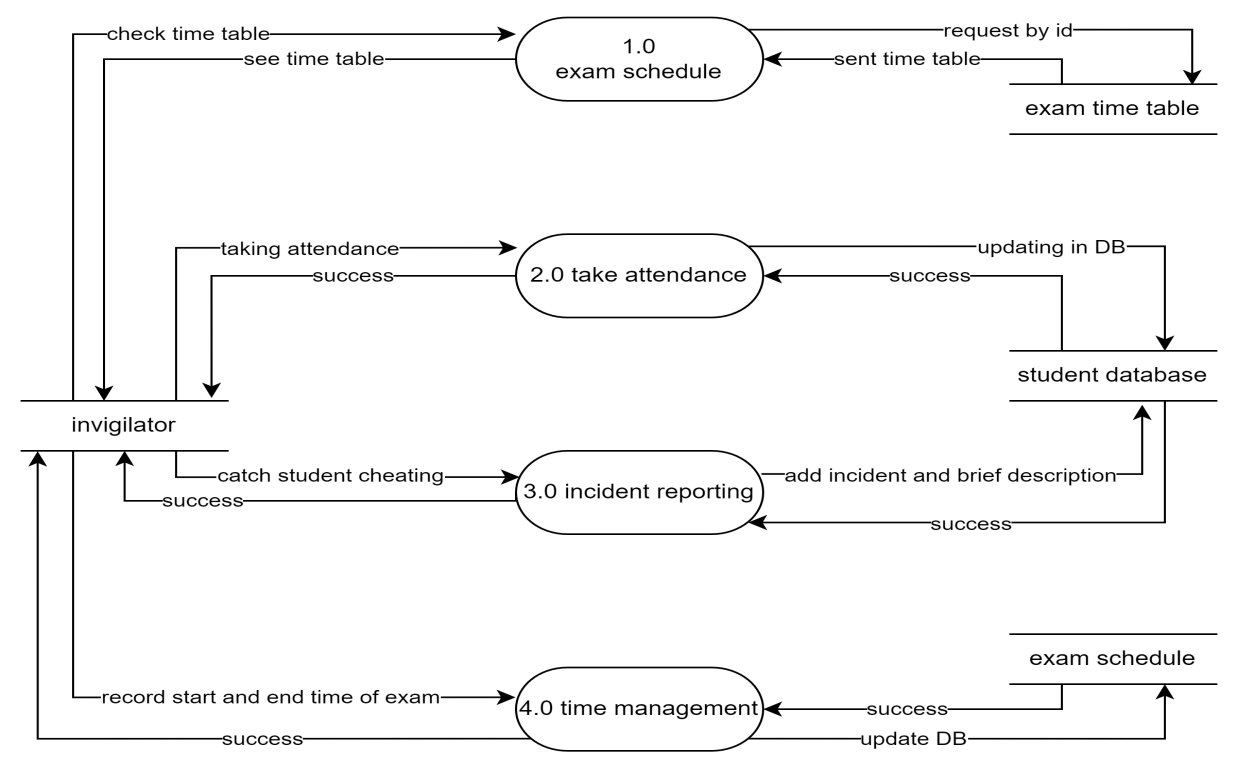


Figure 2.6‑2 DFD level-1 for Exam management system

# External interface requirement (Screens)

## Screen-1: Registration Form

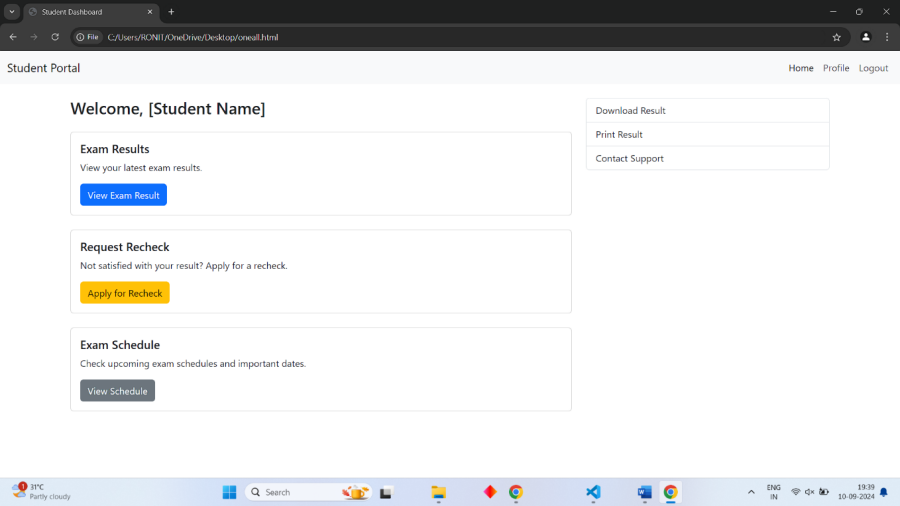


Figure 3.1‑1 Screen-1: Student screen

**Purpose:** This home page will enable students to access key functionalities within the system. They can register for exams, view schedules, review exam rules, access study materials, check results, and apply for re-assessment if needed.

Table 3.1‑1 Screen element of Registration form

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.** | **Screen Element** | **Input Type** | **O/M** | **1/N** | **Description** |
| **1** | Profile | Link | M | 1 | Takes the student to a student profile page. |
| **2** | Re-assessment | Button/card | M | 1 | If student fails they can take re-exam |
| **3** | Exam material | Button/card | M | 1 | Redirects student to a page ,providing them material |
| **4** | Exam Schedule | Button/card | M | 1 | Redirects to a sub page where the student can view the schedule of their exam |
| **5** | Re-check | Button/card | M | 1 | Unsatisfied students can apply for a recheck |
| **6** | Result | Button/card | M | 1 | Students can view the result once it is declared |

