Anh Nguyen

CREATING A WEBSITE CONNECTING STUDENTS AND EDUCATION SYSTEMS OVER THE WORLD

Bachelor's thesis

Information Technology

Bachelor of Engineering

2023

XAMK
South-Eastern Finland
University of Applied Sciences

Degree title	Bachelor of Engineering
Author(s)	Anh Nguyen
Thesis title	Creating a website connecting students and education systems over the world
Commissioned by	_
Year	2023
Pages	42 pages, 3 pages of appendices
Supervisor(s)	Matti Juutilainen

ABSTRACT

When people search for information on studying abroad, the sheer volume of available information can be overwhelming. In addition, relying on popular search engines such as Google can often result in a limited pool of results because they tend to prioritize the most commonly accessed.

The goal of the thesis was to create a website just for the education community where students could choose the best school for their needs and schools can also use student profiles as one of their assessment criteria. This will help students save time and not miss information or choices. Furthermore, school systems can rely on this as a platform website for introducing and promoting their school to students.

This thesis described the basics of Node.js, React.js, HTML, CSS, and MongoDB. Moreover, the thesis illustrated how each component of the website was implemented.

In the end, there were some ideas to improve the website so that it would be better in the future.

Keywords: Nodejs , MongoDB, website, user.

CONTENTS

1 INTRODUCTION

Social media has evolved over the years since the creation of the first social networking site, called Six Degrees, in 1997. The following year, AOL Instant Messenger was introduced, and it was no longer just a platform for personal communication but a way to meet new people online. Over the years, social media networks have developed in various ways, encompassing photo sharing, moment sharing, status updates, video sharing, and image sharing, as demonstrated by major platforms like Facebook, YouTube, and TikTok,etc.

When people started searching for information about studying abroad on the internet, they were confused because there was so much information. Sometimes Google only gives the most selected results if people search "study abroad in country A". Based on the widespread development of social networks and to support students in making comprehensive and appropriate choices for themselves, the aim of this thesis is to create a website connecting students and education systems around the world.

The website aims to solve one of the most prominent issues faced by students: finding an education system that suits their needs. The site offers a platform where students can connect with education systems from around the world, allowing them to compare factors such as curriculum, fees, and scholarships and to know the enrollment information of the schools. The schools can also access links such as the LinkedIn URL to view student information. The target users of my website are those who are interested in studying abroad and exploring different education systems. The user will have a personal page that posts the user’s status and activities, and other users' feelings about the user.

Full-stack technology will be used to build the website. The Javascript, HTML, and CSS languages will be used throughout the website development process. Mongo DB will serve as the database in the backend.

The thesis is divided into five chapters: introduction, theory background, implementation, future improvement, and conclusion. The chapter introduction contains the background of the social media network, the reason for this topic, an overview of the problem, and the solution. Theory chapter includes the definitions of programming languages and tools used in the project. Implementation chapter digs into the website's construction. There are some suggestions for improving the website in the future development chapter. In the last chapter, there is a summary of what is mentioned in the thesis project and a brief description of the result of the project.

2 THEORY BACKGROUND

The current challenge of searching for information about schools is considerable. Nowadays, people often face a series of difficulties when conducting searches on the website, through groups, or via the study aboard counseling center.

On the website, finding information about schools can be confusing due to the huge and varied amount of information. This can lead the searcher to get lost and unsure about which website to rely on for their research. In addition, the presence of numerous school ranking websites further complicates matters since each website provides different results, leading to confusion and difficulty for searchers.

Similarly, finding information through groups poses some challenges. In groups, people often have general or conflicting opinions about schools. Moreover, each individual has limited experience with a specific number of schools, resulting in opinions and information that are limited in scope and not comprehensive. This can result in inefficiency and biased assessments of specific schools.

Turning to the study aboard counseling centers, they often recommend schools with high acceptance rates without considering the personal criteria of the customers. This can distract from the focus on quality and the important factors that customers are concerned about. Centers frequently prioritize providing the best possible success rates without ensuring that the customer's choice is appropriate and beneficial in the long run because the higher the percentage of customers who pass into a school, the more customers their center has.

Furthermore, in the process of self-searching for information to apply to study abroad, students often access the school's website to learn about the necessary criteria and application timelines. However, another difficulty they face is that the websites of different schools have different designs and structures, with some being complex, making it difficult for students to find the information they need. This can leave students feeling frustrated and drained of energy during the search process, potentially causing them to overlook suitable institutions due to minor issues.

From the above challenges and difficulties, the idea of this website was born to assist students in searching for comprehensive information about schools. It also helps all schools increase their visibility and recognition.

The target users of the thesis website are those who are interested in studying abroad and exploring different education systems. The website offers a one-stop solution for students searching for suitable education systems. With its user-friendly interface and seamless technical components, students can explore educational opportunities without hindrance.

To create a seamless user experience, the website was designed to mimic the interface of popular social networks like Facebook and LinkedIn. With a user page, home page, privacy page, and more, the website offers a user-friendly interface that is easy to navigate. Since the website is a combination of front-end and back-end components, both were designed keeping this goal in mind.

The front end of the website is the presentation layer that users interact with directly. This layer involves creating the user interface and designing the layout of the web page. Additionally, the front-end framework used for this project was React. HTML and CSS languages also played an essential role in creating the user interface for the website.

The back-end of the website is responsible for connecting to the database, storing data, and transferring suitable data to the front-end to display. Node.js, a popular server-side runtime environment, was chosen for the back-end because of its lightweight and scalable nature. Furthermore, MongoDB was chosen as the database for this thesis project due to its flexibility and capability to store complex data.

2.1 Node.js

Node.js is a widely adopted back-end JavaScript runtime environment that is known for its versatility, scalability, and performance. It is an open-source platform that enables developers to build server-side applications efficiently. Introduced in 2009 by Ryan Dahl, Node.js has gained significant popularity due to its efficient handling of multiple connections, well-designed API, and robust framework for developing high-performance applications. One of the key factors contributing to its success is the use of the V8 JavaScript engine, which is also utilized in leading web browsers such as Google Chrome. (Semah, 2022.)

In a Node application, execution occurs within a single process, differentiating it from traditional server-side programs. Unlike the conventional approach of starting a new thread for each request, Node handles multiple concurrent connections without facing the challenges of thread concurrency or the overhead associated with multithreading. Node.js operates asynchronously, employing an event loop mechanism. Incoming requests are processed sequentially using an event queue, allowing Node to handle small requests individually without waiting for responses. (Sheldon & Denman, 2022.)

There are some advantages of using node.js. One of the key features of Nodejs is its event-driven architecture, which allows for a non-blocking I/O model. This means that Nodejs can handle multiple simultaneous requests without getting bogged down by any single one of them. This makes it ideal for building web applications that require a lot of I/O operations such as real-time chat applications or online gaming platforms. Node.js offers both client-side and server-side. Nodejs also provides access to a vast array of libraries, known as modules, that makes it easier for developers to build and scale their applications. These modules encompass everything from database connectivity to front-end frameworks like React or Vue. Additionally, Nodejs has a vibrant community that creates modules, shares knowledge, and actively contributes to the development of Nodejs itself. Applications created with Node.js can easily run on major platforms including Windows, Linux and macOS. (Top 5 NodeJS Benefits and Top 5 Limitations: What They Mean for Your Business, 2023.)

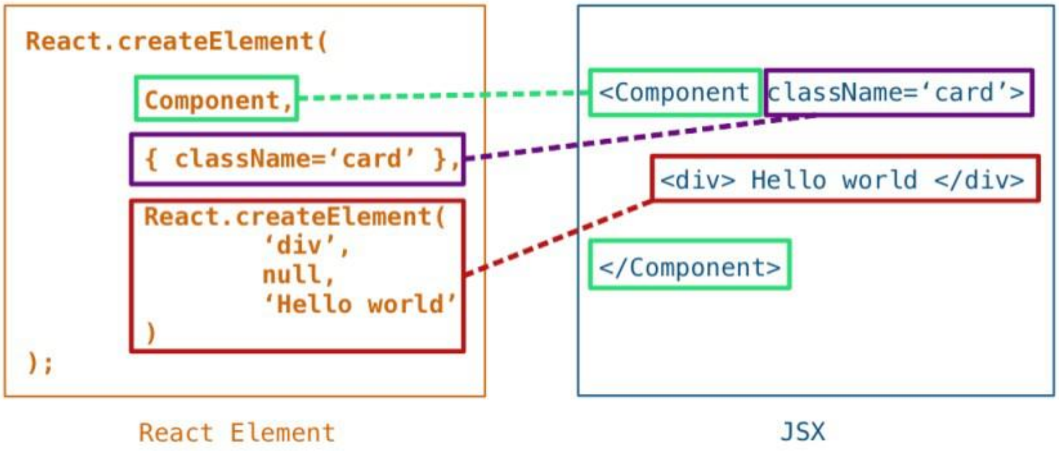
Aside from that, Node.js also has some drawbacks. Node.js is unable to handle heavy CPU-bound tasks, and if we nested callbacks in one another, there might be a callback hell. (Top 5 NodeJS Benefits and Top 5 Limitations: What They Mean for Your Business, 2023.)

2.2 React.js

React.js was invented by Jordan Walke , an employee of Meta organization , in 2013. React gained significant attraction in 2014, when key new features, such as the React Developer Tools, were launched as an extension of the Chrome Developer Tools. React is presently used by several well-known businesses, including Netflix, Facebook, Instagram, Airbnb, Reddit,and Dropbox. (Arancio, 2021.)

React.js is a free, open-source Javascript library that is used for building UI components and user interfaces.React is considered as the view part of the MVC pattern. React is built around components, not templates like other frameworks. Components are self-defined, reusable, and completely separate interface (UI) components.Props and states are two important properties of a React component. State represents the component's "state"; state is private and can only be changed within the component. Props are external and not controlled by the component itself. It is passed from the higher component hierarchically, or simply from the parent component to the child component. (Jordana, 2023.)

Moreover, to speed up the development process and reduce the risks that may occur while coding, React also provides the ability to reuse code by introducing two important concepts, including JSX and virtual DOM:



```
React.createElement(
        Component,
        { className='card' },
        React.createElement(
                'div',
                null,
                'Hello world'
        )
);
```

```
<Component className='card'>
    <div> Hello world </div>
</Component>
```

Figure 1. React without JSX and with JSX (Decpk, 2021.)

- JSX (JavaScript Extension): is a combination of Javascript and XML. It is a React extension that makes it easy to change the DOM tree with simple HTML-style code. Elements are created with JSX, and then JavaScript is used to render them on the site. We can use React without JSX, but using JSX gives us a lot of benefits, like using the formatting library, rendering lists of data, and keeping markup clean and organized. (Verma, 2020.)

