**HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY**

**SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY**

-----🙞🙜🕮🙞🙜-----



**FINAL PROJECT REPORT**

**WEB TECHNOLOGY AND E-SERVICE**

**Builing a Online Examination Web-based System**

**Supervisor: PhD. Do Ba Lam**

**Group: 03**

**Member: Dao Minh Chi – 20200082**

**Hoang Huy Chien – 20200084**

*Hanoi, 6/2023*

CONTENTS

Table of Contents

[ACKNOWLEDGEMENT 3](#_Toc138989552)

[CHAPTER I : INRODUCTION 4](#_Toc138989553)

[1.1. Description 4](#_Toc138989554)

[1.2. Objectives 4](#_Toc138989555)

[1.3. Agents 4](#_Toc138989556)

[1.4. Report’s Organization 4](#_Toc138989557)

[CHAPTER II: SYSTEM DESIGN AND TECHNOLOGIES 6](#_Toc138989558)

[2.1. Use-case diagram 6](#_Toc138989559)

[2.2. Back-end Development 7](#_Toc138989560)

[2.3. Database Management 7](#_Toc138989561)

[2.4. Data Modeling 8](#_Toc138989562)

[2.2.1. User 8](#_Toc138989563)

[2.2.2. Question 9](#_Toc138989564)

[2.2.3. Contest 9](#_Toc138989565)

[2.5. User Authentication and Security 10](#_Toc138989566)

[2.6. Front-end Development 11](#_Toc138989567)

[2.7. Containers Structure 11](#_Toc138989568)

[CHAPTER III: FUNCTIONALITIES 13](#_Toc138989569)

[3.1. User Authentication and Integration 13](#_Toc138989570)

[3.2. Contest Creation and Question-Answer Management 13](#_Toc138989571)

[3.3. User Profile Editing 13](#_Toc138989572)

[3.4. Ranking and Performance Evaluation 14](#_Toc138989573)

[CHAPTER IV: RESULTS 15](#_Toc138989574)

[4.1. Overview 15](#_Toc138989575)

[4.2. Details 15](#_Toc138989576)

[CHAPTER V: MEMBER CONTRIBUTIONS 20](#_Toc138989577)

[5.1. Member list 20](#_Toc138989578)

[5.2. Task Assignment 20](#_Toc138989579)

[CHAPTER VI: CONCLUSION 21](#_Toc138989580)

[6.1. Difficulties 21](#_Toc138989581)

[6.2. Project Summarize 21](#_Toc138989582)

# **ACKNOWLEDGEMENT**

We are writing to express our sincere gratitude to all those who have contributed to the successful completion of our project.

First and foremost, we would like to thank our project supervisor, Mr. Do Ba Lam, for his invaluable guidance, support, and expertise. His extensive knowledge and guidance were essential in shaping our project and helping us overcome challenges. We are deeply grateful for his willingness to share his knowledge and expertise with us, and we are confident that we would not have been able to complete our project without his help.

We would also like to acknowledge the efforts and contributions of our team members, Dao Minh Chi, Hoang Huy Chien. Their dedication, hard work, and teamwork were essential in carrying out the project tasks, conducting experiments, and analyzing the results. We are grateful for their willingness to go above and beyond, and we are confident that our project would not have been as successful without their contributions.

# **CHAPTER I : INRODUCTION**

## Description

An Online examination system is an web-based appliction which provides secure, and time-limited exams or contests for users to participate in via a website. They can be created using a question bank and graded automatically. The system can be used to administer quizzes and tests, provide practice exams, and conduct research on student learning.

## Objectives

The primary objective of our project was to create an efficient and user-friendly system that simplifies attendance checking and exam administration for instructors while providing students with easy access to their attendance records and exam results. By addressing these challenges, we aimed to enhance the overall educational experience.

## Agents

For this project, we have developed an application for users to both create their own contests and participate in others’ (like Quizziz). Thus, it might not be have the admin side for the system. Everyone is admin of their quizzes and user of other posted examinations.

## Report’s Organization

This is final section of Chapter I: Introduction. Henceforth, the report will be structured into five other main chapters, each detailing a significant aspect of our online examination project.

Chapter 2: System design and technologies

In this chapter, we descript about the design of our application system. We discuss the Back-end Development, Database Management, User Authentication and Security, and Front-end Development. The technologies used in our project will also be outlined to provide a technical insight into how we created our application.