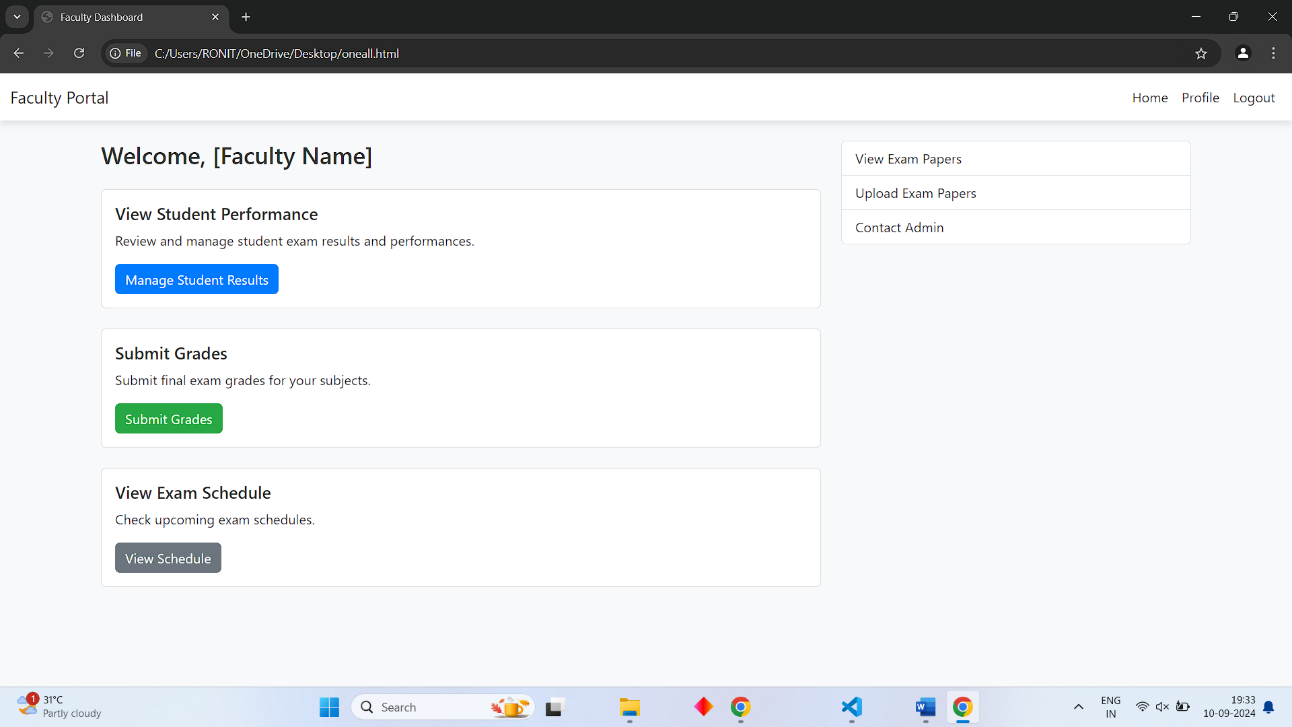
Screen-2: Faculty page

Figure 3.2‑1 Screen-2: Faculty screen

**Purpose:** This home page will allow faculty to manage exam-related tasks, including creating papers, grading, generating results, scheduling re-assessments, and handling recheck requests.

Table 3.2‑1 Screen element of Login form

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.** | **Screen Element** | **Input Type** | **O/M** | **1/N** | **Description** |
| **1** | Make paper | Card/button | M | 1 | Can make a paper of their subject |
| **2** | Declare result | Card/button | M | 1 | Grade the students and show the result |
| **3** | Re-check | Card/button | M | 1 | Accept student request for re-checking paper |
| **4** | Re-assessment | Card/button | M | 1 | Take a re-assessment exam of failed students |

## Screen-3: Invigilator page

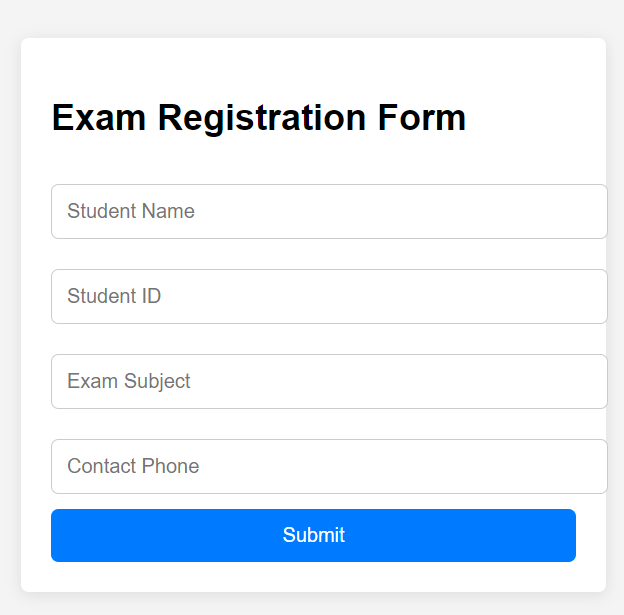
Figure 3.3‑1 Screen-3: Invigilator page

**Purpose:** This home page will enable invigilators to manage exam assignments, take attendance, report incidents, and track time for paper distribution and collection.

Table 3.3‑1 Screen element of Add borrower book

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.** | **Screen Element** | **Input Type** | **O/M** | **1/N** | **Description** |
| **1** | Exam schedule | Card/button | M | 1 | Invigilator can view the exam schedule and where they are being placed |
| **2** | Attendance | Card/button | M | 1 | Taking attendance |
| **3** | Incident reporting | Card/button | M | 1 | Report copycase |
| **4** | Time manage | Card/button | M | 1 | Manage and record the time of exam activities |