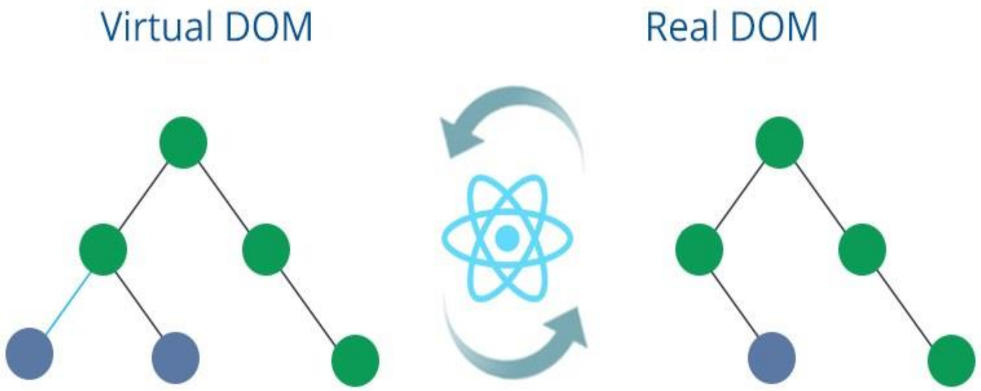


Figure 2. Comparison of how Virtual Dom and Real Dom work (What is virtual dom and its benefits that you must know, n.d.)

- Virtual DOM: is a replica of the actual DOM present on a web page, and ReactJS leverages it to identify the specific portion of the real DOM that requires updates when an event triggers a change in its associated element. For instance, when a user enters a comment in the comment box of a blog post on a website, ReactJS utilizes the Virtual DOM to efficiently determine and update the relevant part of the actual DOM. The user should be able to see their comment added to the list of comments. In the absence of ReactJS, the entire DOM tree would have to be updated to signal this new change. When using React, it will help to scan through the virtual DOM to see what has changed after the user performed the above action and select the right place to update the change. (Mojeed, 2022.)

In short, using Reactjs can help us support building interfaces quickly, limit errors in the coding process, and reuse components.

2.3 HTML && CSS

To work effectively with React.js and its JSX syntax, a strong foundation in HTML is essential, as JSX closely follows the structure of HTML. However, understanding the fundamentals of CSS is equally crucial, as it allows for the styling of web pages, enhancing their appearance and visual appeal.

In 1991, Tim Berners-Lee introduced HTML (Hypertext Markup Language), a programming language specifically designed for structuring and organizing the various components within a website. HTML serves as the backbone of web page development, enabling effective organization and structuring of elements. It provides the essential framework and layout necessary for creating well-designed web pages. Additionally, HTML supports the declaration and embedding of digital files, including music, videos, images, and more. (Hameed, 2023.)

HTML has the advantage of having a large number of support resources, working smoothly on almost any browser today, and the markup used in HTML is usually short,has high consistency, and easy to integrate with other backend languages.

Besides, the disadvantage is that it only applies to static web; each HTML page needs to be generated separately, even if there are many duplicate elements, such as a header and footer. (Astari, 2023.)

CSS was developed by the W3C (World Wide Web Consortium) in 1996 for one simple reason: HTML is not designed for tagging to help with web page formatting. It can only be used to "bookmark" the site. CSS is a language for styling web pages; it is also known as Cascading Style Sheets. It is used to style elements written in a markup language, such as HTML. It can control the formatting of multiple web pages at the same time to save effort for web writers. It separates the appearance of the web page from the main content of the page by controlling the layout, colors, and fonts. (Brian, 2023.)

CSS has three methods of embedding CSS into the website: internal, external, and inline CSS styles:

- The CSS internal style is the style that is uploaded every time the web page is refreshed. So it increases page load time. Developers won't use the same CSS style for multiple pages because it only applies to each page. However, the benefit of the CSS internal style is that when everything is placed on one page, it is easier to share the page for preview.
- The method of using Style External is the most convenient. Everything is saved in a ".css" file. This means you can style it in a ".css" file, then apply CSS to the web page file you want. External styling will greatly improve page load time.
- Inline CSS works with certain elements that have a <style> tag. Every element needs to be styled, so it's not necessarily the best and easiest way to deal with CSS. However, it can be quite handy because if we want to change just one element, we can quickly preview the change. We don't need to go directly to the CSS file to edit, but instead use inline CSS.

(Domantas, 2023.)

CSS can be written once and then reused in subsequent HTML pages. This will help save time. On the other hand, CSS will behave differently for each specific browser. For instance, IE and Opera support CSS in different ways. (Vatsal, 2022.)

2.4 MongoDB

MongoDB was created by Dwight Merriman, Eliot Horowitz and Kevin Ryan in 2007. MongoDB is a document-oriented database, a form of NoSQL database. Therefore, MongoDB will avoid the table-based structure of the relational database in order to accommodate documents like JSON that have a very flexible schema called BSON. MongoDB uses data storage in the form of Document JSON, so each collection will have different sizes and documents. The data is stored in JSON-style documents so queries will be very fast. MongoDB has become one of the most prominent NoSQL databases over time, used as a backend for many websites such as eBay, SourceForge and The New York Times. (Mongodb History And Advantages, n.d.)

MongoDB has the following features:

- Ad hoc queries: support field searches, regular searches, regular expression searches, and range queries.
- Indexing: any field in a BSON document can be indexed.
- Replication: MongoDB provides a replication mechanism, also known as "cloning", which ensures the availability and integrity of data. By creating identical copies of the database, replication safeguards against data loss and provides fault tolerance.
- Aggregation: Aggregation operations process and calculate results from data records. Grouping values from multiple documents allows for performing various operations on the grouped data and returning a single result. Similar to SQL's count(*) and GROUP BY, MongoDB's aggregation provides similar functionality.
- File Storage: MongoDB is used as a file system that takes advantage of the above functions and acts as a way of delivering data over sharding.

(What is MongoDB – Working and Features, 2021.)

MongoDB offers numerous benefits to users. Those benefits are unstructured data storage, no constraints, integrity, high availability, great performance, and storage scalability. Data is cached (written into a buffer) on RAM with limited access to the hard drive, resulting in fast read and write speeds. (Gohel, 2022.)

Despite its benefits, MongoDB also has some drawbacks. It is not applicable to transaction models that require high accuracy due to no constraints. Data takes RAM as the focus of the operation, so when the operation requires a large RAM memory. Because any changes to the default data are not immediately written to the hard drive, the possibility of data loss from an unexpected power failure is very high. (Gohel, 2022.)

3 IMPLEMENTATION

This chapter will demonstrate how the website was developed, and how the backend and frontend were linked.

3.1 Web site overview

The website was designed to connect members of the education community, such as students and teachers. Users can express themselves on the platform by posting status or giving their opinions to other users. Furthermore, the website offers another service for schools to post admissions, and students can use the search engine to find and apply to schools that are suitable with them.

My website is called EduConnected, and these are its functions:

- Sign up and log in: In order to engage with others, users must first create an account and then log in with that account.
- Post and manage status: users will be able to post status regarding things like obtaining a scholarship or finishing an ABC course, as well as modify or delete status. Users can like or comment on others' postings at the same time.

- Add or edit "about" information: Users will be able to create and edit educational information about themselves.
- Express an opinion: users will be able to express their thoughts on someone.
- School search: Users will be able to utilize the school search engine to find their ideal school.
- User search: search for another user.
- Follow other users: Follow other users to stay up-to-date on their latest updates.

In addition, the main color of the website is green because this color symbolizes nature, which is very similar to humans, and it will create a sense of closeness for website visitors. Figure 3 displays the website’s logo.

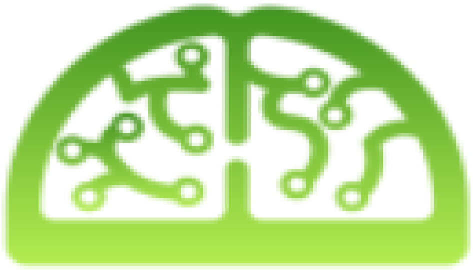


Figure 3. Website logo

This logo contains both the letters E and C. It also looks like the brain, which means that the brain where the students' knowledge is located will be well developed if they can choose for themselves a school that is suitable for them in their favorite country.

3.2 API planning

APIs are methods and protocols that connect to other libraries and applications. It stands for Application Programming Interface. The API enables access to a set of regularly used functions. Thus, it is possible to exchange data between applications. The API works in 4 different ways: SOAP, RPC, Websocket and REST. However, API REST was used in this project. (Dorairajan, 2022.)

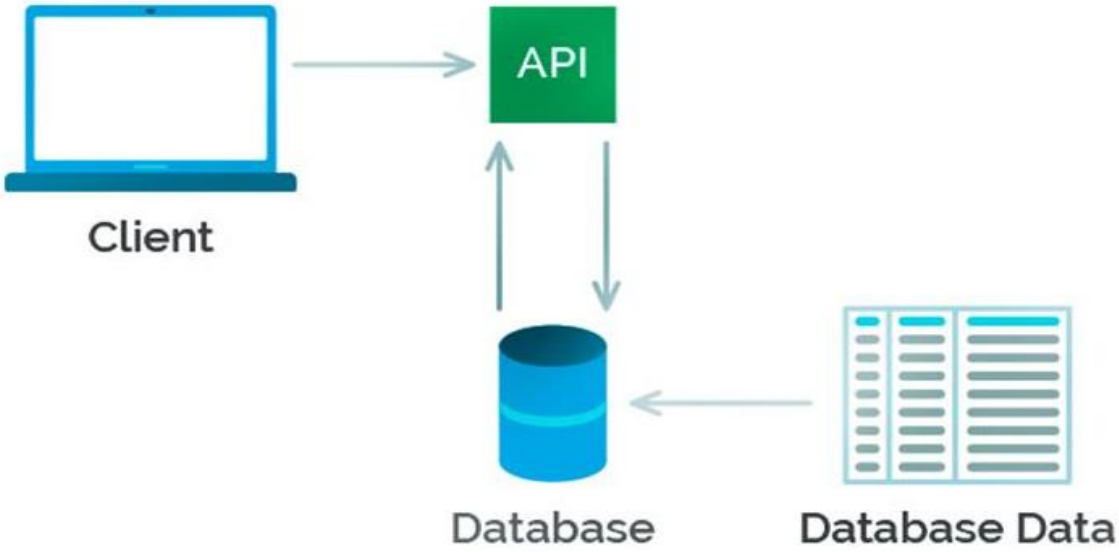


Figure 4. REST API Design (What is Rest API? Learn how Restful API works, 2021.)

Firgure 4 shows the REST API Design. REST which is also known as Representational State Transfer is a form of data structure transformation, an architectural style for writing APIs.It facilitates machine-to-machine communication by utilizing simple HTTP techniques.Instead of using a URL to handle some user information, REST sends an HTTP request to a URL such as GET, POST, DELETE, or PUT,etc. to process the data. (Gupta, 2022.). Four main methods were used in this project:

- GET (SELECT): Returns a Resource or a list of Resources.
- POST (CREATE): Create a new Resource.
- PUT (UPDATE): Update information for Resource.
- DELETE (DELETE): Delete a Resource.

These methods or operations are often called CRUD corresponding to Create, Read, Update, Delete.

Benefits of a restful API include the ability to organize complex applications and easy to use resources, high load management due to HTTP proxy server and cache, providing flexible formats by serializing data in XML or JSON form, allowing the use of standard HTTP procedure calls to retrieve data and requests. (Erinc, 2020.)