Chapter 3: Functionalities

This chapter provides an overview of the functionalities of the system which are designed to be user-friendly and efficient. It includes user authentication, profile editting as well as contests and questions creation.

Chapter 4: Results

In this chapter, we present our results of the project. Demonstratrion of the functionalities we have implemented and discuss. Screenshots of our application will also be shown there.

Chapter 5: Member Contributions

This chapter illustrates the contributions of each team member towards the project. We clarify who was responsible for each part of the system.

Chapter 6: Conclusion

This chapter outlines some difficulties we face while working on the project and concludes our report. It summarizes our project and reflect on our learning experiences during the project development.

# **CHAPTER II: SYSTEM DESIGN AND TECHNOLOGIES**

## Use-case diagram

A diagram of a person

Description automatically generated with low confidence

## Back-end Development

The back-end development of your project uses Node.js and Express.js. Node.js is a JavaScript runtime environment that executes JavaScript code outside of a web browser. This makes it ideal for developing back-end servers, as it allows you to use the same programming language for both the front-end and back-end of your application. Express.js is a web framework built on top of Node.js that provides a number of features (e.g. handle routing, middleware, and templates) that make it easy to develop and deploy back-end servers.

Together, Node.js and Express.js can be used to create efficient and scalable back-end servers. Node.js is event-driven, which means that it can handle a large number of concurrent requests without becoming bogged down. Express.js provides a number of features that make it easy to route requests, process data, and generate responses.

## Database Management

The database management techniques used in this project are:

* MongoDB: MongoDB is a NoSQL database management system that is designed to store large amounts of data efficiently. It is a document-oriented database, which means that data is stored in documents that are similar to JSON objects. This makes it easy to store and query data, as well as to scale the database as needed.
* Mongoose: Mongoose is an Object Data Modeling (ODM) library that is used to define and interact with MongoDB databases. It provides a number of features that make it easy to work with MongoDB, such as schema validation, ORM, and query builders.

Mongoose is used to define the schema for various collections (e.g. attendance records, exam results, etc.). The schema defines the structure of the documents that are stored efficiently in the collections. This makes it easy to work with the data in the collections, as well as to ensure that the data is always consistent. Mongoose is also used to query the data in the collections. This makes it easy to find specific records, or to perform aggregations on the data. The database is scalable and can be easily modified to meet the needs of the project.

Overall, the use of MongoDB and Mongoose makes it possible to develop efficient, scalable, and secure applications that store and manage large amounts of data.

## Data Modeling

There are three main entities models were designed for our developing our system: User, Question, Contest in detail. Following is the brief description of each:

### 2.2.1. User

Each User document may contain fields such as:

* name: represents user’s registered name. Required for attendance and result records.
* email: represents user's email address which needs to be in validated format. Required and unique for login.
* password: this field stores user’s password. Required for login and authentication purposes.
* avatar: stores user’s upload profile picture in the format of soucre link.
* dob: stores user's date of birth.
* phoneNumber: stores user's phone number.
* urlFacebook, urlYoutube, urlWebsite: represent URLs to user’s profile on some online platforms.

Except for first three fields that is required, user does not need to provide these others when registering.

### 2.2.2. Question

Each Question document may contain fields such as:

* level: represent the difficulty of the question. Which is EASY, MEDIUM or HARD.
* title: this field stores overview title of the question.
* description: stores the question in detail.
* explain: optional field to store correct answers explanations.
* answers: which is a list of nested objects contains these following field:
  + answerId: id of the answer
  + position: represents appearing order of this answer
  + content: the visible content of the answer
  + isCorrect: boolean field stores if this answer is correct or not
* groupQuestion: stores the question bank that this question belong to. Reference to groupQuestion entity.

### 2.2.3. Contest

Each Contest document may contain fields such as:

* title: the title of the contest.
* description: stores overview description of a contest.
* imageUrl: stores the source URL of the background image for a contest shown on home page.
* createdBy: refence to an User entity which creates the contest.
* startTime: this field represents the time which is the contest opens participating access. Format in Date.
* endTime: the deadline time that the contest closes.
* examTime: amount of time for participants do the exam. Store in minutes.
* amountQuestion: number of questions on the exam.
* groupQuestion: refer to an question bank entity that the contest imports.
* isActive: the boolean field tells if the contest is closed or not.
* password: stores the code for participants to start the exam.

