|  |
| --- |
| **Requirements and Design Specification Document**  **Online Test Management System**  **Project Team:**  Hamza Zulfiqar (201370105)  Akasha Farrukh (201370031)  Anzah Nadeem (201370005)  Musa Nabeel (201370012)  Submission Date: 05 March, 2023 |

**Document Information**

| **Category** | **Information** |
| --- | --- |
| Customer |  |
| Project | Online Test Management System |
| Document | Requirement Specifications, Use Cases, Use Case Diagram, ERD, Work Breakdown Structure, Gantt Chart |
| Document Version | 1.0 |
| Identifier |  |
| Status | Draft |
| Author(s) | Hamza Zulfiqar, Anzah Nadeem, Musa Nabeel, Akasha Farrukh |
| Approver(s) |  |
| Issue Date |  |
| Document Location |  |
| Distribution |  |

**Table of Contents****:**

* Requirement Specifications
* Use Cases
* Use Case Diagram
* ERD
* Work Breakdown Structure
* Gantt Chart

## **Requirement:**

## **Business Requirements**

**BR: 01**. Business requirements encompass providing an online test management platform where students can attempt objective questions.

**BR: 02**. The student should be able to register to appear in the exam.

**BR: 03**. There should be an admin panel to load questions.

**BR: 04**. There should be an admin panel to check questions.

**BR: 05**. There should be an admin panel to add questions.

**BR: 06**. There should be an admin panel to drop questions.

**BR: 07**. There should be an admin panel to add subjects.

**BR: 08**. There should be an admin panel to drop questions.

**BR: 09**. Questions shall be segregated on the basis of different subjects.

**BR: 010**. The math section shall carry 50% marks, with 25 questions, and a 25-minute time slot.

**BR: 11**. The IQ section shall carry 30% marks, with 15 questions and 15-minute time slot.

**BR: 12**. The English section shall carry 20% marks, with 10 questions and a 10-minute time slot.

**BR: 13**. In total, the students should be given 55 minutes — 50 minutes for the exam and 5 minutes for an introductory questions session from each section.

**BR: 14**. There should be an option to quit session.

**BR: 15**. Stakeholders will only be permitted a maximum of two navigations to access any information.

## **User Requirements**

**<Student>**

**UR: 01**. Students should be able to register to appear in the exam.

**UR: 02**. The student should receive login information (and a password) after successful registration.

**UR: 03**. The student should be able to attempt objective questions.

**UR: 04**. The student should be able to quit the session(s) at any time.

**UR: 05**. The student should be awarded a grade immediately upon test completion.

**<Admin>**

**UR: 06**. The admin panel should be able to load questions.

**UR: 07**. The admin panel should be able to check questions and see if they are correctly attempted.

**UR: 08**. The admin should be able to add or drop subjects.

**UR: 09**. The admin should be able to add or subtract questions.

**UR: 10**. The admin panel should be able to save data.

**UR: 11**. The admin shall be allowed to provide login/password to the student.

**UR: 12**. The admin panel should be able to communicate with DB via web service or API.

## **Functional Requirements**

**FR: 01**. The system shall allow the students to register themselves to appear in the exam.

**FR: 02**. The system shall allow the admin will provide the student with login information (and password) after a successful registration.

**FR: 03**. The system shall provide a facility to admin load the questions.

**FR: 04**. Questions will be divided into categories based on various subjects. The math section shall carry 50% marks (25 questions, 25min), IQ 30% (15 questions, 15min) and English 20% (10 questions, 10min).

**FR: 05**. The system shall allow the admin to add subjects.

**FR: 06**. The system shall allow the admin to drop subjects.

**FR: 07**. The system shall allow the admin to add question.

**FR: 08**. The system shall allow the admin to drop questions

**FR: 09**. The system shall allow the Students to attempt the questions.

**FR: 10**. The system shall allow the admin to save the data and communicate with DB via web service or API.

**FR: 11**. The system shall allow the admin to publish the answers to all the questions right away.

**FR: 12**. The system shall allow the admin to publish results.

**FR: 13**. The system shall provide grades as per the number of questions answered correctly.

**FR: 14**. The system shall provide an option to quit the session(s) at any time.

## **Non-Functional Requirements**

**NFR: 01**. There will be a time limit check for finishing the exam, after which students won't be able to attempt any questions in their particular section.

**NFR: 02**. Only 50 students at a time are allowed to take the exam concurrently.

**NFR: 03**. Stakeholders will only be permitted a maximum of two navigations to access any information.

## **Use Case Diagram & Use Cases Scenarios**

## **Actors:**

* Student
* Admin

## **Use Cases:**

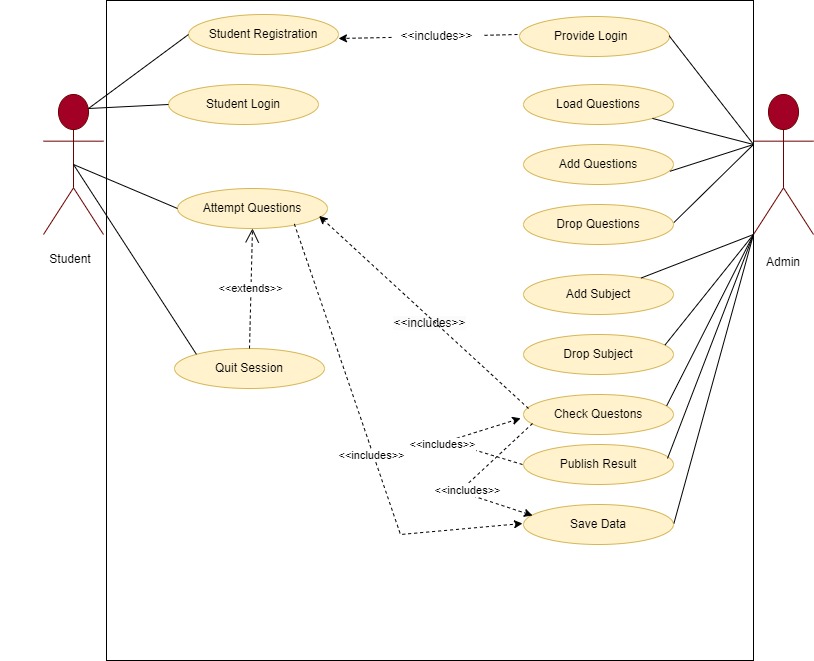
**Student:**

* Student registration
* Student login
* Attempt Questions
* Quit Session

**Admin:**

* Provide login
* Load questions
* Drop questions
* Add subject
* Drop subject
* Check questions
* Publish result
* Save data