This is the API plan for the project:

- API of the schools:

Method	Endpoint	Action
GET	/schools	Get the data of all schools
GET	/schools	Find school by ID and return school data
POST	/schools	Add new school to the resource
POST	/schools/find	Find schools that fit the criteria. The criteria content inside the request body.
PUT	/schools/:id	Update school by id. Id is sent in the route parameter
DELETE	/schools/:id	Delete school by id. Id is sent in the route parameter

- API for the users:

Method	Endpoint	Authenticate	Action
POST	/signup	No	Create new account for user
POST	/login	No	Create log in route , which creates token for other actions
GET	/user	Yes	Fetch the user's information by getting Id from the user that "authenticated" from JWT.
PUT	/user	Yes	Update user by id. Id from the user that "authenticated" from JWT.

PUT	/user/fullname	Yes	Edit full name of user
PUT	/user/email	Yes	Edit email of user
PUT	/user/username	Yes	Edit username of user
PUT	/changepassword	Yes	Edit password
PUT	/user/status	Yes	Add new status of user. New status content inside the request body.
PUT	/user/status/:id	Yes	Change status content. Id is sent in the route parameter. Status content inside the request body.
PUT	/user/deletestatus/:id	Yes	Delete user's status by id. Id is sent in the route parameter.
PUT	/user/about	Yes	Edit about content
PUT	/user/reactionlike/:id	Yes	Add or delete like reaction. Id is sent in the route parameter.
PUT	/user/reactioncomment/:id	Yes	Add comment reaction
PUT	/user/deleteopinion/:id	Yes	Deleted opinion which is written for user
GET	/user/:username	Yes	Fetch other users information by username. Username is sent in the route parameter.
PUT	/user/addopinion/:username	Yes	Fetch other users information by username. Username is sent in the route parameter.
PUT	/user/addopinion/:username	Yes	Add opinion for other users by username. Username is sent in the route parameter.
PUT	/otheruser/deleteopinion/:id	Yes	Delete opinion that is written by user for another user. Id of the other users is sent in the route parameter.

PUT	/otheruser/follow/:id	Yes	Add follow or delete follow between user. Id of the other users is sent in the route parameter.
GET	/statusCheck	No	Fetch the status that is created in 10 minutes of all users.
GET	/logout	Yes	Log the user out and delete the token.

The API tables above provide a range of endpoints and methods that cover the necessary functionalities for the project. The API demonstrates clear endpoints, methods, and descriptions, which should facilitate integration into the project.

3.3 Server implementation

To develop the backend , some modules / frameworks need to be installed: express , body-parser , cookie-parser, cors, bcryptjs, jsonwebtoken, mongoose. The command to install these modules is: npm install + the name of modules + --save.

Express is a web application framework for nodejs. Express supports HTTP methods and middleware creating a very powerful and easy to use API. The Express framework's features comprise the ability to create intermediate classes to return HTTP requests; the ability to utilize routing definitions with varied actions based on HTTP method and URL; and the ability to return HTML pages based on the parameters supplied to the template. (Sharma, 2023.)

The body-parser module in Node.js is a middleware for handling HTTP POST requests. JSON, raw data, text, and URL encoding are all supported by the body-parser module. (Gouravhammad, 2023.)

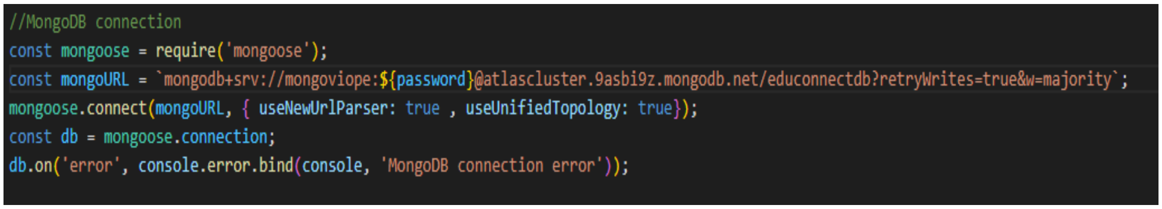
The cookie-parser module is a middleware for handling HTTP cookies in a Node.js application. Cookies are small pieces of data that can be sent from a server to a client and stored on the client's device.Once the cookie-parser middleware is added to the request handling pipeline, it will automatically parse incoming cookie data and make it available to the application's route handlers. It also provides a function for setting cookies in the HTTP response headers of outgoing requests. (Malhotra, 2021.)

Cors is an acronym that stands for cross-resource sharing. For example, when the client calls an API from another source whose domain is different from the current page (the origin source). Cors is a very useful package that provides middleware that allows us to enable Cors with many options to customize and concisely express. (Singhal, 2023.)

Bcryptjs is a javascript module. It is a type of hash function commonly used to encrypt passwords for the purpose of increasing security. (Digamber, 2023.)

Jsonwebtoken is a popular Node.js module that offers a straightforward approach to generating and validating JSON Web Tokens (JWTs). JWTs serve as a commonly adopted token format for authentication and authorization in web applications. These tokens consist of a string of characters that enable secure transmission of information between various entities. In a Node.js application, utilizing the jsonwebtoken module typically involves three essential steps: token creation, token verification, and token decoding. (Kamani, 2022.)

Mongoose is an Object Data Modeling that is used with MongoDB and Node.js. It manages how data is related to one another, offers schema validation, and is used to translate objects between code and MongoDB. (Karnik, 2018.)

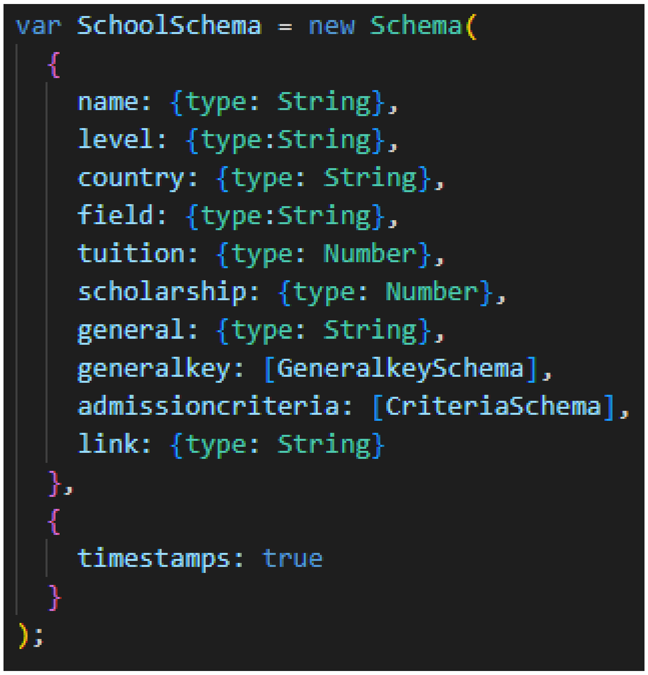


```
//MongoDB connection
const mongoose = require('mongoose');
const mongoURL = `mongodb+srv://mongoviope:${password}@atlascluster.9asbi9z.mongodb.net/educonnectdb?retryWrites=true&w=majority`;
mongoose.connect(mongoURL, { useNewUrlParser: true , useUnifiedTopology: true});
const db = mongoose.connection;
db.on('error', console.error.bind(console, 'MongoDB connection error'));
```

Figure 5. MongoDB connection

Figure 5 illustrates the MongoDB connection. To connect to the MongoDB database, I imported the Mongoose library and then used the Mongoose method to connect to the MongoDB database. The connect method has two arguments:

the connection string to my database and the definition of different connection options. In my case, I used the "useNewUrlParser" option to enable new URL parsing logic. I applied "mongoose.connection" to the constant "db" to access the object instance, which was initialized when the connection was opened. Then I used method "on" for "db" to listen for error events and log them to the console in the case of any connection errors.



```
var SchoolSchema = new Schema(
  {
    name: {type: String},
    level: {type:String},
    country: {type: String},
    field: {type:String},
    tuition: {type: Number},
    scholarship: {type: Number},
    general: {type: String},
    generalkey: [GeneralkeySchema],
    admissioncriteria: [CriteriaSchema],
    link: {type: String}
  },
  {
    timestamps: true
  }
);
```

Figure 6. School Schema

MongoDB is a NoSQL database. JSON can be stored in it, and the structure of these documents is subject to change as it is not mandatory like SQL databases. Mongoose schema defines the structure of MongoDB document. Figure 6 shows the mongoose schema for the school. Basically, the school schema is the collection of the types for the data that are stored in a school. Each school will have similar data listed in the schema, such as name, level, etc. Particularly the general key and the admission criteria will have a type of schema other than the primitive type. These two properties will have an array of different general contents and criteria. For more details about the school schema, table 1 shows the school's relational tables:

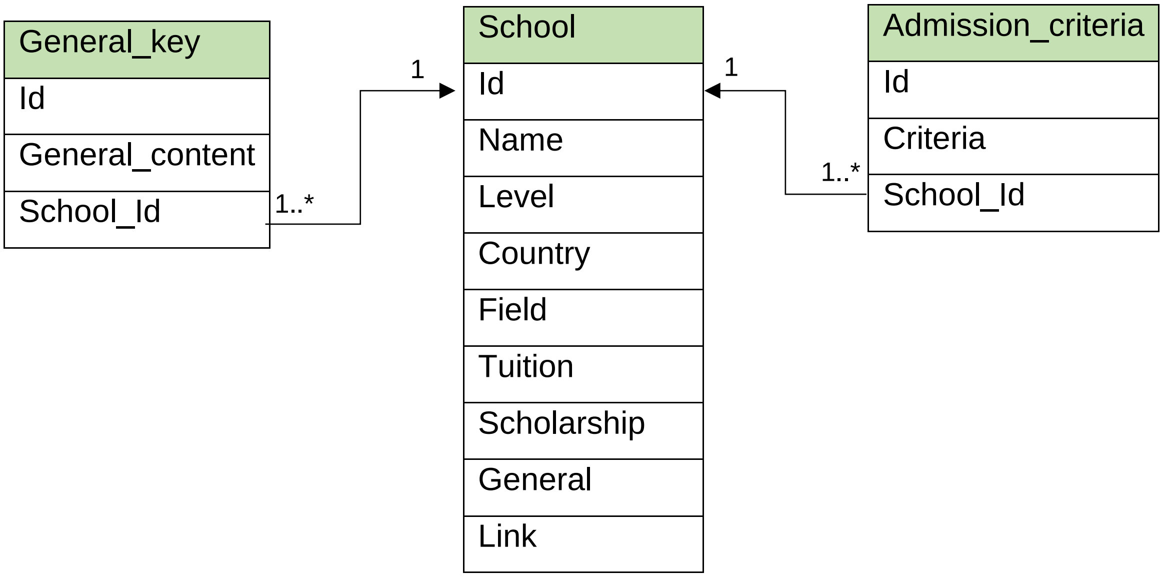
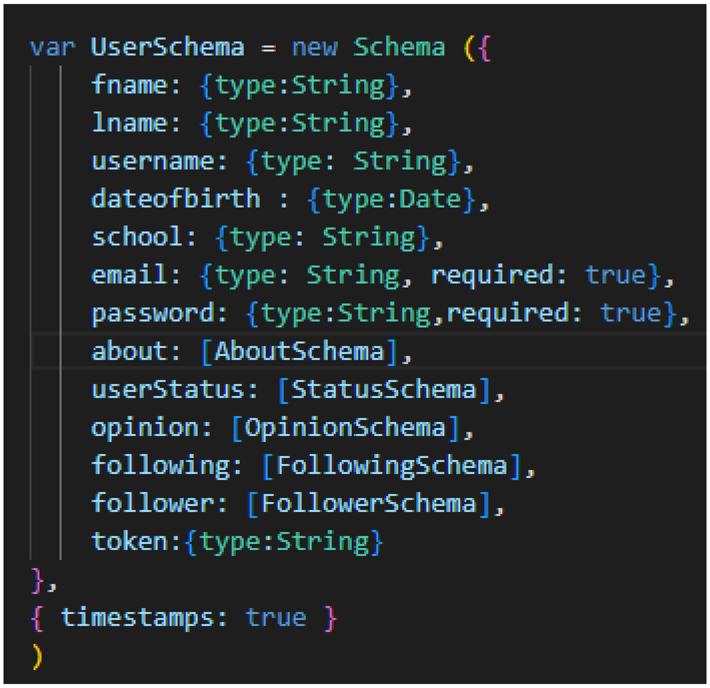


