Project Title	**Online Examination System**
Technologies	MERN
Domain	Education
Project Level	Difficult

Table

Table

1. **Problem Statement:** 2

..... 2

2. **Features:** 2

3. **Project Evaluation metrics:** 3

3.2. Database: 3

3.3. API Details or User Interface: 3

3.4. Deployment: 3

3.5. Solutions Design: 3

3.6. System Architecture: 3

3.7. Optimization of solutions: 3

4. Submission requirements:

..... 4

4.1. High-level Document: 4

4.2. **Low-level document:** 4

4.3. **Architecture:** 5

4.4. **Wireframe:** 5

4.5. **Project code:** 5

4.6. **Detail project report:** 5

4.7. **Project demo video:** 5

1. Problem Statement:

The system for online exams is a web application. This kind of test is comprised of multiplechoice questions. It offers a user-friendly environment for both students and test administrators. It creates a network between educational institutions and students. Institutions enter the exam questions they desire on the website. The qualified students see these questions as a test. The student's responses are then reviewed, and a score is computed and kept. The institutes can then use this score to identify the students who passed or to assess their performance. It has several features for building question and answer databases, as well as mechanisms for authorizing users so that only approved users may access this program. There are typically two sorts of authorization: administrator and user. The administration is in charge of entering test questions and answers into that software. A vital requirement is admin authorization. User authentication might be easy. Automatic evaluation and result calculation are included. It offers different test summaries to students and test administrators.

2. Features:

- Registration
- Login
- Add Institutes
- Update/Delete Institutes
- Create Questions and Papers
- View Results for Admin
- Take exam
- View Results for User
- Logout

3. Project Evaluation metrics:

3.1. Code:

- You are supposed to write code in a modular fashion ● Safe: It can be used without causing harm.
- Testable: It can be tested at the code level.
- Maintainable: It can be maintained, even as your codebase grows.
- Portable: It works similarly in every environment (operating system).
- You have to maintain your code on GitHub.
- You must keep your GitHub repo public so anyone can check your code.
- Proper readme file you have to maintain for any project development.
- You should include the basic workflow and execution of the entire project in the readme file on GitHub.
- Follow the coding standards.

3.2. Database:

MongoDB is a source-available cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas.

3.3. API Details or User Interface:

You have to expose your complete solution as an API or try to create a user interface for your model testing. Anything will be fine for us.

3.4. Deployment:

Deploy the application on your preferred service.

3.5. Solutions Design:

You have to submit complete solution design strategies in High-level Document (HLD), Lowlevel Document (LLD), and Wireframe documents**.**

3.6. System Architecture:

You have to submit a system architecture design in your wireframe document and architecture document.

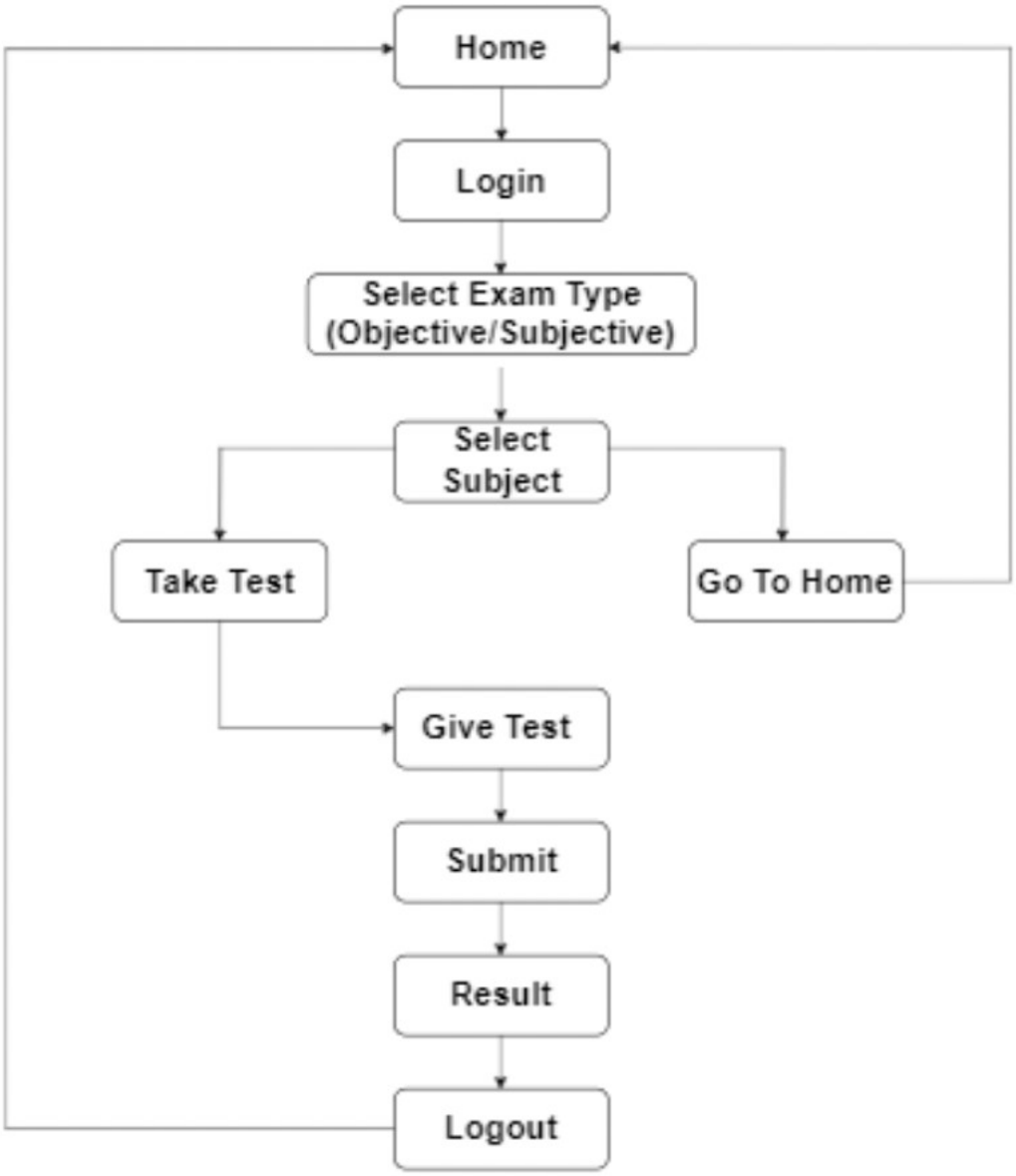
3.7. Optimization of solutions:

Try to optimize your solution on the code level, and architecture level, and mention all of these things in your final submission.

4. Submission requirements:

4.1. High-level Document:

You have to create a high-level document design for your project. You can reference the HLD form below the link.



Sample link: HLD Document Link

4.2. Low-level document:

You have to create a Low-level document design for your project; you can refer to the LLD from the link below.

Sample link: LLD Document Link

4.3. Architecture:

You have to create an Architecture document design for your project; you can refer to the Architecture from the link below.

Sample link: Architecture sample link

4.4. Wireframe:

You have to create a Wireframe document design for your project; refer to the Wireframe from the link below.

Demo link: Wireframe Document Link

4.5. Project code:

You have to submit your code to the GitHub repo in your dashboard when the final submission of your project.

Demo link: Project code sample link

4.6. Detail project report:

You have to create a detailed project report and submit that document as per the given sample.
Demo link: DPR sample link

4.7. Project demo video:

You have to record a project demo video for at least 5 Minutes and submit that link.