## **Use Case Diagram**



## **Use Case Scenarios:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<UC-01: Student Registration>** | | | | | |
| **Actors:**  Student | | | | | |
| **Feature:** | | | By doing this student will be able to take part in exam. | | |
| **Use case Id:** | | | 01 | | |
| **Pre-condition:** | | | Must open website for registration and have valid email address. | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | **Software Reaction** | |
| **1** | A student wants to appear in the online exam. | | |  | |
| **2** | The student registers himself on the application by providing his full name, username, password, email address, and country. | | | Students’ credentials are inserted into the system. | |
| **3** |  | | | The system generates a login and password for the student, and notifies admin. | |
| **4** | Student registered. | | |  | |
| **Alternate Scenarios:** Write additional, optional, branching, or iterative steps. Refer to specific action numbers to ensure understandability. | | | | | |
| **Step#** | | **Action** | | | **Software Reaction** |
| **1a:** | | A student wants to appear in the online exam. | | |  |
| **2a:** | | The student registers using Facebook. | | | System obtains credentials from Facebook ID. |
| **3a:** | |  | | | The system generates a login and password for the student, and notifies admin. |
| **4a:** | | Student registered. | | |  |
| **Post Conditions** The student is registered and his credentials are added to the DB. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<UC-02: Admin Provides Login>** | | | | | |
| **Actors:**  Admin | | | | | |
| **Feature:** | | | Admin provides the login information to register students only. | | |
| **Use case Id:** | | | 02 | | |
| **Pre-condition:** | | | Admin is logged in to his account. Student has formerly registered and has provided his credentials. | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | **Software Reaction** | |
| **1** | Admin is notified that a new student registered himself. | | |  | |
| **2** | Admin checks credentials. | | | System provides the generated login and password to the admin. | |
| **3** | Admin provides login and password to the student. | | |  | |
| **Alternate Scenarios:** Write additional, optional, branching, or iterative steps. Refer to specific action numbers to ensure understandability. | | | | | |
| **Step#** | | **Action** | | | **Software Reaction** |
| **1a:** | | Admin is notified that a new student registered himself. | | |  |
| **2a:** | | Admin checks credentials. | | | System detects error in credentials (e.g., CNIC is incorrect) |
| **3a:** | | Admin notifies the student that credentials are incorrect and requests for correct credentials. | | |  |
| **4a:** | | Admin is provided correct credentials by student. | | | System verifies details and provides generated login and password to student. |
| **5a:** | | Admin provides generated login and password to student. | | |  |
| **Post Conditions** The student is provided with his login details. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<UC-03: Question Adding by Admin>** | | | | | |
| **Actors:**  Admin | | | | | |
| **Feature:** | | | Admin will be able to add question for exam | | |
| **Use case Id:** | | | 03 | | |
| **Pre-condition:** | | | Admin is logged in to his account. | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | **Software Reaction** | |
| **1** | The admin wants to add a question. | | |  | |
| **2** | Admin selects the “add question” option from the admin interface. | | | The system opens a webpage, requesting the subject, new objective question to be added, and its 4 options. | |
| **3** | Admin provides the objective question, its 4 options, and the subject it is to be added to. | | | The system adds the objective question to the given subject portion. | |
| **Alternate Scenarios:** Write additional, optional, branching, or iterative steps. Refer to specific action numbers to ensure understandability. | | | | | |
| **Step#** | | **Action** | | | **Software Reaction** |
| **1a:** | | The admin wants to add a question. | | |  |
| **2a:** | | Admin selects the “add question” option from the admin interface. | | | The system opens a webpage, requesting the subject, new objective question to be added, and its 4 options. |
| **3a:** | | Admin provides the objective question, less than 4 options, and the subject it is to be added to. | | | The system saves the objective question for later updating. |
| **4a:** | | Admin adds missing options. | | | The system adds the question to the subject. |
| **Post Conditions** The relevant question is added to the mentioned subject. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<UC-04: Question Deletion by Admin>** | | | | | |
| **Actors:**  Admin | | | | | |
| **Feature:** | | | Admin will be able to delete question for exam | | |
| **Use case Id:** | | | 04 | | |
| **Pre-condition:** | | | Admin is logged in to his account. The entered question number is present in the exam. | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | **Software Reaction** | |
| **1** | The admin wants to delete a question. | | |  | |
| **2** | Admin selects the “delete question” option from the admin interface. | | | The system requests the question number to be deleted. | |
| **3** | Admin provides the question number. | | | System requests for confirmation in a pop-up dialogue box. | |
| **4** | Admin confirms deletion. | | | The system deletes the question from the exam. | |
| **Alternate Scenarios:** Write additional, optional, branching, or iterative steps. Refer to specific action numbers to ensure understandability. | | | | | |
| **Step#** | | **Action** | | | **Software Reaction** |
| **1a:** | | The admin wants to delete a question. | | |  |
| **2a:** | | Admin selects the “delete question” option from the admin interface. | | | The system requests the question number to be deleted. |
| **3a:** | | Admin provides the question number. | | | System requests for confirmation in a pop-up dialogue box. |
| **4a:** | | Admin does not confirm deletion. | | | The system does not delete the question. |
| **5a:** | |  | | | The process repeats (1, 2, 3) and the admin confirms deletion, so the system deletes the question. |
| **Post Conditions** The relevant question is deleted from the exam. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<UC-05: Subject Adding by Admin>** | | | | | |
| **Actors:**  Admin | | | | | |
| **Feature:** | | | Admin will be able to add Subjects for exam | | |
| **Use case Id:** | | | 05 | | |
| **Pre-condition:** | | | Admin is logged in to his account. | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | **Software Reaction** | |
| **1** | The admin wants to add a subject. | | |  | |
| **2** | Admin selects the “add subject” option from the admin interface. | | | The system requests the admin to enter the subject name and code to be added. | |
| **3** | The admin provides the subject name and code. | | | The system asks for confirmation to add with a pop-up dialogue box. | |
| **4** | The admin confirms the addition. | | | The system adds the specified subject, with the code specified. | |
| **Alternate Scenarios:** Write additional, optional, branching, or iterative steps. Refer to specific action numbers to ensure understandability. | | | | | |
| **Step#** | | **Action** | | | **Software Reaction** |
| **1a:** | | The admin wants to add a subject. | | |  |
| **2a:** | | Admin selects the “add subject” option from the admin interface. | | | The system requests the admin to enter the subject name and code to be added. |
| **3a:** | | The admin provides the subject name and code. | | | The system asks for confirmation to add with a pop-up dialogue box. |
| **4a:** | | The admin does not confirm the addition. | | | The system does not add the subject and returns to normal working. |
| **5a:** | |  | | | The process repeats (1, 2, 3) and the admin confirms addition, so the system adds the subject. |
| **Post Conditions** The relevant subject is added to the exam. (Questions will be added to this subject via the “Question adding” scenario) | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<UC-06: Subject Deletion by Admin>** | | | | | |
| **Actors:**  Admin | | | | | |
| **Feature:** | | | Admin will be able to delete subjects for exam | | |
| **Use case Id:** | | | 06 | | |
| **Pre-condition:** | | | Admin is logged in to his account. The specified subject code is present in the exam. | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | **Software Reaction** | |
| **1** | The admin wants to delete a subject. | | |  | |
| **2** | Admin selects the “delete subject” option from the admin interface. | | | The system requests the admin to enter the subject code. | |
| **3** | The admin provides the subject code. | | | The system asks for confirmation through a pop-up dialogue box. | |
| **4** | The admin confirms the deletion process. | | | The system deletes the subject with the specified code. | |
| **Alternate Scenarios:** Write additional, optional, branching, or iterative steps. Refer to specific action numbers to ensure understandability. | | | | | |
| **Step#** | | **Action** | | | **Software Reaction** |
| **1a:** | | The admin wants to delete a subject. | | |  |
| **2a:** | | Admin selects the “delete subject” option from the admin interface. | | | The system requests the admin to enter the subject code. |