Table 1. School's relational table



```
var UserSchema = new Schema ({
    fname: {type:String},
    lname: {type:String},
    username: {type: String},
    dateofbirth : {type:Date},
    school: {type: String},
    email: {type: String, required: true},
    password: {type:String,required: true},
    about: [AboutSchema],
    userStatus: [StatusSchema],
    opinion: [OpinionSchema],
    following: [FollowingSchema],
    follower: [FollowerSchema],
    token:{type:String}
},
{ timestamps: true }
)
```

Figure 7. User Schema

Figure 7 shows the mongoose schema for the user. The user schema is a collection of the types of data that are stored in a user. This is built on the same concept as school schema. Table 2 shows the user's relational tables:

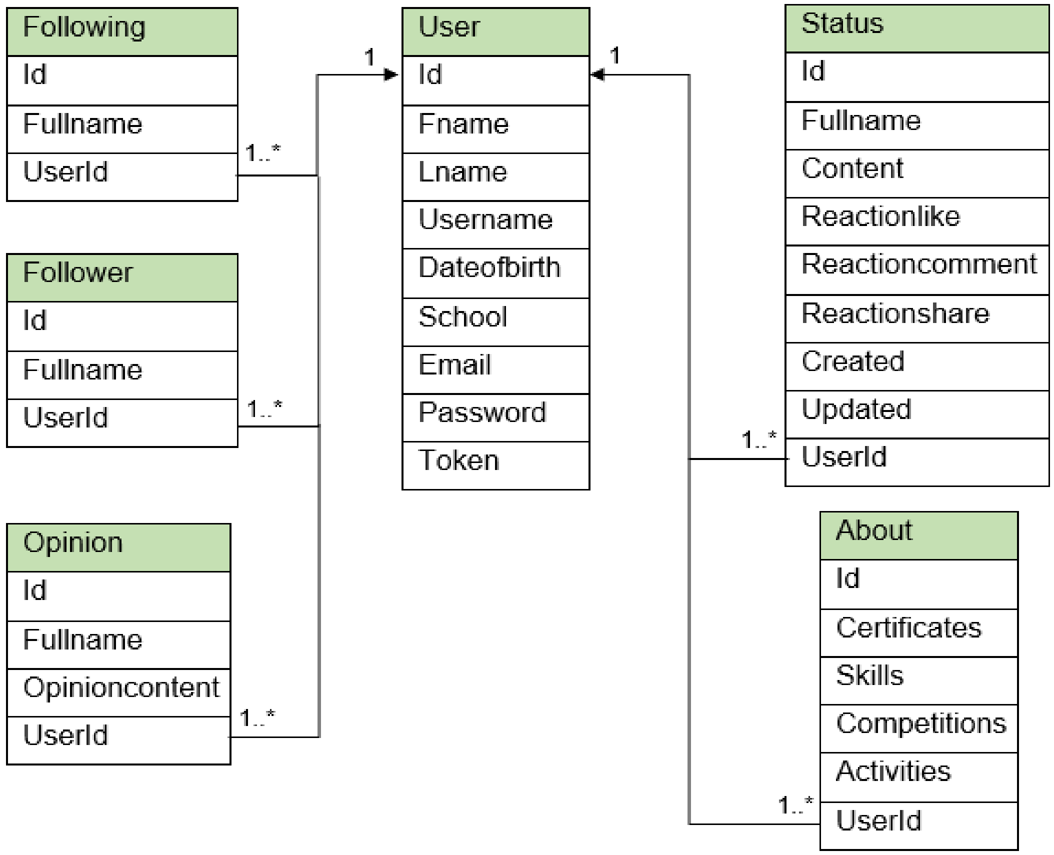
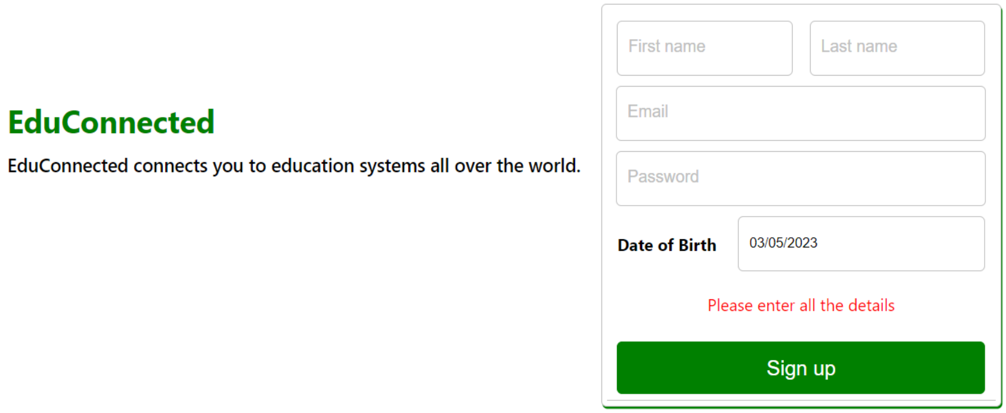


Table 2. User’s relation table

3.4 Sign up page

Figure 8 shows the sign-up page for the user to create a new account. The users need to enter all the input for sign-up. After the user enters all the input and clicks the sign up button, these data will be sent to the endpoint "/signup" of the backend to start checking the input. The registration process requires the user to provide their full name, email, and password. To create an account successfully, the users must ensure that their email address follows the correct syntax and that their password meets specific criteria. The password must be at least eight characters long and include an uppercase letter, a lowercase letter, at least one special character, and at least one number. If the users enter incorrect inputs, they will receive an error message immediately above the sign-up button, as shown in the figure.



EduConnected

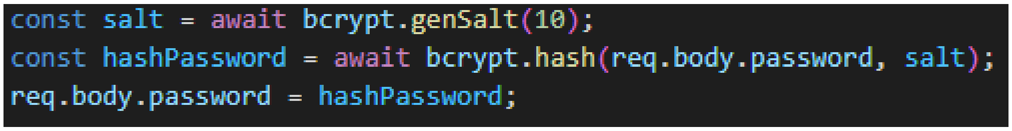
EduConnected connects you to education systems all over the world.

Date of Birth 03/05/2023

Please enter all the details

Figure 8. Sign up page

Once the users satisfy all the necessary conditions, their password will be hashed using the salt generated. Figure 9 demonstrates how the password is hashed. In this project, a salt with a complexity level of 10 rounds was generated, which is generally considered a good balance between security and performance. After saving the user's sign-up data, the token will be created based on the "id" ,which was just created with the user, with the secret key and expiration time so that the user can enter their own personal account. The token will be saved in the cookie and automatically deleted upon expiration or when the user logs out.

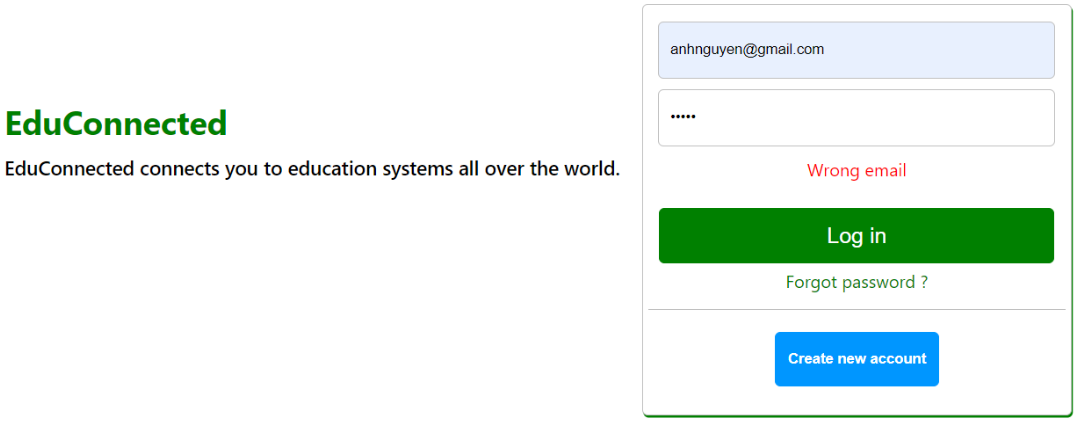


```
const salt = await bcrypt.genSalt(10);
const hashPassword = await bcrypt.hash(req.body.password, salt);
req.body.password = hashPassword;
```

Figure 9. Hash the password

3.5 Log in page

The log-in works similarly to the sign-up page. If the user has an account, she or he has to enter the correct email and password. If either email or password is wrong, then the user will see the error message above the log in button. Because each user will have a unique email, the email is checked in the backend by finding in the database the user who has the same email as the one given. Figure 10 illustrates the log in page.



EduConnected

EduConnected connects you to education systems all over the world.

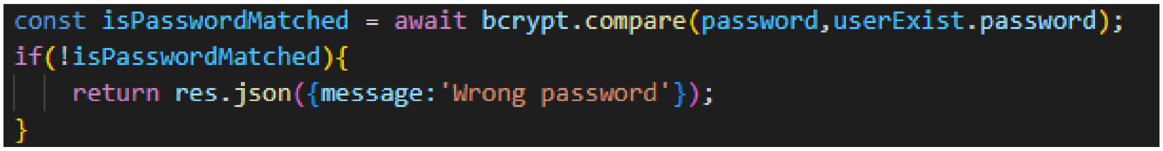
anhnguyen@gmail.com

••••••

Wrong email

Figure 10. Log in page

Figure 11 shows the code for checking whether the password is right or wrong. The first line uses the compare() function of the bcrypt library to compare a user-entered password to a previously hashed password retrieved from the database and stored in the userExist.password field. The compare() function is also asynchronous and returns a promise, which is why await is used to wait for the result. If the user-entered password matches the stored hashed password, compare() returns true, and isPasswordMatched is set to true. If the passwords do not match, compare() returns false, and isPasswordMatched is set to false and returns a message that will appear above the log-in button. If the passwords match, the login attempt is considered successful, the token will be created and stored in the cookie, and the user is granted access to the protected resources of the application.