## Screen-2: Exam Registration Form



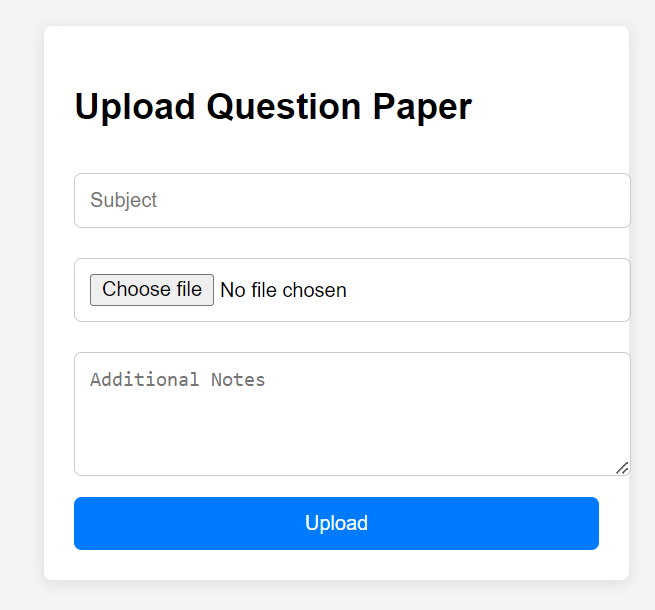
*Figure 3.2-1 Screen-2: exam registration form*

**Purpose:** This form will be used by the system’s users to access records and features of the system. The users will input the correct combination of their username and password to be able to login to the system.

*Table 3.2-1 Screen element of exam registration form*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.** | **Screen Element** | **Input Type** | **O/M** | **1/N** | **Description** |
| **1** | Student name | Textbox | M | 1 | Student name field should be editable and accept the student name. |
| **2** | studentID | textbox | M | 1 | Student want to enter it ID |
| **3** | Exam subject | textbox | M | 1 | Stubent want to choose it exam |
| **4** | Contact number | textbox | m | 1 | Student want enter its mobile number |
| **5** | submit | Button | ------ | ------ | Student can submit it exam detail |

## Screen-3: Upload Question paper



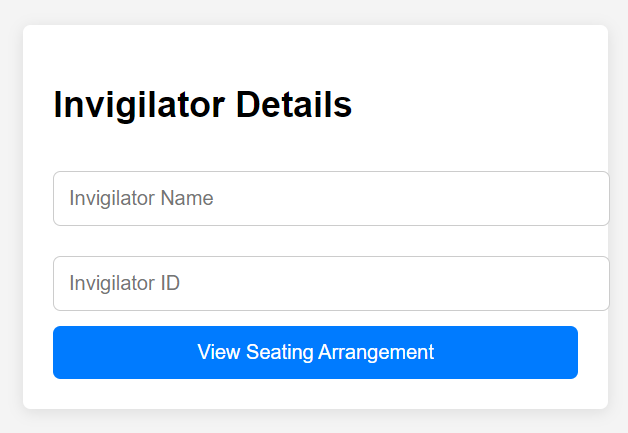
*Figure 3.3-1 Screen-3: upload question paper*

**Purpose:** This module will allow the system administrator to add, edit, update or delete borrowers of book. The admin can add borrower information and manage it.

*Table 3.3-1 Screen element of upload question paper*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.** | **Screen Element** | **Input Type** | **O/M** | **1/N** | **Description** |
| **1** | Subject name | Textbox | M | 1 | Teacher want to enter it subject name. |
| **2** | Choose the question file | Select file | M | 1 | Teacher can upload the question paper file. |
| **3** | Give some nots | Textbox | O | 1 | Teacher can give some notice for exam paper. |
| **4** | submit | button | ---- | ---- | It is submit button from this paper will upload. |

## Screen-3: Add Invigilator details



*Figure 3.3-1 Screen-3: invigilator form*

**Purpose:** This module will allow the system administrator to add, edit, update or delete borrowers of book. The admin can add borrower information and manage it.