| **3a:** | | The admin provides the subject code. | | | The system asks for confirmation through a pop-up dialogue box. |
| **4a:** | | The admin does not confirm the deletion process. | | | The system does not delete the subject with the specified code. |
| **5a:** | |  | | | The process repeats (1, 2, 3) and the admin confirms deletion, so the system deletes the subject. |
| **Post Conditions** The relevant subject is deleted from the exam. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<UC-07: Question Loading by Admin>** | | | | | |
| **Actors:**  Admin | | | | | |
| **Feature:** | | | Admin will be able to load question before exam. | | |
| **Use case Id:** | | | 07 | | |
| **Pre-condition:** | | | Admin is logged in to his account. | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | **Software Reaction** | |
| **1** | The admin wants to load questions for students. | | |  | |
| **2** | Admin chooses the “load questions” option from the admin interface. | | | The system asks for confirmation to load questions at the current time via a pop-up dialogue box. | |
| **3** | Admin confirms the query. | | | The system loads the questions for students to attempt. | |
| **Alternate Scenarios:** Write additional, optional, branching, or iterative steps. Refer to specific action numbers to ensure understandability. | | | | | |
| **Step#** | | **Action** | | | **Software Reaction** |
| **1a:** | | The admin wants to load questions for students. | | |  |
| **2a:** | | Admin chooses the “load questions” option from the admin interface. | | | The system asks for confirmation to load questions at the current time via a pop-up dialogue box. |
| **3a:** | | Admin does not confirm the query. | | | The system does not load the questions yet. |
| **4a:** | |  | | | The process (1, 2, 3) repeats and admin confirms, so questions are loaded. |
| **Post Conditions** Questions are loaded for students to attempt. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<UC-08: Student Login>** | | | | | |
| **Actors:**  Student | | | | | |
| **Feature:** | | | Website have login page, where student is able to login and after logging in student will be able look his progress and able to take part in exam. | | |
| **Use case Id:** | | | 08 | | |
| **Pre-condition:** | | | The student is a fully registered person. | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | **Software Reaction** | |
| **1** | The student wants to login. | | |  | |
| **2** | The student enters his login and password. | | | The system will verify the login and password. | |
| **3** |  | | | The system logs the student in and takes him to his personalized student interface. | |
| **Alternate Scenarios:** Write additional, optional, branching, or iterative steps. Refer to specific action numbers to ensure understandability. | | | | | |
| **Step#** | | **Action** | | | **Software Reaction** |
| **1a:** | | The student wants to login. | | |  |
|  | | The student enters his login and password. | | | The system will verify the login and password. |
| **2a:** | |  | | | The login/password is incorrect. |
| **3a:** | |  | | | The system displays an error to the user and requests to re-enter credentials. |
| **4a:** | | The student enters his login and password again. | | | The system will verify the login and password. |
| **5a:** | |  | | | The system logs the student in and takes him to his personalized student interface. |
| **Post Conditions:** The student has logged into his/her account. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<UC-09: Exam Attempting by Student>** | | | | | |
| **Actors:**  Student | | | | | |
| **Feature:** | | | This will enable students to attempt questions in exam. | | |
| **Use case Id:** | | | 09 | | |
| **Pre-condition:** | | | The questions are loaded by the admin panel. The student is logged into his account. | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | **Software Reaction** | |
| **1** | The student wants to attempt the exam. | | |  | |
| **2** | The student selects the “Attempt exam” option on his logged-in student interface. | | | The system loads the objective questions for the student to attempt. | |
| **3** | The student attempts the exam question by choosing one of the 4 options provided. | | | The system stores the answer. | |
| **4** | The student completes the exam. | | | The system stores all answers in one place of the database, with the username of the password associated with the set of answers. | |
| **Alternate Scenarios:** Write additional, optional, branching, or iterative steps. Refer to specific action numbers to ensure understandability. | | | | | |
| **Step#** | | **Action** | | | **Software Reaction** |
| **3a:** | | The student repeats the process for each question of each section. (iterative) | | | The system stores each answer for each question. |
| **Post Conditions** The student has attempted all questions and answers are stored in the system database. | | | | | |
| **Step#** | **Description** | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<UC-10: Exam Checking by Admin>** | | | | | |
| **Actors:**  Admin | | | | | |
| **Feature:** | | | This will enable admin to check exam questions. | | |
| **Use case Id:** | | | 10 | | |
| **Pre-condition:** | | | The admin must be logged in. The student must have attempted the exam. The system must have stored all answers in the database. | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | **Software Reaction** | |
| **1** | The admin is notified that the student has completed the exam. | | |  | |
| **2** | The admin requests access to the stored answers. | | | The system retrieves data from the database and displays it on the admin interface. | |
| **3** | The admin identifies whether each question has been attempted correctly. | | | The system stores the result of each identification — using 1 for correct answers and 0 for incorrect answers. | |
| **4** | The admin will complete the checking of one particular exam. | | | The system will calculate the total number of correctly attempted questions. | |
| **5** |  | | | The system calculates the grade based on a preset percentage: 90+ (A+), 80+ (A), 70+ (B), 60+ (C), 50+ (D), 40+ (E), under 40 (F) | |
| **6** |  | | | The system will declare results, making it visible to the student. | |
| **Alternate Scenarios:** Write additional, optional, branching, or iterative steps. Refer to specific action numbers to ensure understandability. | | | | | |
| **Step#** | | **Action** | | | **Software Reaction** |
| **3a:** | | The admin repeats the process for each question of each section. (iterative) | | | The system stores each result for each question. |
| **Post Conditions** The result is calculated and displayed on the student interface. | | | | | |
| **Step#** | **Description** | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<UC-11: Quit Session by Student>** | | | | | |
| **Actors:**  Student | | | | | |
| **Feature:** | | | This option will enable student to quit session any time. | | |
| **Use case Id:** | | | 11 | | |
| **Pre-condition:** | | | The student must be logged in. The student should be attempting the exam. | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | **Software Reaction** | |
| **1** | The student wants to quit the session. | | |  | |
| **2** | The student selects the “Quit” option displayed on the bottom of the screen. | | | The system asks for a confirmation to quit via a pop-up dialogue box. | |
| **3** | The student confirms. | | | The system exits and logs out the student. | |
| **Alternate Scenarios:** Write additional, optional, branching, or iterative steps. Refer to specific action numbers to ensure understandability. | | | | | |
| **Step#** | | **Action** | | | **Software Reaction** |
| **1a:** | | The student wants to quit the session. | | |  |
| **2a:** | | The student selects the “Quit” option displayed on the bottom of the screen. | | | The system asks for a confirmation to quit via a pop-up dialogue box. |
| **3a:** | | The student does not confirm. | | | The student does not exit. |
| **4a:** | |  | | | The process repeats iteratively (1, 2, 3) and the student confirms. The system exits and logs out the student. |
| **Post Conditions** The student has quit the session. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **<UC-12: Data Saving by Admin>** | | | | | |
| **Actors:**  Admin | | | | | |
| **Feature:** | | | This feature will give access to admin to save data of exam. | | |
| **Use case Id:** | | | 12 | | |
| **Pre-condition:** | | | The admin must be logged in. | | |
| **Scenarios** | | | | | |
| **Step#** | **Action** | | | **Software Reaction** | |
| **1** | The admin wishes to save data. | | |  | |
| **2** | The admin requests the system to save using the “save” option. | | | The system saves the relevant data to the database. | |
| **Alternate Scenarios:** Write additional, optional, branching, or iterative steps. Refer to specific action numbers to ensure understandability. | | | | | |
| **Step#** | | **Action** | | | **Software Reaction** |
| **1a:** | | The admin wishes to save data. | | |  |
| **2a:** | | The admin requests the system to save using the “save” option. | | | Memory is full in the system. |
| **3a:** | |  | | | The system requests admin to delete some data. |
| **4a:** | | Admin deletes data and frees memory. | | | The system saves the relevant data to the database. |
| **Post Conditions** The data is saved to the database. | | | | | |