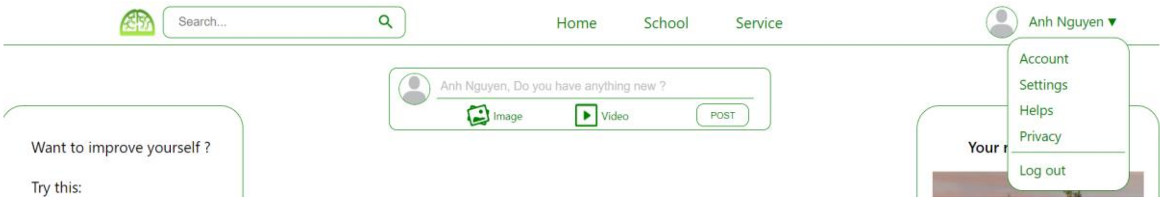
```
const isPasswordMatched = await bcrypt.compare(password,userExist.password);
if(!isPasswordMatched){
    return res.json({message:'Wrong password'});
}
```

Figure 11. Compare password

If the user forgets the password, she or he can click on "Forgot password?". In addition, if the user does not have an account, she or he can click on the "Create new account" button, and the page will navigate to the sign-up page.

3.6 Header bar

Figure 12 shows the header bar , which will be appear in every web page of this website. The header bar includes the logo of the website , the search bar, navigation to home page , search school page ,service page, the user avatar , user name. When clicking on username , it will appear a container that contains the navigation to account page , settings page , helps page , privacy page , and log out function.



Anh Nguyen, Do you have anything new ?

Want to improve yourself ?

Try this:

Figure 12. Header bar

Figure 13 shows the search bar. When the user enters any letter, there is a function that first converts each user's full name and the letter entered to lower case, then finds the user whose full name includes that letter. After that, any user matching the letter will be listed, and each name contains a link to that user.

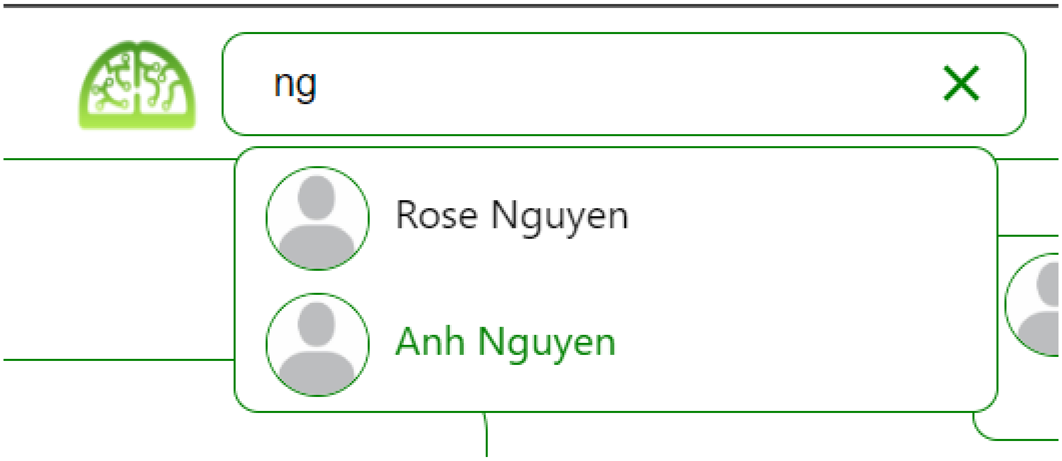
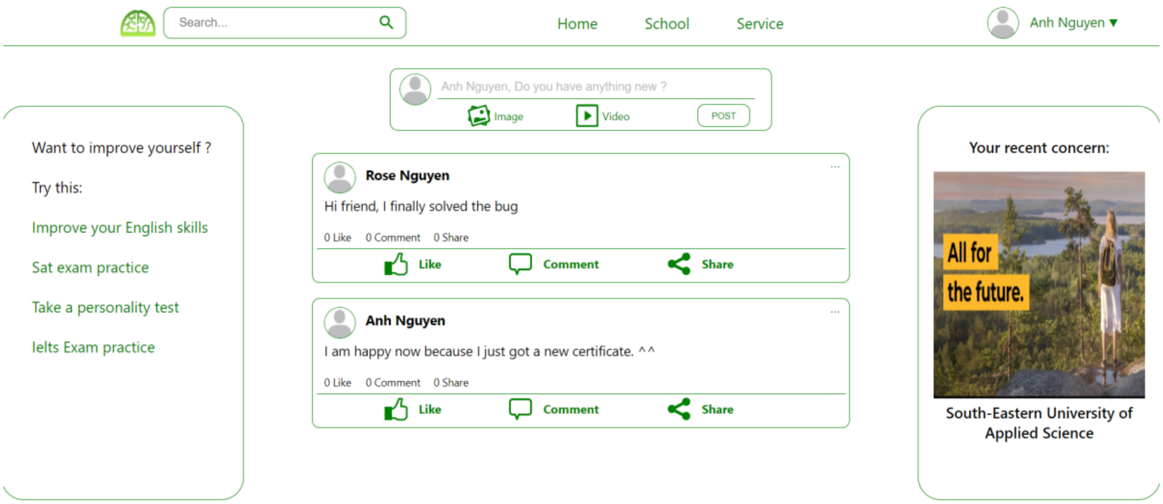


Figure 13. Search bar

3.7 Home page

Figure 14 shows the home page. On the right side of the home page, there is a box that contains several links to improving the user's skills, knowledge, and personality. On the left side, there will be some advertisement for the school that the user recently searched about or any topic that the user is concerned about. On the center, users can write new statuses and see the status of other users, which is updated within ten minutes. The users can also react to the other users status by liking or commenting.

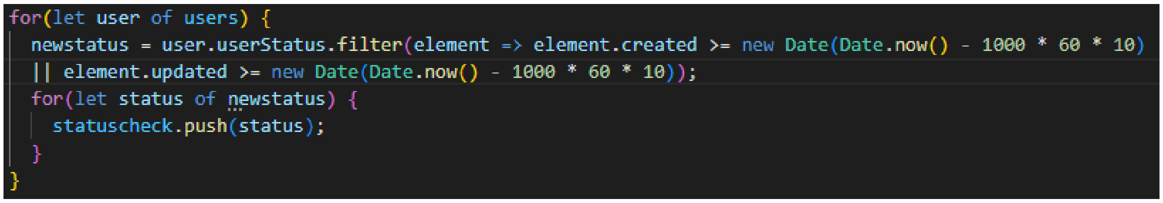


Hi friend, I finally solved the bug

I am happy now because I just got a new certificate. ^^

Figure 14. Home page

Figure 15 shows the method to get a new status from all users. Each user will contain an array of statuses, so I found the status that was just created or updated within ten minutes in this array. If one of the conditions is true, then the data for that status, including content and reaction, will be added to the new array called "statuscheck".

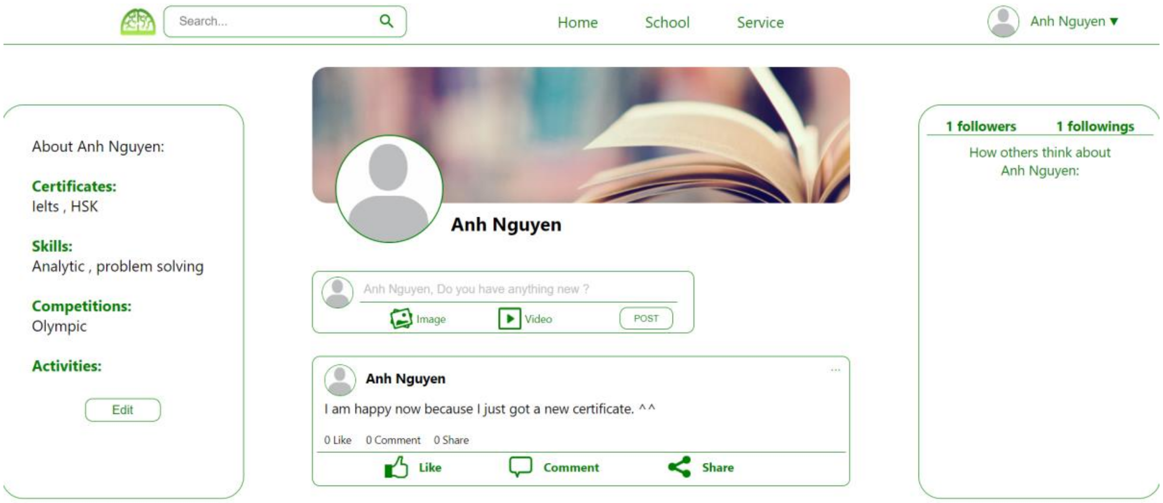


```
for(let user of users) {
  newstatus = user.userStatus.filter(element => element.created >= new Date(Date.now() - 1000 * 60 * 10)
  || element.updated >= new Date(Date.now() - 1000 * 60 * 10));
  for(let status of newstatus) {
    statuscheck.push(status);
  }
}
```

Figure 15. Fetch new status

3.8 User page

After selecting "Account" from the navigation bar, I entered the user page that is shown in figure 16. On the right of the user page, there is information about the user, such as certificates, skills, competitions, and activities in which the user participated. On the right is the opinion of other users—what others think about users. On the top of the container, it also shows how many people follow the user or how many people the user follows. The avatar, the background of the user, the place where the user can write their status, and all the statuses of the users are all located in the center of the page.



About Anh Nguyen:

Certificates:
Ielts , HSK

Skills:
Analytic , problem solving

Competitions:
Olympic

Activities:

Anh Nguyen

Anh Nguyen, Do you have anything new ?

Anh Nguyen

I am happy now because I just got a new certificate. ^^

0 Like 0 Comment 0 Share

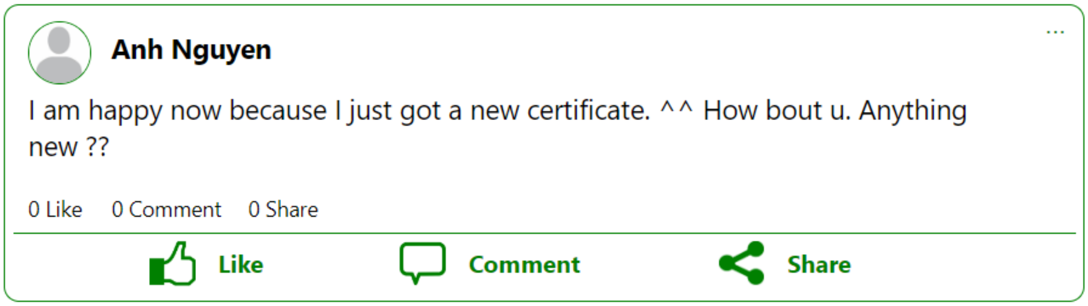
1 followers 1 followings

How others think about
Anh Nguyen:

Figure 16. User page

The user can edit the "about" section by clicking the "Edit" button. After clicking the "save" button, the data from the "about" section will be sent to the endpoint "/user/about" to update the user’s "about" property in the database.