*Table 3.3-1 Screen element of invigilator form*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.** | **Screen Element** | **Input Type** | **O/M** | **1/N** | **Description** |
| **1** | Inviglater name | Textbox | M | 1 | Invaglater want to enter it name for it detail. |
| **2** | InvagleaterID | Textbox | M | 1 | Invaglater want to enter it id for seating arrangement |
| **3** | View seating  Arrangement | button | ---- | ----- | Invaglater can seen it seating aeragment |

# Database design

## List of Tables

* Exam
* Question Paper
* Student
* Question
* Faculty
* mark sheet

Table 4.1‑1 Table: Exam

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Data Type** | **Null** | **Keys & Constrains** | **Default Value & Description** |
| **ExamID** | int | NN | PK (Auto Increment) |  |
| **ExamStartDate** | datetime | AN |  |  |
| **ExamEndDate** | datetime | AN |  |  |
| **ExamCode** | varchar(100) | NN |  |  |
| **ExamTime** | datetime | AN |  |  |
| **ExamCourse** | varchar(100) | NN |  |  |

Table 4.1‑2 Table: Question Paper

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Data Type** | **Null** | **Keys & Constrains** | **Default Value & Description** |
| **QuestionPaperID** | int | NN | PK (Auto Increment) |  |
| **SubjectID** | int | NN | FK | Reference of Subject Table |
| **QuestionPaperDate** | DateTime | AN |  |  |
| **QuestionPaperMarks** | int | NN |  |  |
| **ExamID** | int | NN | FK | Reference of Exam Table |

Table 4.1‑3 Table: Student

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Data Type** | **Null** | **Keys & Constrains** | **Default Value & Description** |
| **StudentID** | int | NN | PK (Auto Increment) |  |
| **StudentName** | varchar(100) | AN |  |  |
| **BirthDate** | Boolean | AN |  |  |
| **EnrollmentNumber** | varchar(100) | AN |  |  |
| **Gender** | varchar(10) | NN |  |  |
| **Email** | varchar(100) | AN |  |  |
| **PhoneNumber** | number | AN |  |  |
| **Date Of Birth** | datetime | AN |  |  |

Table 4.1‑4 Table: Question

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Data Type** | **Null** | **Keys & Constrains** | **Default Value & Description** |
| **QuestionID** | int | NN | PK (Auto Increment) |  |
| **Question** | varchar(10000) | NN |  |  |
| **QuestionPaperID** | int | NN | FK | Reference of Question Paper Table |
| **QuestionMarks** | int | NN |  |  |
| **QuestionNumber** | int | NN |  |  |

Table 4.1‑4 Table: Faculty

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Data Type** | **Null** | **Keys & Constrains** | **Default Value & Description** |
| **FacultyID** | int | NN | PK (Auto Increment) |  |
| **FacultyName** | varchar(100) | NN |  |  |
| **Date Of Birth** | int | NN | FK | Reference of Question Paper Table |
| **Phone Number** | int | NN |  |  |
| **Email** | int | NN |  |  |
| **AddressID** | int | NN | FK | Reference of Address Table |

# Stories and Scenario

**Story-1: Add a New Exam**

|  |  |  |
| --- | --- | --- |
| *Story #* ***S1*** | **:** | As an: Exam Administrator  I want to: Create a new exam in the system  So that: Students can be scheduled for the exam and receive their results. |
| **Priority** | **:** | High |
| **Estimate** | **:** | XL |
| **Reason** | **:** | : Creating a new exam is essential for organizing and scheduling exams within the institution, ensuring that all students are evaluated appropriately. |

### Scenario# S1.1

|  |  |  |
| --- | --- | --- |
| *Scenario#* ***S1.1*** | **:** | Add a New Exam with Valid Information |
| **Prerequisite** | **:** | The Exam Administrator is logged in to the Exam Management System. |
| **Acceptance**  **Criteria** | **:** | **Given:** The Exam Administrator is on the exam creation page, and valid exam details (e.g., exam name, date, time, duration, and subject) are provided.  **When:**  The Exam Administrator selects the "Create New Exam" option.  The Exam Administrator enters valid exam details.  The Exam Administrator clicks the "Save" button to schedule the exam.  **Then:**  The system successfully creates the exam.  The Exam Administrator receives a confirmation message with the exam's unique identification number. |