**Agile Stories**

**Stories and their acceptance criteria:**

**1**. As a student, I want to be able to register for the online test application, so that I can participate in the exam.

**Acceptance Criteria:**

* The registration page should have fields for name, email, password, and confirm password.
* The system should validate the email format and check if the email is already registered.
* The password should have minimum 8 characters and should match the confirm password field.
* Upon successful registration, the system should redirect the user to the login page.

**3W’s**

* Who: student
* What: register for the online test application
* Why: to participate in the exam

**3C’s**

**Card**:

Online Test Application Registration

**Conversation**:

As a student, I want to be able to register for the online test application, so that I can participate in the exam.

**Confirmation**:

Confirmation of Eligibility: Before the student can register for the online test application, the system should verify that the student meets the eligibility criteria. This confirmation will ensure that only eligible students can register for the exam and participate in it.

Confirmation of Registration: Once the student has completed the registration process, the system should confirm that the registration has been successful. This confirmation will give the student assurance that they have successfully registered and can participate in the exam.

Confirmation of Payment: If there is a fee for the online test application, the system should confirm that the payment has been made successfully. This confirmation will ensure that the student has paid the fee and can participate in the exam without any issues.

**2**. As a student, I want to be able to login to the online test application, so that I can access the exam.

**Acceptance Criteria:**

* The login page should have fields for email and password.
* The system should validate the email format and check if the email and password match the registered data.
* Upon successful login, the system should redirect the user to the dashboard page.

**3W’s**

* Who: student
* What: login to the online test application
* Why: to access the exam

**3C’s**

**Card**:

Student wants to login to online test application to access exam.

**Conversation**:

Student:

I need to access my exam on the online test application, but I don't know how to login.

Tech Support:

Sure, we can help you with that. Can you provide us with your login credentials?

Student:

Yes, my username is [username] and my password is [password].

Tech Support:

Great, let me guide you through the login process.

**Confirmation:**

Tech Support:

You have successfully logged in to the online test application. Can you see the exam now?

Student:

Yes, I can. Thank you for your help!

**3**. As an admin, I want to be able to load questions to the online test application, so that students can attempt them.

**Acceptance Criteria:**

* The admin panel should have a section for managing questions and categories.
* The admin should be able to add, edit, and delete questions and categories.
* The questions should be divided into three categories: Math, IQ, and English.
* The Math category should have 25 questions, the IQ category should have 15 questions, and the English category should have 10 questions.

**3W’s**

* Who: admin
* What: load questions to the online test application
* Why: so that students can attempt them

**3C’s**

**Card**: Admin wants to load questions to the online test application for students to attempt.

**Conversation:**

Admin:

Hi, I want to be able to load questions to the online test application.

Developer:

Sure, we can add that feature. Can you give me more details on what kind of questions you want to add and how many?

Admin:

Yes, I want to add multiple choice questions and maybe some short answer questions. There will be around 50 questions per test.

Developer:

Got it. And how do you want to load these questions? Will you be using a CSV file or do you want to manually enter them one by one?

Admin:

I prefer to use a CSV file for faster loading.