## User Authentication and Security

User authentication and security are essential for any web application. In this project, the following techniques are used to authenticate users and protect the application from unauthorized access:

* JSON Web Token (JWT): JWT is a standard for creating and verifying JSON-encoded access tokens. These tokens are used to verify the identity of users when they make requests from client-side resources.
* bcryptjs: bcryptjs is a password hashing library that is used to hash and salt passwords before storing them in the database. This helps to protect passwords from being guessed or cracked.

The use of JWT and bcryptjs makes it possible to authenticate users securely and protect the application from unauthorized access.

The reason for us to choose these technologies is highlighted as follows:

* JWT is a lightweight and secure token format: JWT tokens are small and easy to transmit, which makes them ideal for use in web applications. They are also secure, as they are signed using a secret key.
* bcryptjs is a robust password hashing library: bcryptjs uses a slow hashing algorithm that makes it difficult to crack passwords. This helps to protect passwords from being guessed or stolen.
* The combination of JWT and bcryptjs provides a high level of security: The combination of JWT and bcryptjs provides a high level of security for user authentication and authorization. JWT tokens are used to verify the identity of users, while bcryptjs is used to hash and salt passwords. This helps to protect passwords from being guessed or cracked, and it also makes it difficult for unauthorized users to access the application.

Overall, the use of JWT and bcryptjs makes it possible to develop secure web applications that protect user data from unauthorized access.

## Front-end Development

The front-end development of your project uses the following technologies:

* HTML5
* CSS3
* Bootstrap
* JavaScript/jQuery
* AJAX

The use of these technologies makes it possible to develop visually appealing and interactive web pages that are responsive and efficient. Since:

* HTML5 is a powerful markup language: HTML5 provides a number of new features that make it possible to create more complex and interactive web pages.
* CSS3 is a versatile style sheet language: CSS3 provides a number of new features that make it possible to control the visual design and layout of web pages with more precision.
* Bootstrap is a popular toolkit: Bootstrap is a popular toolkit that provides a number of features that make it easy to develop responsive web pages.
* JavaScript/jQuery are powerful programming languages: JavaScript and jQuery are powerful programming languages that can be used to add interactivity and functionality to web pages.
* AJAX is a efficient technique: AJAX is an efficient technique that can be used to retrieve data from the server and database without reloading or refreshing the entire page content.

## 2.7. Containers Structure

In this project, we have developed various containers for front-end designing. We would like to discuss about the detail of some main containers as follows:

* **Login**: The login page that allows users to log in the system if have they have an registed account or immigration from other platforms.
* **Register**: The page for creating a new account to log in our system.
* **Home**: The first page a user sees after logging in. It provides an overview of the platform and its features, making it easier for users to navigate and use the application.
* **Contest**: The contest management screen. From there, users can create their own contest as well as view, update, or delete existed ones. It also provides functionalities to search contest and view the stastistic of attendance and results of contestees.
* **GroupQuestion**: The page for managing user’s question banks. It allows users to create, view, update or delete question bank like the Contest screen above.
* **Question**: Inside the question bank screen which allows users to manage each question in the same group.
* **ExamTest**: The page shows when an user currently doing a exam on a partcipated contest. It allows users to view the question of the exam, choose and save the answers, changing questions and submittion. It also shows the remaining time for the exam as well.
* **ExamDetail**: This page is the page shown statistic of an contest mentioned above. Furthermore, it also provides functionalities to view contestees’ history and the contest’s overview information before starting it.
* **User**: The user profile page. From there, an user can edit his/her information and change account’s password.

# **CHAPTER III: FUNCTIONALITIES**

As the potential of the technologies and our system designs mentioned in the previous chapter, the Online Examination Web-based System is expected to offer the following functionalities:

## User Authentication and Integration

* Standard Login: For standard login, user can access to the system through a registered account by using email and password. The email had already validated when registering (just checking format, not yet verification).
* Login with Google or Facebook: In addition to the standard login functionality, we have integrated login options using Google and Facebook accounts, providing users with a convenient and secure authentication method.

## 3.2. Contest Creation and Question-Answer Management

* Contest Creation: Instructors can create contests, allowing them to design customized exams with specific parameters such as time limits, question types, and difficulty levels.
* Question-Answer Management: The system provides an interface for instructors to add, edit, and manage questions and answers for contests. This allows for flexibility in creating comprehensive and tailored exams.

## 3.3. User Profile Editing

* Edit User Information: Users can modify their personal information, such as their name, email address, and profile picture. This functionality enables individuals to keep their profiles up-to-date and accurate.
* Change password: Users can change their password for sercurity purposes or stuffs. They need to provide the old password to do this.