### Scenario# S1.2

|  |  |  |
| --- | --- | --- |
| *Scenario#* ***S1.2*** | **:** | Add a New Exam with Invalid Information |
| **Prerequisite** | **:** | The Exam Administrator is logged into the Exam Management System. |
| **Acceptance**  **Criteria** | **:** | **Given:** The Exam Administrator is on the exam creation page.  **When:**  The Exam Administrator selects the "Create New Exam" option.  The Exam Administrator enters incomplete or incorrect exam details.  The Exam Administrator clicks the "Save" button to schedule the exam.  **Then:**  The system displays error messages for the incorrect or missing information. |

### Scenario# S1.3

|  |  |  |
| --- | --- | --- |
| *Scenario#* ***S1.3*** | **:** | Attempting to Create a Duplicate Exam |
| **Prerequisite** | **:** | The Exam Administrator is logged into the Exam Management System. |
| **Acceptance**  **Criteria** | **:** | **Given:** The exam details (e.g., exam name, date, time, and subject) are available, and an exam with the same details is already scheduled.  **When:**  The Exam Administrator clicks on the “Create New Exam” button.  The Exam Administrator enters details that match an existing exam. |
|  | | **Then**: Generate unique book id, barcode and spine label for various book of same title. |

### Scenario# S1.4

|  |  |  |
| --- | --- | --- |
| *Scenario#* ***S1.4*** | **:** | Editing an Existing Exam |
| **Prerequisite** | **:** | The Exam Administrator is logged into the Exam Management System, and the exam to be edited is already scheduled. |
| **Acceptance**  **Criteria** | **:** | **Given:** The Exam Administrator is on the exam management page and selects an existing exam from the list of scheduled exams.  **When:**  The Exam Administrator selects the "Edit Exam" option.  The Exam Administrator updates the exam details (e.g., date, time, duration, or subject).  The Exam Administrator clicks the "Save" button to update the exam information. |
|  | | **Then**: The system successfully updates the exam details.  The Exam Administrator receives a confirmation message |

### Scenario# S1.5

|  |  |  |
| --- | --- | --- |
| *Scenario#* ***S1.5*** | **:** | Deleting an Existing Exam |
| **Prerequisite** | **:** | The Exam Administrator is logged into the Exam Management System, and the exam to be deleted is already scheduled. |
| **Acceptance**  **Criteria** | **:** | **Given:** The Exam Administrator is on the exam management page and has selected an existing exam from the list of scheduled exams.  **When:**  The Exam Administrator selects the "Delete Exam" option.  The system prompts the Exam Administrator to confirm the deletion.  The Exam Administrator confirms the deletion. |
|  | | **Then**: The system successfully removes the exam from the schedule |

# Test cases

|  |  |  |  |
| --- | --- | --- | --- |
| **Project Name:** | **EMI Calculator** | **Test Designed by:** | Shreyansh patel |
| **Module Name:** | **Login** | **Test Designed date:** | 01-10-2023 |
| **Release Version:** | **1.0** | **Test Executed by:** | **R. B. Gondaliya** |
|  |  | **Test Execution date:** | 15-01-2023 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Pre-condition**: **Web application should be accessible** | | | |  |
| **Test Case ID** | **Test Title** | **Test Type** | **Description** | **Test Case ID** |
| **TC\_001** | Login to web application with valid credential | Functional | Login to Library management system web application through valid credential | TC\_001 |
| **TC\_002** | Verify login page elements | GUI | varify that all elements are availabe on login page | TC\_003 |