Developer:

Alright, we'll make sure to include that in the feature.

**Confirmation:**

Developer:

We've added the feature for you to load questions using a CSV file. You can now easily upload your questions and students can start attempting them. Let us know if you need any further assistance.

Admin:

Thank you, that's exactly what I needed. I appreciate your help.

**4**. As a student, I want to be able to see the instructions and time limits for each category, so that I can plan my exam strategy.

**Acceptance Criteria:**

* The dashboard page should display the time limits and question counts for each category.
* The system should provide an introductory session for each category, which lasts for 5 minutes and does not count towards the exam time.
* The system should prevent the student from answering questions after the time limit for each category has elapsed.

**3W’s**

* Who: student
* What: see instructions and time limits for each category
* Why: to plan exam strategy

**3C’s**

**Card:**

"As a student, I want to be able to see the instructions and time limits for each category."

**Conversation:**

Student:

"It would be really helpful if we could see the instructions and time limits for each category before the exam starts."

Teacher:

"I agree. It can be difficult to plan your strategy without knowing those details."

Student:

"Yes, I always feel like I'm rushing through one section because I didn't realize how much time I had left."

**Confirmation:**

The exam software now includes a feature where students can view the instructions and time limits for each category before beginning the exam, allowing them to better plan their strategy and use their time efficiently.

**5**. As a student, I want to be able to attempt questions in each category, so that I can complete the exam.

**Acceptance Criteria:**

* The exam page should display the questions one by one, with radio button options for each question.
* The system should prevent the student from navigating to the next question before answering the current question.
* The system should automatically save the answers as the student proceeds through the questions.

**3W’s**

* Who: student
* What: attempt questions in each category
* Why: to complete the exam

**3C’s**

**Card:**

As a student attempting an exam, the goal is to complete the test by answering questions in each category.

**Conversation:**

The student should start by reviewing the exam and identifying the different categories of questions. Then, they can prioritize their approach and attempt at least one question in each category to ensure that they cover all the required material. It is important to manage their time effectively, so they have enough time to answer all the questions.

**Confirmation:**

The student should double-check that they have answered at least one question in each category before submitting the exam. This will help ensure that they have not missed any essential questions and increase their chances of scoring well. It is always a good idea to review their answers and ensure that they have not made any mistakes before submitting the test.

**6**. As an admin, I want to be able to see the results of the online test application, so that I can evaluate the students' performance.

**Acceptance Criteria:**

* The admin panel should have a section for viewing and managing the exam results.
* The system should calculate the grades for each category based on the number of correct answers.
* The system should display the overall grade for the exam and the grades for each category.

**3W’s**

* Who: admin
* What: see the results of the online test application
* Why: to evaluate the students' performance

**3C’s**

**Card:**

Feature: Online Test Results Viewing

As an admin, I want to be able to see the results of the online test application, so that I can evaluate the students' performance.

**Conversation:**

Product Owner:

"We need to add a feature that allows admins to view the results of the online tests."

Developer:

"Alright, so you want a page where the admin can see a list of all the tests taken and their corresponding scores?"

Product Owner:

"Yes, and they should be able to filter the results by test type, date range, and student name."

Developer:

"Got it. We'll also need to make sure the page is secure so that only admins can access it."

**Confirmation:**

Product Owner:

"Can you show me a mockup of what the results page will look like?"

Developer:

"Sure, here's a design I came up with. The admin can select the filters they want to use and then see the results in a table format."

Product Owner:

"Looks good. Let's also make sure we test this thoroughly to ensure the data is accurate and the filters are working properly."

**7.** As a student, I want to be able to quit the exam session, so that I can end the exam if necessary

**Acceptance Criteria**:

* The exam page should have a button for quitting the exam.
* The system should display a confirmation message before quitting the exam.
* The system should save the answers and end the exam session upon confirmation.

**3W’s**

* Who: student
* What: quit the exam session
* Why: to end the exam if necessary

**3C’s**

**Card:**

User Story: As a student, I want to be able to quit the exam session, so that I can end the exam if necessary.

**Conversation:**

Developer:

Hi, can you tell me more about why you need the ability to quit the exam session?

User:

Sure, sometimes during an exam, unexpected things can happen, like feeling ill or having an emergency. I want to be able to end the exam and leave if I really must

.

Developer:

That makes sense. Would you like a confirmation message to pop up asking if you're sure you want to quit the exam session?

User:

Yes, that would be helpful to make sure I don't accidentally quit the exam.

**Confirmation:**

Developer: Great, I'll add a "Quit Exam" button to the interface, and when you click it, a confirmation message will appear. If you choose "Yes," the exam will end, and your results will be saved up to that point. Does that sound good?

User: Yes, that's perfect. Thank you!

Developer: You're welcome! Let me know if you have any other suggestions or feedback.

User: Will do, thanks again.

Developer: No problem, have a great day!

**8.** As an admin, I want to be able to add or drop subjects and questions, so that I can manage the content of the exam.

**Acceptance Criteria**:

* The admin panel should have a section for managing the exam content.
* The admin should be able to add or drop categories and questions.
* The system should update the exam content and data accordingly.

**3W’s**

* Who: admin
* What: add or drop subjects and questions
* Why: to manage the content of the exam

**3C’s**

**Card:**

Admin Subject and Question Management

**Conversation:**

Admin:

Hi, I would like to have the ability to add or drop subjects and questions in the exam.

Developer: Sure, we can add that feature for you. Do you have any specific requirements for this?

Admin:

Yes, I want to be able to easily manage the content of the exam. I should be able to add new subjects and questions or delete the existing ones as and when required.

Developer:

Got it. We can provide you with a simple interface where you can add, edit, or delete subjects and questions with ease. We will also ensure that the changes made to the exam are updated in real-time and are reflected on the platform.

Admin:

That sounds great. How will I know if the changes have been successfully made?

Developer:

You will receive a confirmation message once the changes have been made. We will also keep a log of all the changes made by you for future reference.

**Confirmation:**

The admin can easily add or drop subjects and questions in the exam using the provided interface. The changes made are updated in real-time and the admin receives a confirmation message once the changes have been successfully made. A log of all the changes made is also maintained for future reference.

**9**. As a student, I want to be informed if I am approaching the time limit for the exam, so that I can manage my time effectively and complete the exam on time.

**Acceptance Criteria:**

* The exam page should display a timer that counts down from the total exam time.
* The system should provide a warning message when the remaining time is 5 minutes or less.
* The system should end the exam automatically when the remaining time reaches zero.