When using the status feature in the website, users can share their thoughts and updates with their followers. Once a user has written and posted their status, the content will be sent to the "user/status" endpoint to update the "userStatus" property, allowing followers to view the user's latest updates.However, users may want to edit or delete their status at a later time. In order to perform these actions, the endpoint requires the "id" of the status. This unique identifier is used to locate the specific status in the array of "userStatus" and make the necessary changes. If a user chooses to edit their status, the client side will take the "id" of the status and send the "id" in the route parameter. The server will use the "id" to locate the specific status and then update the content with the user's new message. This allows users to make changes to their status without having to delete and recreate it. If a user decides to delete their status, the application will again use the "id" to locate the specific status and remove it from the array of "userStatus". This ensures that only the most up-to-date and relevant information is available to the user's followers. Figure 17 shows the status after editing.



Anh Nguyen

I am happy now because I just got a new certificate. ^^ How bout u. Anything new ??

Figure 17. The status after editing

Every user can like and comment on the post. The like reaction works like this: if the user clicks once, then the server will check if this user liked it or not; if the user liked it, then delete the user from the liked list; otherwise, add the user to the liked list.

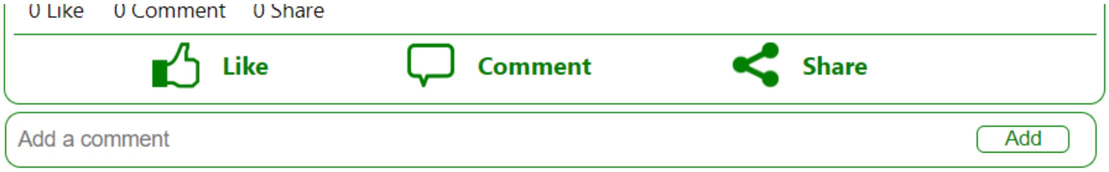
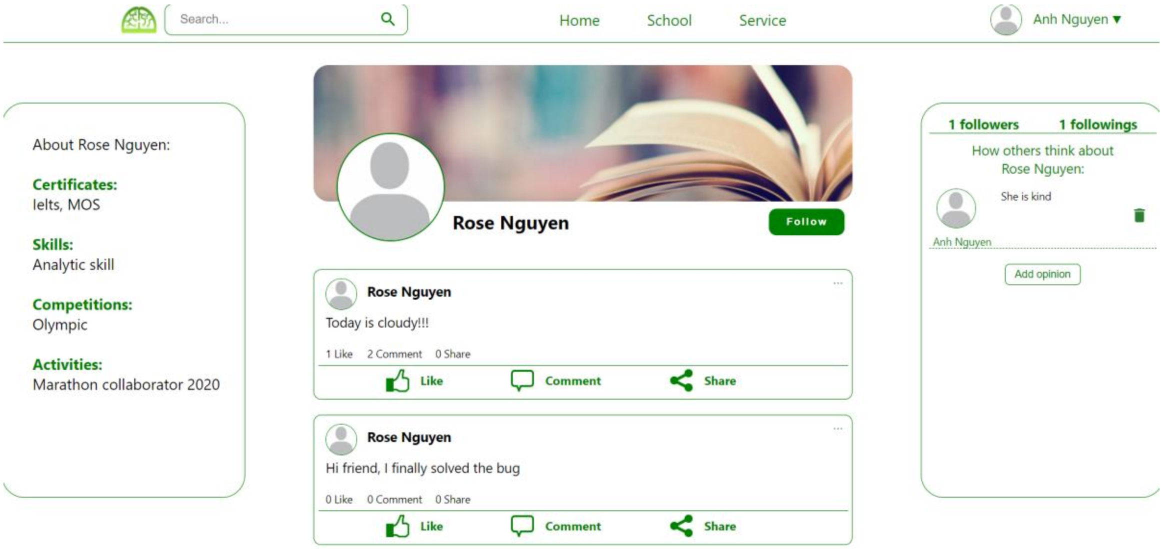


Figure 18. The window to add comment

Figure 18 shows the window to add a comment after clicking on the comment icon. After clicking add, the content and user’s full name will be added to the reaction comment list in the database. The user can also check who liked or commented on the status.

3.9 Other user page

Figure 19 displays the profile of a user named "Rose Nguyen." On the right side of the page, there is various information about this user, such as their skills, education, and competitions. This allows visitors to quickly learn more about the person behind the profile. In the center of the page, there is Rose's avatar and background image, which help to personalize the profile and make it stand out. There is also her latest status update. The user has the option to click the "Follow" button. When clicked, the user will be added to Rose's list of followers, while Rose will be added to the user's list of people the user is following. This feature makes it easy to stay connected with others on the platform and build a community of like-minded individuals. In addition to following Rose, users can also share their opinions about her by clicking the "Add Opinion" button. This opens up a section on the right side of the page where users can leave thoughts and comments about Rose. However, it's important to note that only the user who wrote the opinion and the person who received it will have the authorization to delete it. This helps to maintain a respectful and safe environment for all users on the platform.



About Rose Nguyen:

Certificates:
Ielts, MOS

Skills:
Analytic skill

Competitions:
Olympic

Activities:
Marathon collaborator 2020

Rose Nguyen

Rose Nguyen
Today is cloudy!!!
1 Like 2 Comment 0 Share

Rose Nguyen
Hi friend, I finally solved the bug
0 Like 0 Comment 0 Share

1 followers 1 followings

How others think about Rose Nguyen:

She is kind
Anh Nguyen

Figure 19. Other user page

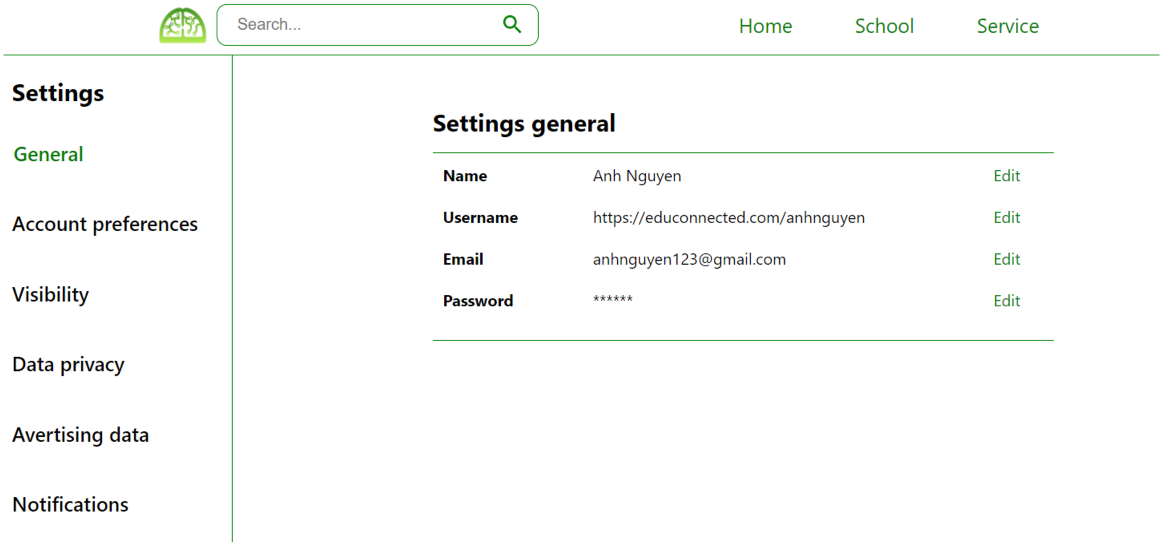
In order to add an opinion, the client will send the content in the request body and the username of the user who is receiving the opinion on the route parameter. The server will find the receiver user and add the opinion to the receiver user's opinion list.

Deleting an opinion can be accomplished in a few different ways, depending on who wants to delete it. If the receiver of the opinion decides they no longer want it on their profile, the server simply needs the "id" of the opinion to delete it from their opinion list. However, if the person who wrote the opinion wants to delete it, the client will need to send both the receiver "id" and the opinion "id" to the server. The server will then locate the opinion in the receiver's data, check if the writer is the same person who wrote the opinion, and delete it if they have authorization. If they do not have authorization, an alert will be triggered to inform them that they cannot delete the opinion.

The follow function operates in a similar way to the reaction like function. When a user clicks the "Follow" button on someone's profile, the client will send that person's "id" to the server. The server will first find that person, and then check if the user is already following them or not. If the user is following them, the user will be removed from the follower list, and that person will be removed from the user's following list. If the user is not already following them, the user will be added to the follower list, and that person will be added to the user's following list.

3.10 Setting page

The settings page is shown in Figure 20. The settings page contains some general data about the user, such as email, username, name, and password. Each of them can be edited by clicking "Edit".

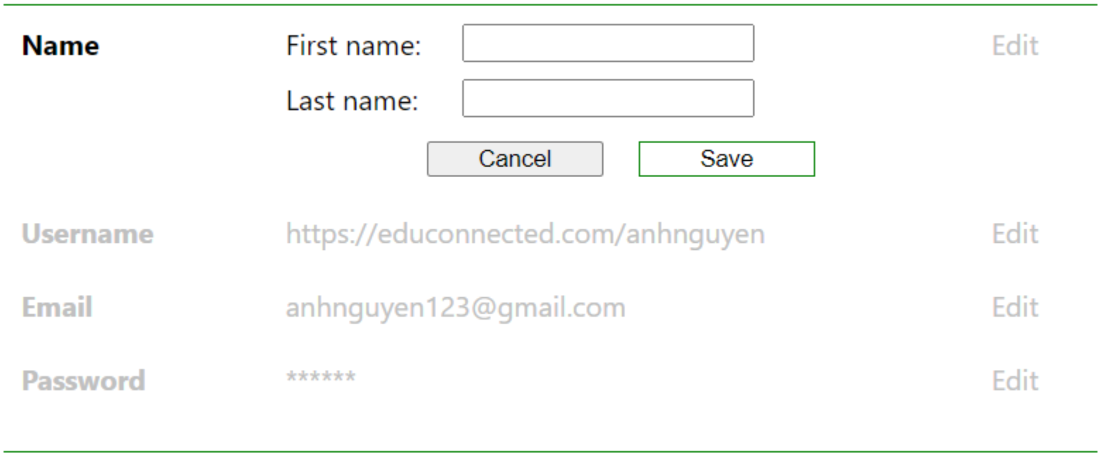


Settings general

Name	Anh Nguyen	Edit
Username	https://educonnected.com/anhnguyen	Edit
Email	anhnguyen123@gmail.com	Edit
Password	******	Edit

Figure 20. Setting page

Each of them will have a different endpoint. For example, a full name will have the endpoint "/user/fullname" . After clicking the "Save" button, the client will send the content in the request body, and then the server will update the user's data. Figure 21 demonstrates the example setting.



Name First name: Last name:

Username https://educonnected.com/anhnguyen

Email anhnguyen123@gmail.com

Password ******

Figure 21. Setting example

However, editing password required the user to enter the current password , and retype the new password. The password still have same condition as when sign up. The password must be at least eight characters long and contain at least one uppercase letter, one lowercase letter, one special character, and one number. If the password is not pass the condition , or the user enter wrong current password , or the user retype the password that not the same with new password , then there will be the error above those button. Otherwise , the new password will be hashed and store in the database. Figure 22 displays the password setting.

