Table of Contents

[1. INTRODUCTION: 3](#_Toc139468515)

[1.1 PURPOSE 3](#_Toc139468516)

[1.2 DOCUMENT CONVENTIONS 3](#_Toc139468517)

[1.3 INTENDED AUDIENCE AND READING SUGGESTIONS 3](#_Toc139468518)

[1.4 PROJECT SCOPE 3](#_Toc139468519)

[2. OVERALL DESCRIPTION 4](#_Toc139468520)

[2.1 USER CLASS and CHARACTERISTICS 4](#_Toc139468521)

[2.2 OPERATING ENVIRONMENT 4](#_Toc139468522)

[3. SYSTEM FEATURES 5](#_Toc139468523)

[1. Dashboards for each type of users 5](#_Toc139468524)

[2. Admin customization and creation for each subject 5](#_Toc139468528)

[3. Make admin users able to customize and create and store questions 6](#_Toc139468532)

[4. Exam Creation 6](#_Toc139468536)

[5. Submit admission request by non-admin teachers to be able to create questions 7](#_Toc139468540)

[6. Submit a request by non-admin teachers to create new question 7](#_Toc139468544)

[7. Alerts 7](#_Toc139468547)

[8. Feedback System 7](#_Toc139468550)

[9. Save questions in favorites by student users 8](#_Toc139468553)

[10. Emails 8](#_Toc139468555)

[11. Direct chat 8](#_Toc139468558)

[12. Save Questions in drafts for question-creator users 8](#_Toc139468560)

[4. EXTERNAL INTERFACE REQUIREMENTS 8](#_Toc139468561)

[4.1 USER INTERFACES: 8](#_Toc139468562)

[4.2 HARDWARE INTERFACES: 9](#_Toc139468563)

[4.3 SOFTWARE INTERFACES: 9](#_Toc139468564)

[4.4 COMMUNICATION INTERFACES 9](#_Toc139468565)

[5. OTHER NONFUNCTIONAL REQUIREMENTS 9](#_Toc139468566)

[5.3 SECURITY REQUIREMENTS 9](#_Toc139468567)

[5.4 SOFTWARE QUALITY ATTRIBUTES 10](#_Toc139468568)

# 1. INTRODUCTION:

## 1.1 PURPOSE

The purpose of this document is to build a company/enterprise level website specialized in education field scoop, It’s main aim is to let students able to examine themselves to gain knowledge by Q&A strategy and to be prepared to face actual exams.

## 1.2 DOCUMENT CONVENTIONS

This document uses the following conventions.

|  |  |
| --- | --- |
| DB | Database |
| ERD | Entity Relationship diagram |
| CSR | Client-Side Rendering |
| SPA | Single page application |

## 1.3 INTENDED AUDIENCE AND READING SUGGESTIONS

Target audience for this writing is software development team who will contribute in building this project including developers, designer, content-creators and not only for software development team but also for the project owner.

To assist readers in gaining a deeper understanding of this document’s content, there are provided links in recommended resources These may include books , academic articles, or websites that provide additional information on the topic.

This software application assumes a basic level of technical knowledge and will use some technical terms in section 1.2. However, the application will be designed to be accessible to those without a technical background. To ensure that users are able to understand the technical concepts presented in the application, the application will provide definitions for any technical terms used. Additionally, the application will provide links to external resources that provide more in-depth explanations of technical concepts, for users who are interested in learning more.

## 1.4 PROJECT SCOPE

The project scoop includes design, development, and deployment of a new online examination system, before creating exams, this project will make teacher users able to create questions and store them in the DB with descripting these questions with several details that will make the system able to group it with other questions in the right category, then after all of that, system will be able to generate questions according to users entries to create a specially customized exam for him.

And finally system will include statistical methods to encourage all types of users including questions creators and examinees.