**3W’s**

* Who: The story revolves around a student.
* What: The student wants to be informed when they are approaching the time limit for an exam so that they can manage their time effectively and complete the exam on time.
* Why: The student wants to ensure that they can complete the exam on time and avoid losing marks due to not finishing the exam.

**3C’s**

**Card:**

A notification system that informs students when they are approaching the time limit for the exam.

**Conversation:**

The notification can be integrated into the exam software and set to alert students when they have a certain amount of time left. The alert could appear on the screen or be accompanied by a sound or vibration. Students can also be given the option to dismiss the notification or snooze it for a few minutes.

**Confirmation:**

The notification system can be tested and piloted with a small group of students to gather feedback and make improvements. Once the system is refined, it can be rolled out to all students taking exams on the platform. Students can be given clear instructions on how the notification system works and its purpose, as well as the consequences of not managing their time effectively during the exam.

**10.** As a stakeholder, I want to be able to access information about the online test application, so that I can understand its features and benefits and make an informed decision.

**Acceptance Criteria:**

* The online test application should have a landing page that provides an overview of the features and benefits.
* The landing page should have a navigation menu that allows stakeholders to access relevant information, such as FAQs, pricing, and contact details.
* The navigation menu should be limited to two levels and use clear labels for easy understanding.

**3W’s**

* Who: The stakeholder.
* What: Wants to access information about an online test application.
* Why: To understand its features and benefits and make an informed decision.

**3C’s**

**Card:**

Title:

Online Test Application Information Access

Description:

As a stakeholder, I want to be able to access information about the online test application, so that I can understand its features and benefits and make an informed decision.

**Conversation:**

Stakeholder:

Can I get more information about the online test application?

Team Member:

Of course, we have a dedicated section on our website with all the information you need to make an informed decision. We can also set up a call to discuss any specific questions or concerns you may have.

Stakeholder:

That would be great, thank you. Can you tell me about any unique features of the application?

Team Member:

Absolutely. One of the unique features is the ability to customize the test questions and format to fit your specific needs. We also offer real-time monitoring and reporting to track student progress and identify areas for improvement.

**Confirmation:**

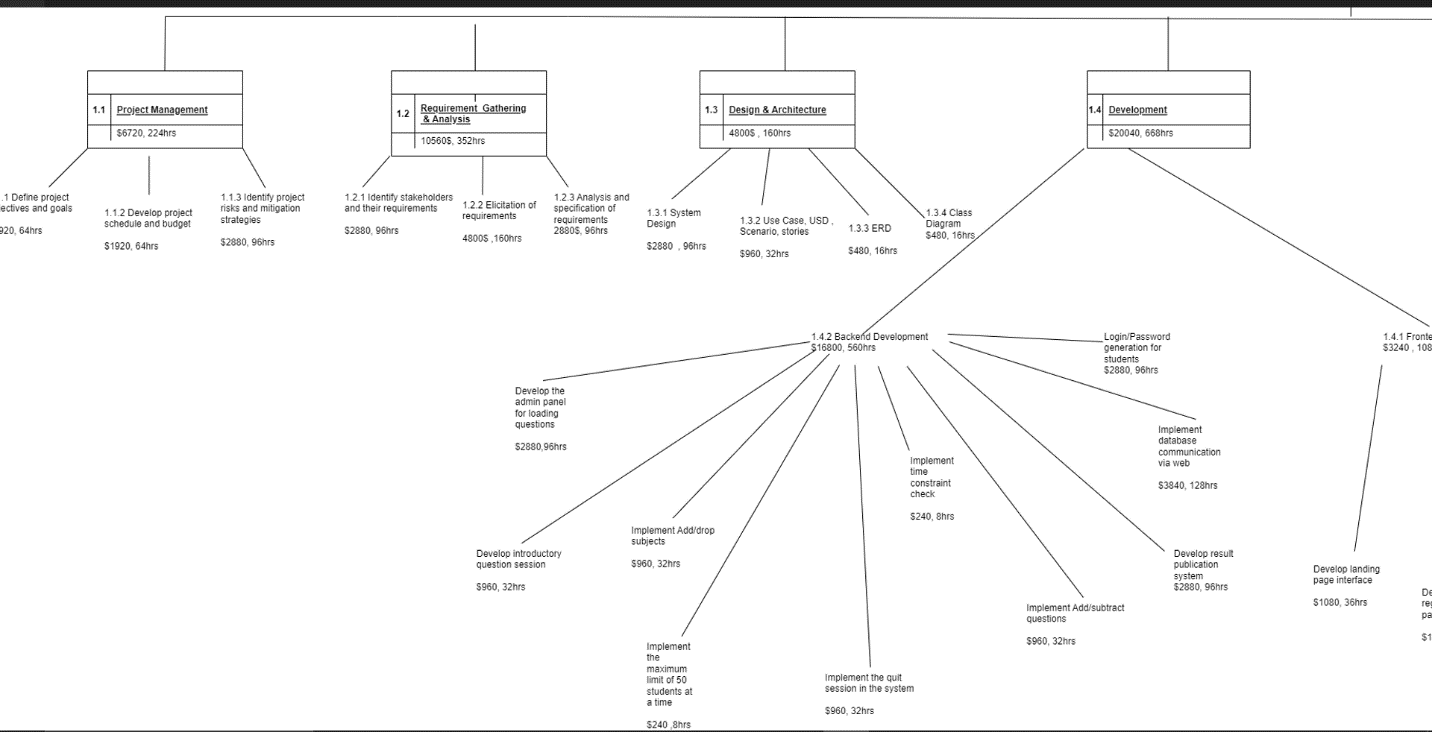
Stakeholder:

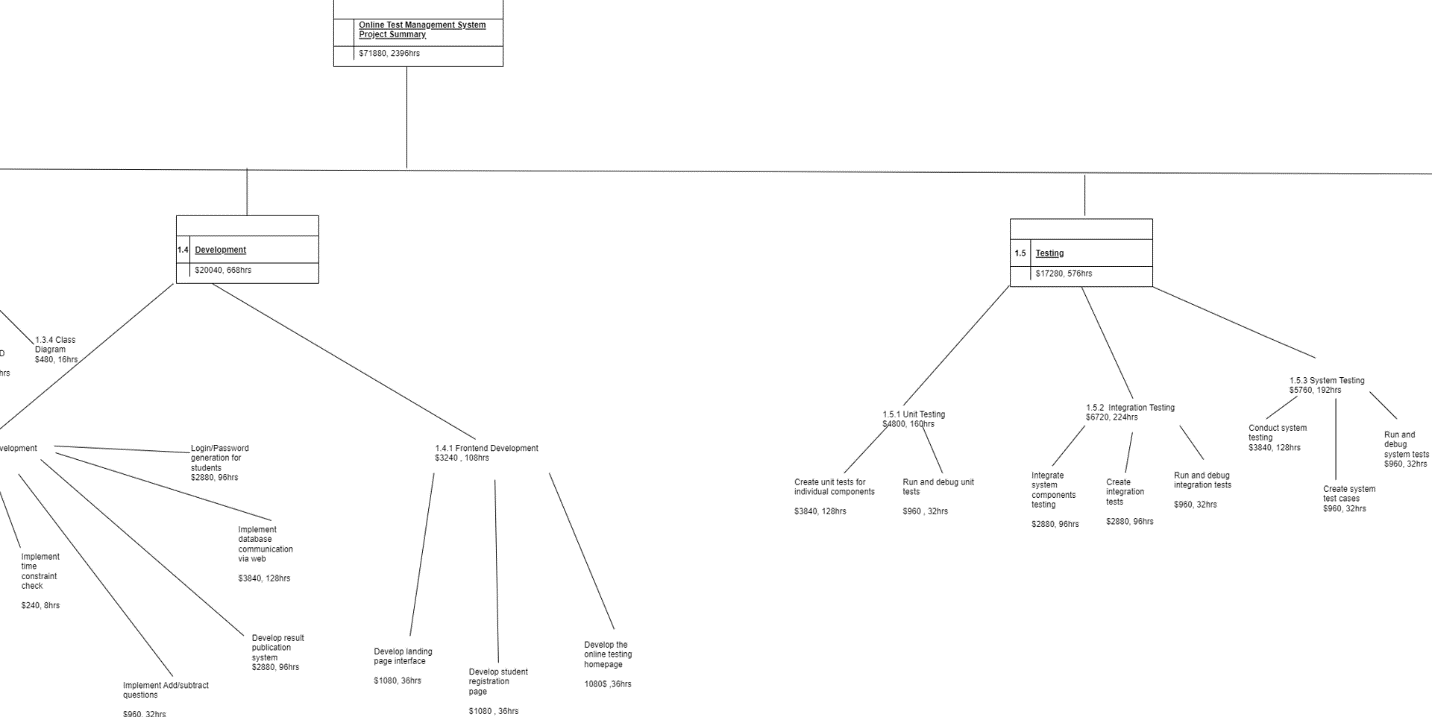
Thank you for providing me with more information about the online test application. I feel much more confident in my decision now.

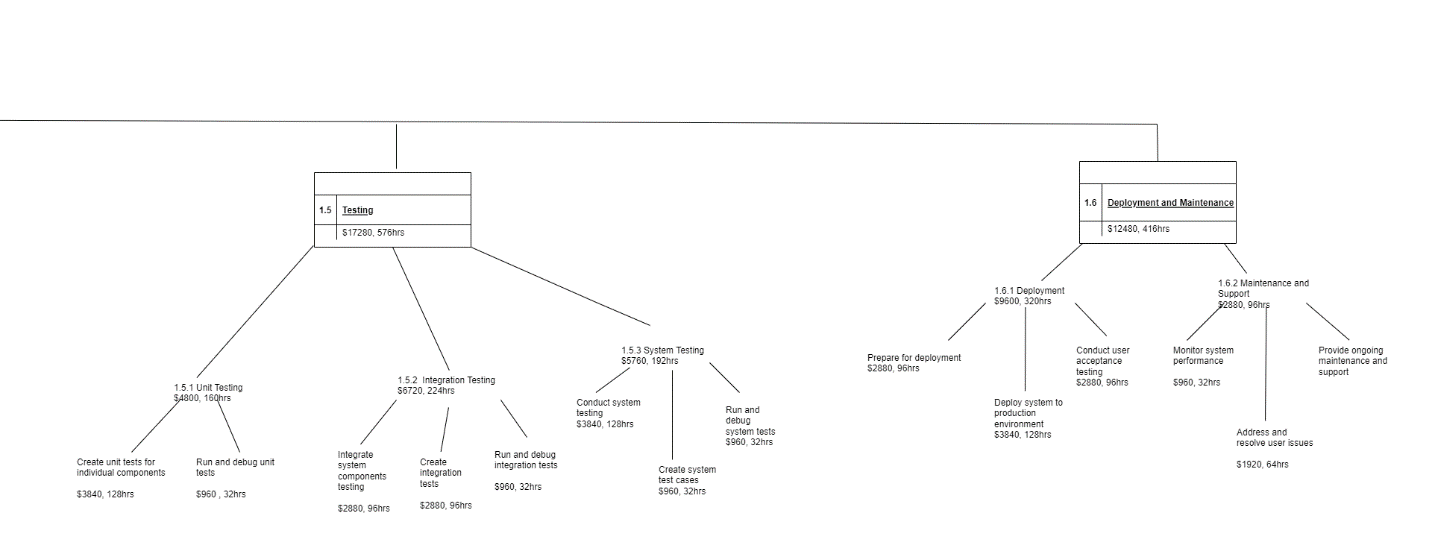
Team Member:

You're welcome. If you have any further questions or need additional information, please don't hesitate to reach out to us.