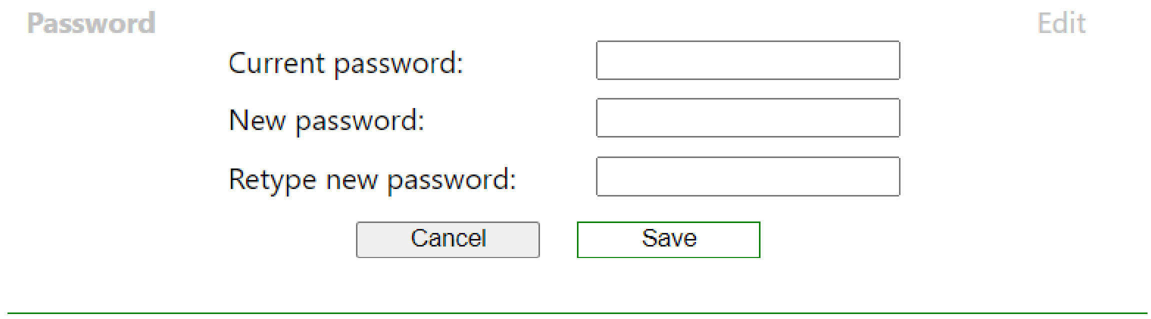
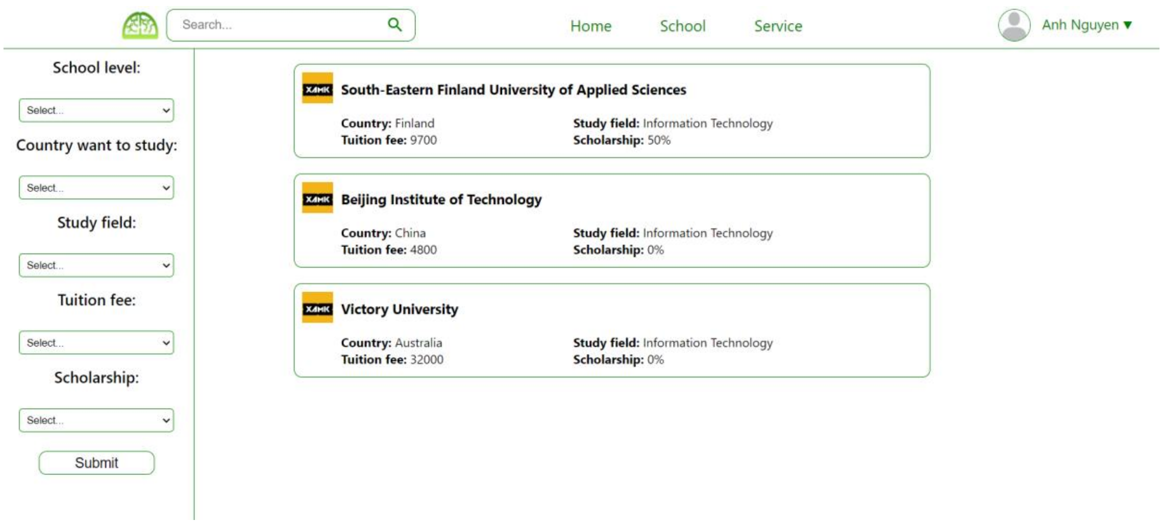


Figure 22. Password setting

3.11 School page

The school page, as shown in figure 23, offers users the ability to sort and filter schools based on five different criteria located on the left side of the page. These criteria include location, tuition fees, study program,school level, and scholarship. Users can select any combination of these criteria to narrow down the list of schools to their specific preferences. The filter feature is designed to ensure that users have the ability to tailor their search for schools to their unique needs without having to manually sift through a long list of options.



South-Eastern Finland University of Applied Sciences

Country: Finland
Tuition fee: 9700
Study field: Information Technology
Scholarship: 50%

Beijing Institute of Technology

Country: China
Tuition fee: 4800
Study field: Information Technology
Scholarship: 0%

Victory University

Country: Australia
Tuition fee: 32000
Study field: Information Technology
Scholarship: 0%

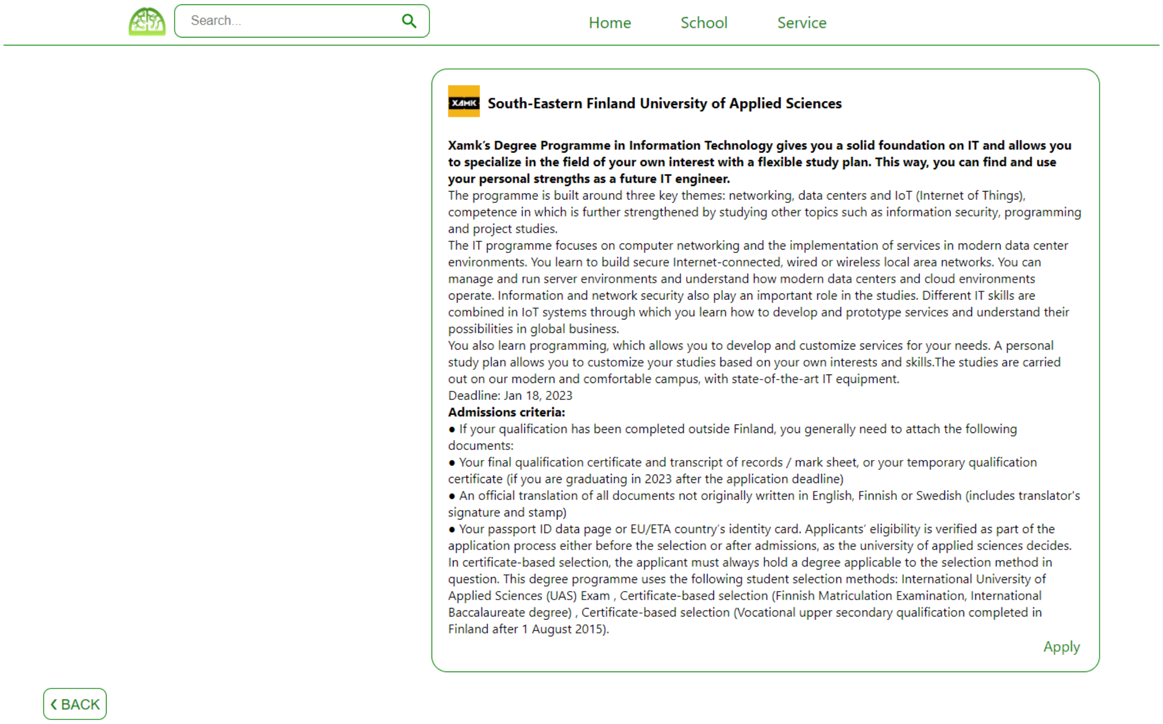
Figure 23. School page

Users will get a list of schools that meet their criteria on the right side of the website. By default, the list displays all schools, but as users apply filters, the list dynamically updates to show only the schools that meet their criteria.

Additionally, users can click on any school name from the list to view detailed information about that school.

For the search action, after the user chooses the criteria and clicks submit, the client will send the content of the criteria in the request body to the endpoint "schools/find" . Then the server finds and returns the list of schools that meet the criteria.

Each of the school's names that are listed contains a hyperlink that helps navigate to the school detail page. This link also contains the "id" of the school. The link's "id" functions as a parameter. Then, the "id" will be passed to the endpoint "schools/:id" to send to the server. After that, the server will return only the school that has this "id".



South-Eastern Finland University of Applied Sciences

Xamk's Degree Programme in Information Technology gives you a solid foundation on IT and allows you to specialize in the field of your own interest with a flexible study plan. This way, you can find and use your personal strengths as a future IT engineer.
The programme is built around three key themes: networking, data centers and IoT (Internet of Things), competence in which is further strengthened by studying other topics such as information security, programming and project studies.
The IT programme focuses on computer networking and the implementation of services in modern data center environments. You learn to build secure Internet-connected, wired or wireless local area networks. You can manage and run server environments and understand how modern data centers and cloud environments operate. Information and network security also play an important role in the studies. Different IT skills are combined in IoT systems through which you learn how to develop and prototype services and understand their possibilities in global business.
You also learn programming, which allows you to develop and customize services for your needs. A personal study plan allows you to customize your studies based on your own interests and skills.The studies are carried out on our modern and comfortable campus, with state-of-the-art IT equipment.
Deadline: Jan 18, 2023
Admissions criteria:
● If your qualification has been completed outside Finland, you generally need to attach the following documents:
● Your final qualification certificate and transcript of records / mark sheet, or your temporary qualification certificate (if you are graduating in 2023 after the application deadline)
● An official translation of all documents not originally written in English, Finnish or Swedish (includes translator's signature and stamp)
● Your passport ID data page or EU/ETA country's identity card. Applicants' eligibility is verified as part of the application process either before the selection or after admissions, as the university of applied sciences decides. In certificate-based selection, the applicant must always hold a degree applicable to the selection method in question. This degree programme uses the following student selection methods: International University of Applied Sciences (UAS) Exam , Certificate-based selection (Finnish Matriculation Examination, International Baccalaureate degree) , Certificate-based selection (Vocational upper secondary qualification completed in Finland after 1 August 2015).

Figure 24. School detail page

The school detail page, as shown in figure 24, provides users with comprehensive information about a particular school. In addition to the basic information such as the name and location of the school, users can also find details on the school's focus on the program and areas of expertise, admission criteria, application deadlines, and other relevant information. Furthermore, the page features a link that redirects the user to the school's application page, making the application process more convenient and accessible for the user. The "Back" button on the page allows users to easily return to the previous page, enabling them to continue their search for other schools.

3.12 Service page

Figure 25 represents the service page of the website, which offers a variety of services related to different fields. All users who require assistance in any of the fields listed on the page can take advantage of these services. When a user selects a service, they are routed to the service detail page, where they can consult the service in more detail and contact for advice and cooperation.