# 2. OVERALL DESCRIPTION

## 2.1 USER CLASS and CHARACTERISTICS

Administrator user should have the following functionalities

* Accept and reject questions that are submitted by teachers
* Change the color theme, logo, and description for each subject
* Have a statistics that are represented by graphics for number of solved exams in each subject and ranking for teachers that are submitting questions
* Can access reviews and feedbacks sent by students
* Can respond to all types of questions

Teacher user should have the following functionalities

* Submit a request to create new questions
* Graph of accepted and rejected questions that are submitted by them.
* May have and exam demo.
* Save questions in drafts.
* Can only access reviews that are typed by students for them, or reviews that are created by admin teachers
* Can respond to questions for questions that he created only

Student user should have the following functionalities

* Solve exams
* Customize exams by customizing time (systems should predict time), number of questions, level of hardness and questions that are created by a specific teacher.
* Direct chat with teachers
* Add questions to favorites
* Degrees graph that represent their marks and level of question
* Time progress in solving questions according to time and difficulty level.
* No of solved exams
* Can sent reviews and feedbacks to the system and a part of this review can be sent by to the teacher.

## 2.2 OPERATING ENVIRONMENT

It will run on a cloud-based infrastructure.

# 3. SYSTEM FEATURES

Features are prioritized using Moscow prioritization method.

## Dashboards for each type of users

### Description and priority:

Must-have

### Stimulus/Response Sequences

The stimulus for the creation of each dashboard is the login credentials of the user. When a user logs into the system, the system will use their credentials to identify their user type and retrieve the relevant data for their dashboard.

The response sequence for each dashboard will vary depending on the user type and their specific needs. However, in general, the dashboard should display relevant data in a clear and concise manner, with the ability to drill down into more detailed information if necessary.

### Functional Requirements

* **Customization:** Each user should be able to customize their dashboard to suit their specific needs. This may include selecting which data is displayed, changing the layout of the dashboard, and setting alerts for key performance indicators.
* **Real-time data:** The dashboard should display real-time data whenever possible, to ensure that users have access to the most up-to-date information.
* **Accessibility:** The dashboard should be accessible from any device with an internet connection, and should be optimized for both desktop and mobile viewing.
* **Security:** The dashboard should be protected by appropriate security measures, such as user authentication and authorization, to ensure that sensitive data is not accessible to unauthorized users.

## Admin customization and creation for each subject

### Description and priority:

Must-have

### Stimulus/Response Sequences

The stimulus for this feature is when the system administrator accesses the create subject page. The response sequence includes the ability to create a new one.

### Functional Requirements

* The system must allow the administrator to create a new subject and input relevant information such as name, age, gender, and any other relevant details.
* The system must allow the administrator to customize settings for each subject, such as their access permissions, data visualization preferences, and notification settings.

## Make admin users able to customize and create and store questions

### Description and priority:

Must-have

### Stimulus/Response Sequences

The stimulus for this feature is when the system administrator accesses the create subject page. The response sequence includes the ability to create a new one.

### Functional Requirements

* + - The system must allow the administrator to create a new question and input relevant information such as question, answer, question type, and any other relevant details.
    - The system must allow the administrator to customize settings for each subject, such as their access permissions, data visualization preferences, and notification settings.

## Exam Creation

### Description and priority:

Should-have

### Stimulus/Response Sequences

When any type of users visit exam customization page and input exam preferences in terms of number, type and difficulty level of questions.

The response of this stimulus should be choosing the most suitable questions according to user entries to use them in that exam.

### Functional Requirements

* + - User should choose number, type and difficulty level of the questions.
    - User should choose exam duration.
    - System should provide duration suggestion according to chosen questions.
    - Exam preference may be save by the user to shorten configuration time when the re-examine himself again.

## Submit admission request by non-admin teachers to be able to create questions

### Description and priority:

Should-have

### Stimulus/Response Sequences

When new teacher user registers in the application, he should be redirected to send teacher admission request.

This request should be sorted in database until one of the admin users respond.