## 3.4. Ranking and Performance Evaluation

* Participant Ranking: The system tracks and ranks participants based on their exam scores. Users can view their rankings in comparison to others, fostering a sense of healthy competition and motivation.
* Contest Summary and Evaluation: The system generates comprehensive summaries and evaluations for contest creators. This feature provides valuable insights into the overall performance of participants, facilitating the assessment and improvement of exams.

# **CHAPTER IV: RESULTS**

## Overview

We are pleased to report that our project has been successfully completed, and we have obtained remarkable results. Throughout the development process, we transformed our initial plans and designs into a fully functional and robust system that fulfills the requirements and expectations of our users.

The back-end server was built using Node.js and Express.js, providing a fast and scalable runtime environment for our application. MongoDB, coupled with Mongoose, efficiently manages attendance records and exam results, ensuring seamless data retrieval and manipulation. User authentication is secure through the implementation of JSON Web Token (JWT) and bcryptjs for password hashing and salting.

On the front-end side, we utilized HTML5, CSS3, and Bootstrap to create a visually appealing and responsive user interface. JavaScript/jQuery enhanced the user experience by implementing interactive features and form validation, while AJAX facilitated efficient data retrieval without the need for page reloads.

## Details

On the login page (Figure 1), there are options for stardard login or immigration login as well as register option to create a new account. Only registed account can log in to the system, if an unregisted email try to login or wrong password typing, the system will assert an error message “User is not found” or “Wrong password”.