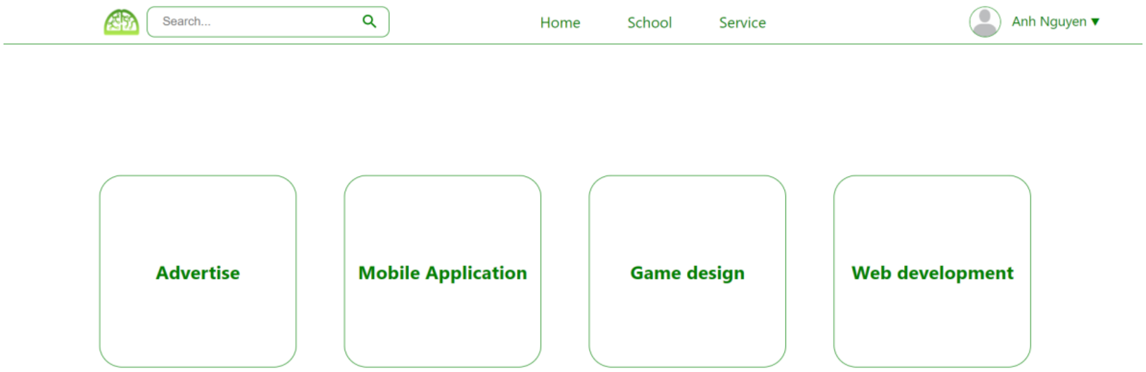
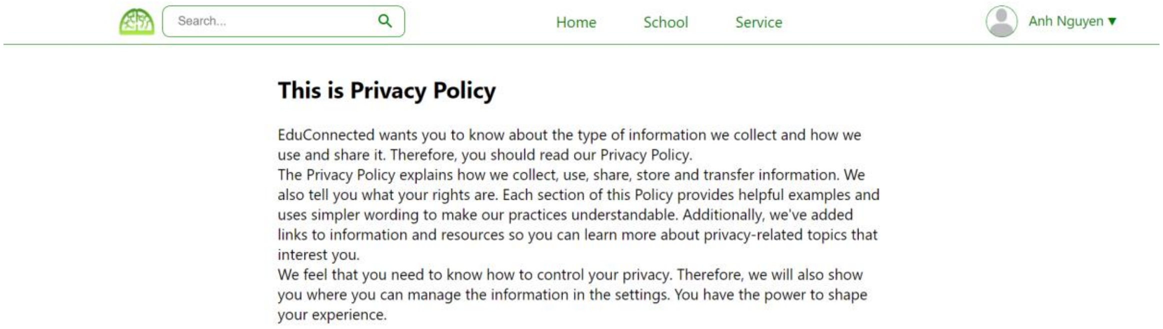


Figure 25. Service page

The available services cover a wide range of fields, including advertising, mobile application development, game design, and web development. For advertising services, the service provider can assist the user in promoting their products or services by creating advertisements that are visually appealing and attracting attention. Mobile application development services can help users build custom applications for their business, organization, or personal use. Game design services can assist users create engaging and entertaining games for various platforms. Lastly, web development services can help users in building and maintaining a website that is responsive, user-friendly, and visually appealing.

3.13 Privacy page

The Privacy page, displayed in Figure 26, is an important part of the website as it helps the users understand how their information is collected, used, and shared. The page contains several sections dedicated to describing the privacy policy of the website, including examples to help the users understand how their data is being handled. The page also outlines the users' rights to manage their information and privacy settings. In addition to providing a comprehensive understanding of the website's privacy policy, the privacy page also ensures transparency in data collection and usage. However, the website guarantees that the collected information is not shared with any third-party without the user's permission. The privacy page also allows the users to manage their privacy settings according to their preferences. Users can also manage the visibility of their personal information on the website, and their communication preferences in the settings.



This is Privacy Policy

EduConnected wants you to know about the type of information we collect and how we use and share it. Therefore, you should read our Privacy Policy.
The Privacy Policy explains how we collect, use, share, store and transfer information. We also tell you what your rights are. Each section of this Policy provides helpful examples and uses simpler wording to make our practices understandable. Additionally, we've added links to information and resources so you can learn more about privacy-related topics that interest you.
We feel that you need to know how to control your privacy. Therefore, we will also show you where you can manage the information in the settings. You have the power to shape your experience.

Figure 26. Privacy page

4 FUTURE DEVELOPMENT

Implementing a message box feature to the website would be a valuable addition to enhance user communication. Users can easily communicate with one another through texting or calling, and recruiters can easily set up interviews with potential candidates. The messaging feature can also help to build a sense of community among users and increase interaction on the site. Furthermore, the ability to communicate directly with recruiters through the messaging feature can help streamline the recruitment process and make it more efficient for both recruiters and candidates.

Posts should also be shared in the message section via other platforms such as Instagram, Messenger, Viber, and WhatsApp, not just on the website. Additionally, the function to upload an avatar and image for the user should be added. Users will be able to share their favorite moments, photos such as recently received certifications on their timeline.

Adding a low-cost service so that schools can post admissions themselves, and when there is any update, the schools will edit it themselves so that enrollment information is always up-to-date. For schools, this will help students learn more about new majors or new admissions. For students, they will always have the latest information about enrollment at the school they are choosing. This also benefits the website; there will be no lack of information, leading to misleading information for users.

To help students make better decisions when choosing a school, adding a section of student reviews and comments about the school on the school detail page should be considered. This will help other students read and feel about the school based on previous student experiences, thereby making more appropriate choices. Besides, this evaluation and comment section will also assist the school's developers in gathering more information about the school and determining what actions to take to improve school quality.

5 CONCLUSION

The goal of this project was to create a website that connects the education system around the world. Through this website, schools attract students from different countries and can also evaluate a student through their personal page on the website, while students can browse through a vast selection of schools and find the one that best suits their needs.

This project was built with full-stack technologies that require knowledge of ReactJS, NodeJS, HTML, and CSS. NodeJS served as the server-side application, responsible for transferring data between the client and the database. ReactJS, on the other hand, was used as the client-side application, enabling users to interact with the website's user interface. The HTML code supported the design of the front-end and provides the structural framework for the website. Finally, CSS was utilized to stylize the web pages and make them visually appealing to the users.

The implementation phase defined how the front-end and back-end will interact. The client interacts with the server through endpoints and transmits data to the server in JSON format, which is then processed by the server according to the specified HTTP methods. The server validates the data received from the client and sends back an appropriate response, whether it is a success message, an error message or a JSON data.

The result of the thesis was that the website was developed with various functions, such as allowing users to log in to their personal account, update their profile information, post status updates, react to others' posts, follow and unfollow other users, and search for schools based on different criteria.

Further improvement for the website is to add some news functions, such as a message box feature that allows users to communicate directly with one another; a share function that enables users to share posts on other messaging platforms such as Messenger and so on; an upload image function that would allow users to share images; a service that enables schools to post their admissions requirements directly; and a rating system on the school detail page.

REFERENCES

Arancio, S. 2021. ReactJS: A brief history. WWW document. Available at: https://medium.com/@sjarancio/reactjs-a-brief-history-3c1e969a477f [Accessed 19 Mar 2023]

Astari, S. 2023. What Is HTML? Hypertext Markup Language Basics Explained. WWW document. Available at: https://www.hostinger.com/tutorials/what-is-html [Accessed 20 Mar 2023]

Brian, P. 2023. CSS. WWW document. Available at: https://techterms.com/definition/css [Accessed 20 Mar 2023]

Decpk, 2021. React JSX in Depth. WWW document. Available at: https://www.geeksforgeeks.org/react-jsx-in-depth/ [Accessed 19 Mar 2023]

Digamber. 2023. Using NPM Bcrypt JS Package to Hash Password in Node. WWW document. Available at: https://www.positronx.io/using-npm-bcryptjs-package-to-hash-password-in-node/ [Accessed at 02 May 2023]

Dorairajan, A. 2022. What is an API? WWW document. Available at: https://blog.axway.com/learning-center/apis/basics/what-is-an-api [Accessed 01 May 2023]

Domantas, G. 2023. Types of CSS: Inline, External and Internal Definitions and Differences Explained. WWW document. Available at: https://www.hostinger.com/tutorials/difference-between-inline-external-and-internal-css [Accessed 20 Mar 2023]

Erinc, Y. 2020. The Benefits of Going RESTful – What is REST and Why You Should Learn About It WWW document. Available at: https://www.freecodecamp.org/news/benefits-of-rest/ [Accessed 01 May 2023]

Gohel, H. 2022. MongoDB : Advantages and Disadvantages. WWW document. Available at: https://www.infraveo.com/blog/mongodb-advantages-and-disadvantages/ [Accessed 20 Mar 2023]

Gouravhammad. 2023. Body-parser middleware in Node.js. WWW document. Available at: https://www.geeksforgeeks.org/body-parser-middleware-in-node-js/ [Accessed at 02 May 2023]

Gupta, L. 2022. What is REST? WWW document. Available at: https://restfulapi.net/ [Accessed 01 May 2023]

Hameed, S. 2023. HTML History | Explained. WWW document. Available at: https://linuxhint.com/html-history/ [Accessed 20 Mar 2023]

Hines, K. 2022. The History Of Social Media. WWW document. Available at: https://www.searchenginejournal.com/social-media-history/ [Accessed 16 Mar 2023]

Jordana, A. 2022. What Is React & How Does It Actually Work?. WWW document. Available at: https://www.hostinger.com/tutorials/what-is-react [Accessed 19 Mar 2023]

Karnik, N. 2018. Introduction to Mongoose for MongoDB. WWW document. Available at: https://www.freecodecamp.org/news/introduction-to-mongoose-for-mongodb-d2a7aa593c57/#:~:text=Mongoose%20is%20an%20Object%20Data,of%20those%20objects%20in%20MongoDB. [Accessed 02 May 2023]

Kamani, S. 2023. Implementing JWT Authentication in Node.js. WWW document. Available at: https://www.sohamkamani.com/nodejs/jwt-authentication/?utm_content=cmp-true [Accessed 02 May 2023]

Malhotra, A. 2021. Cookie Parser. WWW document. Available at: https://tutorial.techaltum.com/cookie-parser.html [Accessed at 02 May 2023]

Mojeed, I. 2022. What is the virtual DOM in React?. WWW document. Available at: https://blog.logrocket.com/virtual-dom-react/ [Accessed 19 Mar 2023]

MONGODB HISTORY AND ADVANTAGES. WWW document. Available at: https://bytescout.com/blog/2019/09/mongodb-history-and-advantages.html [Accessed 20 Mar 2023]

Saini, A. 2021. What is MongoDB – Working and Features. WWW document. Available at: https://www.geeksforgeeks.org/what-is-mongodb-working-and-features/ [Accessed 20 Mar 2023]

Semah, B. 2022. What Exactly is Node.js? Explained for Beginners. WWW document. Available at: https://www.freecodecamp.org/news/what-is-node-js/ [Accessed 19 Mar 2023]

Sharma, A. 2023. What Is Express JS In Node JS? WWW document. Available at: https://www.simplilearn.com/tutorials/nodejs-tutorial/what-is-express-js [Accessed 02 May 2023]

Sheldon, R. Denman,J. 2022. Node.js (Node). WWW document. Available at: https://www.techtarget.com/whatis/definition/Nodejs [Accessed 19 Mar 2023]

Singhal, A. 2023. CORS in Nodejs. WWW document. Available at: https://www.scaler.com/topics/nodejs/cors-in-node-js/ [Accessed 02 May 2023]

Top 5 NodeJS Benefits and Top 5 Limitations: What They Mean for Your Business. 2023. WWW document. Available at: https://anywhere.epam.com/business/node-js-pros-and-cons [Accessed 19 Mar 2023]

Vatsal, S. 2022. Advantages And Disadvantages of CSS: Know All About The Design Language. WWW document. Available at: https://unstop.com/blog/advantages-and-disadvantages-of-css [Accessed 20 Mar 2023]

Verma, D. 2020. What is JSX in React JS?. WWW document. Available at: https://blog.dheerajverma.in/what-is-jsx-in-react-js/ [Accessed 19 Mar 2023]

What is virtual dom and its benefits that you must know. WWW document. Available at: https://blog.arrowhitech.com/virtual-dom-its-definition-and-benefits-that-you-must-know/ [Accessed 19 Mar 2023]

What is Rest API? Learn how Restful API works. 2021. WWW document. Available at: https://itnavi.com.vn/blog/rest-api-la-gi [Accessed 02 May 2023]

LIST OF FIGURES