### Functional Requirements

* + - All teacher’s user functionalities in this app should be blocked to that user until the admission request get a response.

## Submit a request by non-admin teachers to create new question

### Description and priority:

Should-have

### Stimulus/Response Sequences

When an accepted teacher user to create question submit a new one, a question request should be sent to admin users to respond with acception or rejection.

## Alerts

### Description and priority:

Could-have

### Stimulus/Response Sequences

When a user do a specific action such as creating a new question or subject, change profile settings and etc.

The response to this action should be an alter message to confirm user’s action.

## Feedback System

### Description and priority:

Could-have

### Functional Requirements

* + - Various entries should be provided to the user according to be able to understand user’s experience with this application such as feedback message, title, a scale to scale the ease of users manipulation with app, fairness of questions classification and etc.

## Save questions in favorites by student users

### Description and priority:

Could-have

## Emails

### Description and priority:

Would-have

### Stimulus/Response Sequences

When users hit a specific statistical number, such as

* + - When Teacher user don’t create questions for three days
    - When admission requests bucket is not empty and admin user don’t respond to any one for 12 hours
    - When student user don’t examine himself for 2 days
    - When any kind of user hit a statistical target that is set by the user himself.

Then email should be sent to that user.

## Direct chat

Would-have

### Functional Requirements

* + - When new message arrives, system should push a notification in notifications bucket

## Save Questions in drafts for question-creator users

Would-have

# 4. EXTERNAL INTERFACE REQUIREMENTS

## 4.1 USER INTERFACES:

The user interface of the software system shall be designed to be intuitive and easy to use, with a modern and responsive design. The user interface shall consist of a dashboard that displays the main functions of the system and a navigation bar for accessing different sections of the software. The dashboard shall include visualizations of important metrics and data, such as charts and graphs, to help users understand the system's performance. User interface in exam pages should be easy a pretty straight forward and not distracting, to make help users to put their full focus in the exam.

## 4.2 HARDWARE INTERFACES:

This website should be able to run on any smart modern device, including old poor specs ones.

## 4.3 SOFTWARE INTERFACES:

Following are the software used for the Exams Bank application.

|  |  |
| --- | --- |
| Software use | Description |
| Laravel | Laravel is used in this website as handling almost all of the business logic, APIs, authentication and authorization heavy duties in Laravel is pretty straight forwards. This website is mainly a backend project and Laravel is the best choice for this case.  Also Laravel is used to build server side rendered pages in only some specific pages that SEO is a must for it. |
| Vue | Vue is used to build the whole application after the user is signed in to take the advantage of robust JavaScript packages and it is used mainly to build exam SPA (single page application) to provide superb exam experience for the user even if there is internet connection interruption. |
| MySQL |  |

## 4.4 COMMUNICATION INTERFACES

The software system shall communicate with other systems and devices using HTTPS protocol, also it supports all types of web browsers.

# 5. OTHER NONFUNCTIONAL REQUIREMENTS

## 5.3 SECURITY REQUIREMENTS

Security requirements specify the measures that the software system must implement to protect against unauthorized access, data breaches, and other security threats. These requirements may include authentication, authorization, data encryption, and security auditing.

The system must also include intrusion detection and prevention mechanisms to protect against hacking attempts. The system must be tested and validated to ensure that it meets all security requirements.

## 5.4 SOFTWARE QUALITY ATTRIBUTES

* **MAINTAINABILITY:** The web application must support regular updates to ensure compatibility with the latest web technologies, Bug fixes, and security patches.
* **USABILITY:** The flight schedules should satisfy a maximum number of customers’ needs. The application should be accessible on multiple devices, including desktop and mobile.
* **PORTABILITY:** The web application must support offline access to specific pages to provide users with uninterrupted service in areas with limited or no internet connectivity. The offline pages should provide uninterrupted exam experience. This may be achieved in this application by using CSR rendering modern methods such as SPA.