|  |  |
| --- | --- |
| Test Case Title | Login to web application with valid credential |
| Test Type | Functional |
| Test Priority | High |
| Precondition | Web application should be accessible |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Access the Exam Management System URL | The site should launch properly | Site launched successfully | Pass |  | https://exam.management.system/login |  |
| 2 | Log in with valid credentials | The system should navigate to the dashboard | Navigated to the dashboard | Pass |  | Username: admin@example.com, Password: Admin@123 |  |
| 3 | Navigate to the Exam Creation page | The system should display the Exam Creation page | Exam Creation page displayed | Pass |  |  |  |
| 4 | Enter valid exam details (e.g., exam name, date, time, duration) | The system should accept the input details | Input details accepted | Pass |  | Exam Name: Math Final, Date: 20-01-2024 |  |
| 5 | Click on the "Save" button | The system should successfully create the exam and display a confirmation message with a unique exam ID | Exam created successfully with ID: EXAM001 | Pass |  |  |  |

|  |  |
| --- | --- |
| Test Case Title | Create a New Exam with Invalid Information |
| Test Type | Functional |
| Test Priority | Medium |
| Precondition | Web application should be accessible |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Access the Exam Management System URL | The site should launch properly | Site launched successfully | Pass |  | https://exam.management.system/login |  |
| 2 | Log in with valid credentials | The system should navigate to the dashboard | Navigated to the dashboard | Pass |  | Username: admin@example.com, Password: Admin@123 |  |
| 3 | Navigate to the Exam Creation page | The system should display the Exam Creation page | Exam Creation page displayed | Pass |  |  |  |
| 4 | Enter invalid exam details (e.g., leave date field blank) | The system should display an error message for the invalid/missing details | Error message displayed | Pass |  |  |  |
| 4 | Verify that User is not able to Login with  blank  Username  or  Password | Set required field validation  message  for  Username and Password | Display an error of wrong username and password | Fail | Not performs  a validation  function  fix it |  | Bug\_002 |

|  |  |
| --- | --- |
| Test Case Title | Attempt to Create a Duplicate Exam |
| Test Type | GUI |
| Test Priority | Medium |
| Precondition | Web application should be accessible |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Access the Exam Management System URL | The site should launch properly | Site launched successfully | Pass |  | https://exam.management.system/login |  |
| 2 | Log in with valid credentials | The system should navigate to the dashboard | Navigated to the dashboard | Pass |  | Username: admin@example.com, Password: Admin@123 |  |
| 3 | Navigate to the Exam Management page and select an existing exam | The system should display the selected exam's details | Selected exam details displayed | Pass |  | Exam ID: EXAM001 |  |
| 4 | Update the exam details (e.g., change the date) | The system should accept the updated details | Details updated successfully | Pass |  | New Date: 25-01-2024 |  |
| 5 | Click on the "Save" button | The system should successfully update the exam and display a confirmation message | Exam updated successfully | Pass |  |  |  |

|  |  |
| --- | --- |
| Test Case Title | Attempt to Create a Duplicate Exam |
| Test Type | GUI |
| Test Priority | Medium |
| Precondition | Web application should be accessible |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Step** | **Test Case Description** | **Expected Result** | **Actual Result** | **Status** | **Comment** | **Data** | **Bug ID** |
| 1 | Access the Exam Management System URL | The site should launch properly | Site launched successfully | Pass |  | https://exam.management.system/login |  |
| 2 | Log in with valid credentials | The system should navigate to the dashboard | Navigated to the dashboard | Pass |  | Username: admin@example.com, Password: Admin@123 |  |
| 3 | Navigate to the Exam Management page and select an existing exam | The system should display the selected exam's details | Selected exam details displayed | Pass |  | Exam ID: EXAM001 |  |
| 4 | Update the exam details (e.g., change the date) | The system should accept the updated details | Details updated successfully | Pass |  | New Date: 25-01-2024 |  |
| 5 | Click on the "Save" button | The system should successfully update the exam and display a confirmation message | Exam updated successfully | Pass |  |  |  |

# References

* http://www.w3schools.com/html/html\_intro.asp
* https://www.w3schools.com/php/default.asp
* https://www.javatpoint.com/uml