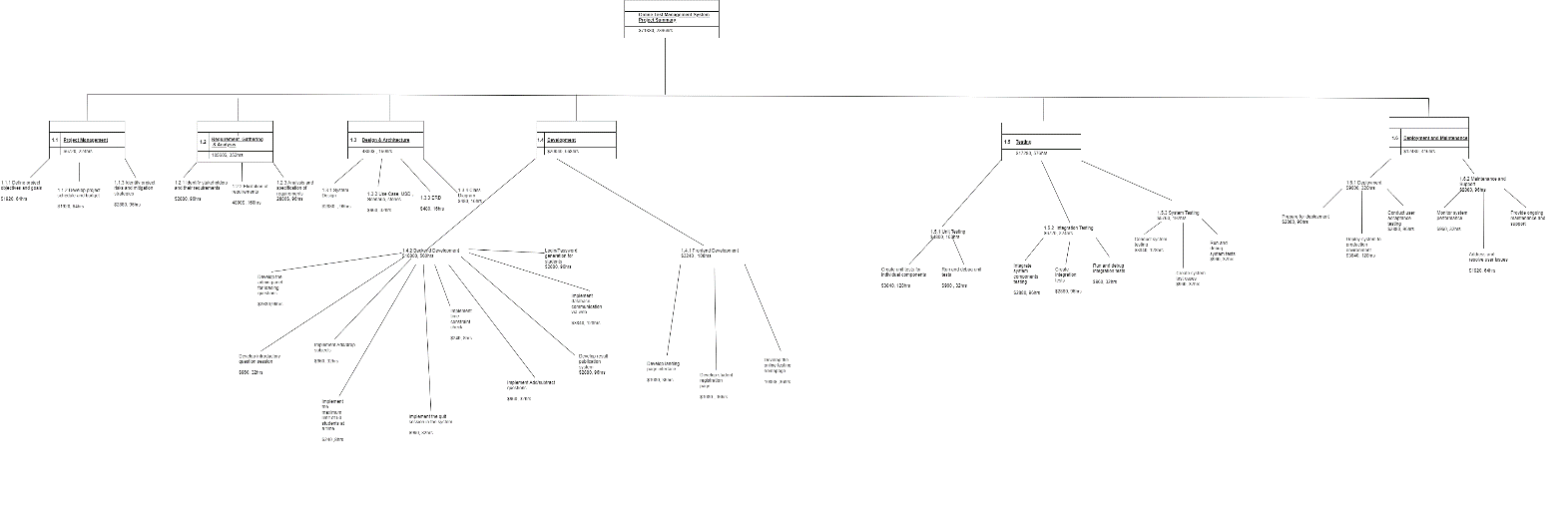
**Work Breakdown Structure**



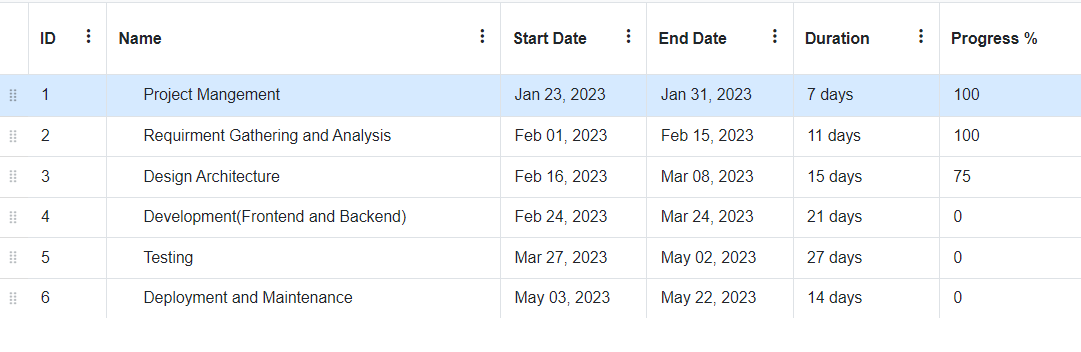


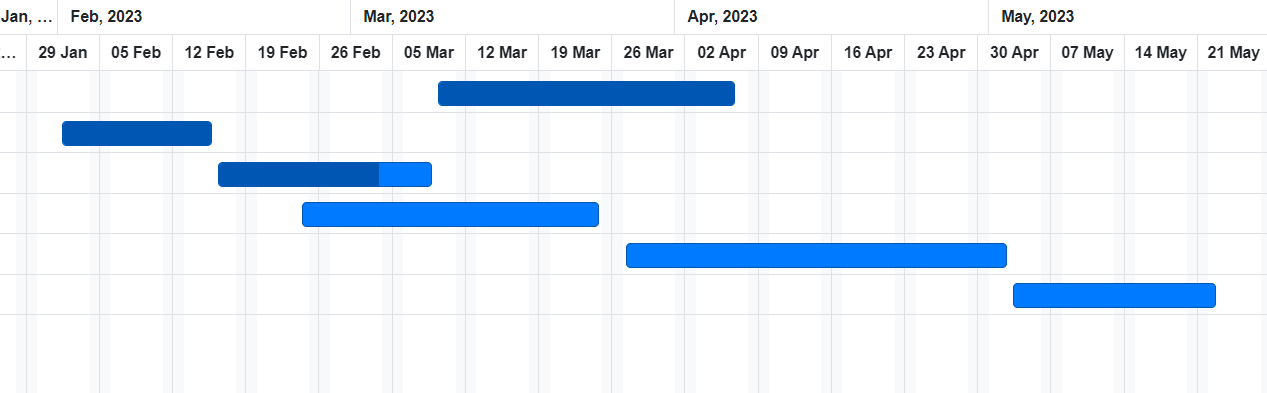


**Whole Picture of WBS:**

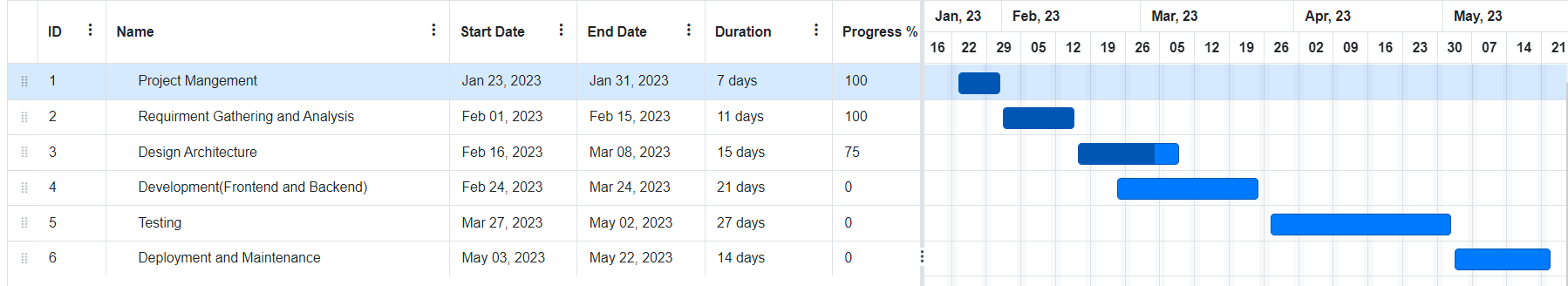


**Gantt Chart**





**Whole Picture of Gantt Chart:**



**Entity Relationship Diagram**