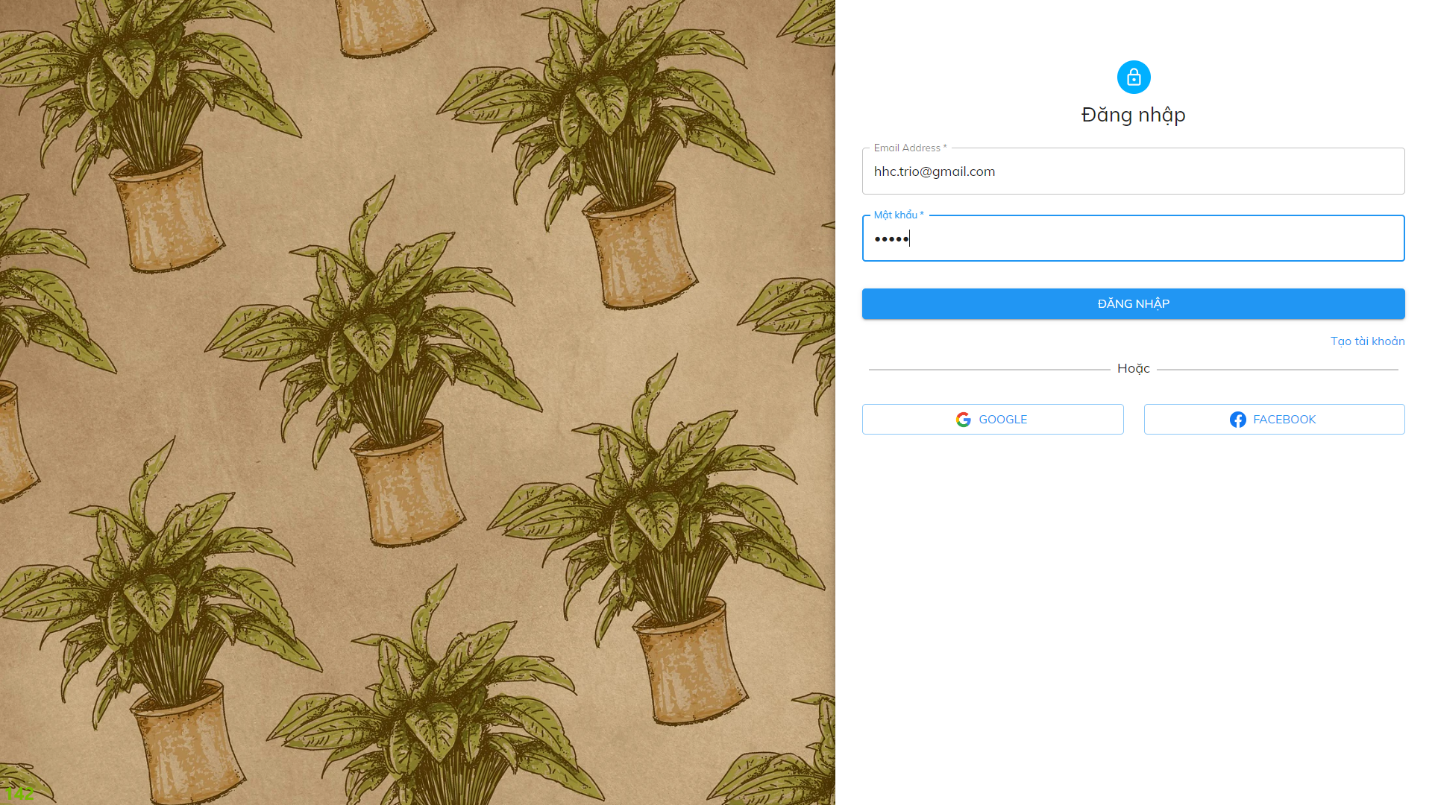


Figure . Login page

A screenshot of a computer

Description automatically generated

Figure . Home page

Home page (Figure 2) shows all the published contest from users which specifies contest name, create by whom, created date, time, etc. for each. The system also provides filter to show only currently active contest, closed contest or contest that user had participated in, …

A screenshot of a computer

Description automatically generated

Figure . Contest management page

Users can view their own created contest from contest management page (Figure 3) including both published and private ones. For each contest, they can see code to join it by hovering through the lock icon in case they had forgotten.

A screenshot of a computer

Description automatically generated

Figure . Question management page

The question management page inside each group of questions is shown as Figure 4. From this, user can both add questions manually and import the questions file in the right format in order to the system to correctly read (the format can be viewed by clicking to the information icon (i) by the left to upload button). After add or import, each question can be update by a form appeared when user clicks to the question.

A screenshot of a computer

Description automatically generated with medium confidence

Figure . User profile page

User profile page (Figure 5) allows users to change their information like user name, phone number, etc. Email can not be changed since it is unique to each account. They also can change profile picture by clicking to it, the system will open browser for user to upload their image.

# **CHAPTER V: MEMBER CONTRIBUTIONS**

## 5.1. Member list

|  |  |  |
| --- | --- | --- |
| **No.** | **Full Name** | **Student ID** |
| 1 | Dao Minh Chi | 20200082 |
| 2 | Hoang Huy Chien | 20200084 |

## 5.2. Task Assignment

The contributions of each group member is shown as the following table:

|  |  |  |
| --- | --- | --- |
| **No.** | **Task** | **Contributors** |
| 1 | Login and Register Implementation | Dao Minh Chi |
| 2 | Immigration and Authenication | Dao Minh Chi |
| 3 | Multiple choice examination design | Dao Minh Chi, Hoang Huy Chien |
| 4 | Home Page | Dao Minh Chi |
| 5 | User Profile | Hoang Huy Chien |
| 6 | Database design | Dao Minh Chi |
| 7 | Contest Management and Exam details | Dao Minh Chi, Hoang Huy Chien |
| 8 | Question and group management | Hoang Huy Chien |
| 9 | Back-end logic | Dao Minh Chi |
| 10 | Report and Presentation slide | Dao Minh Chi, Hoang Huy Chien |

# **CHAPTER VI: CONCLUSION**

## 6.1. Difficulties

During the development process, we encountered several challenges that required innovative solutions. Designing a user-friendly interface that was both intuitive and visually appealing proved to be a major difficulty. Additionally, ensuring optimal performance and scalability to handle a large volume of data and traffic presented significant challenges. However, through collaboration, problem-solving, and persistence, we overcame these obstacles successfully.

## 6.2. Project Summarize

The Web-based System for Attendance and Multiple-Choice Exams project stands as a testament to our team's skills and expertise in software development. Through the implementation of Node.js, Express.js, MongoDB, and various front-end technologies, we have created a robust, efficient, and user-friendly system.

Our project addresses the challenges associated with attendance checking and exam administration, providing instructors and students with a valuable solution. We are confident that our system will greatly contribute to enhancing the educational experience for all stakeholders involved.

We invite you to explore our project further on GitHub, where you can find the complete source code, documentation, and additional details: <https://github.com/Chisskj/A-System-For-Web-Subject>

In conclusion, we have successfully achieved our objectives and are excited about the future of the System for Attendance and Multiple-Choice Exams. We remain dedicated to delivering a high-quality system and will continue to refine and expand it based on user feedback and technological advancements.