***PRN222 Project***

***Edunext Web***



Student: Nghiem Thi Hong Ngoc

Lecturer: Do Thi Loan

Subject: PRN222

Class: SE1816-NET

# 

# Acknowledgments

I would like to express my sincere gratitude to **LoanDT34**, my lecturer for the **PRN222** course, for providing guidance and valuable insights throughout the development of this project. Her support has been instrumental in helping me understand and apply **C# .NET MVC** in real-world web application development.

This project, **EduNext**, was created as part of my studies to enhance the learning experience for students by providing a structured platform for managing courses, study materials, exercises, and student interactions. The journey of developing this application has significantly improved my technical skills in **C# .NET** and deepened my understanding of web application development.

I would also like to thank my peers and anyone who has contributed ideas, feedback, and support during the development process. Your encouragement has been greatly appreciated.

Thank you!

Table of Contents

[Acknowledgments 3](#_4d34og8)

[1. Theory 5](#_17dp8vu)

[a. ASP.NET 5](#_3rdcrjn)

[i. Overview 5](#_26in1rg)

[ii. Key Features 5](#_lnxbz9)

[b. Entity Framework 5](#_35nkun2)

[i. Overview 5](#_1ksv4uv)

[ii. Approach 5](#_44sinio)

[c. CSHTML 6](#_2jxsxqh)

[i. Overview 6](#_z337ya)

[ii. Key feature 6](#_3j2qqm3)

[d. Razor Syntax 6](#_1y810tw)

[i. Overview 6](#_4i7ojhp)

[ii. Syntax 6](#_2xcytpi)

[e. SQL Server 8](#_3whwml4)

[i. Definition 8](#_2bn6wsx)

[ii. Syntax 8](#_3as4poj)

[f. Bootstrap 9](#_1pxezwc)

[2. System Design 9](#_32hioqz)

[a. Function 9](#_1hmsyys)

[b. Use Case Diagram 10](#_41mghml)

[c. Screen Flow 11](#_2grqrue)

[d. Class Diagram 12](#_tm3dtduoc407)

[e. Entity Relationship Diagram 13](#_3fwokq0)

[f. Sequence diagram 13](#_pznca09jrs2k)

[3. Website Design 15](#_1mrcu09)

[a. Login interface 15](#_46r0co2)

[b. Register interface 15](#_2lwamvv)

[c. PreResetPassword interface 16](#_wpl3xbfu1hvh)

[d. ResetPassword interface 16](#_mxuzaimqgi5)

[e. User Details interface 17](#_2r0uhxc)

[f. Update User interface 17](#_1664s55)

[g. Change Password interface 18](#_tpsqyydothm5)

[h. Course Index interface 18](#_111kx3o)

[i. Create Course interface 19](#_fzugmy2dvsc)

[j. Update Course interface 19](#_3l18frh)

[k. Semester Index interface 20](#_25b2l0r)

[l. Create Semester interface 20](#_fpnpzxv3d4rm)

[m. Update Semester interface 21](#_ifrjdghwogk9)

[n. Classes Index interface 21](#_tq806zk2oz7k)

[o. Class Enrollments Index interface 22](#_agxv84fkgvv7)

[p. Slot Index interface 22](#_il5i39gpi4ji)

[q. Update Slot interface 23](#_mnebx0tkjvzf)

[r. User Index interface 23](#_206ipza)

[s. Create User interface 24](#_4k668n3)

[t. Update User interface 24](#_2zbgiuw)

[u. Class Slot Content Detail interface 25](#_1egqt2p)

[v. Question Details Interface 25](#_3ygebqi)

[w. Assignment Details interface 25](#_2dlolyb)

[x. Create Assignment interface 26](#_sqyw64)

[y. Update Assignment interface 26](#_1lk0mu9xl7pg)

[z. Update Question interface 27](#_3cqmetx)

[Conclusion 27](#_34g0dwd)

# Theory

## ASP.NET

### Overview

ASP.NET is a **web application framework** developed by Microsoft that allows developers to build **dynamic web applications, websites, and services**. It is part of the .NET platform and provides powerful tools and libraries for creating **secure, scalable, and high-performance web applications**.

### Key Features

1. **MVC Architecture** – Supports **Model-View-Controller (MVC)** for better code organization and separation of concerns.
2. **Razor Syntax** – A simple and efficient way to write server-side code within HTML.
3. **Cross-Platform** – Works on **Windows, macOS, and Linux** when using .NET Core.
4. **Integration with Databases** – Supports **Entity Framework (EF)** for database operations.
5. **Dependency Injection** – Helps in managing dependencies and improving code maintainability.

## Entity Framework

### Overview

Entity Framework (EF) is an **Object-Relational Mapping (ORM) framework** for .NET applications. It simplifies **database interactions** by allowing developers to work with **C# objects instead of raw SQL queries**. EF helps in performing CRUD (Create, Read, Update, Delete) operations efficiently while maintaining **data integrity and abstraction**.

### Approach

1. **Code-First** – Developers define models using C# classes, and EF generates the database schema automatically.
2. **Database-First** – Uses an existing database, and EF generates models based on it.
3. **Model-First** – Allows developers to create an **EDMX** model visually, which generates both classes and database schema.

## CSHTML

### Overview

CSHTML (**C# HTML**) is a file format used for **views in ASP.NET MVC and ASP.NET Core MVC**. It combines **HTML markup** with **Razor syntax** to create **dynamic web pages**.

### Key feature

1. **Uses Razor Syntax** – Allows embedding **C# code inside HTML**.
2. **Dynamic Content Rendering** – Supports **model binding** for displaying data.
3. **Supports Layouts & Partial Views** – Enables reusable UI components.
4. **Separation of Concerns** – Keeps UI logic separate from business logic (Controller & Model).

## Razor Syntax

### Overview

Razor is a **lightweight, server-side templating engine** used in **ASP.NET MVC and ASP.NET Core MVC**. It allows embedding **C# code inside CSHTML files** for generating **dynamic web content**. Razor syntax is **easy to learn**, **clean**, and **optimized for performance**.

### Syntax

1. C# Code Block



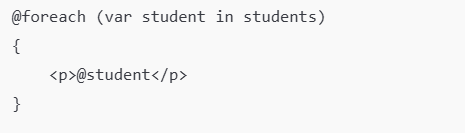
1. Inline Expressions



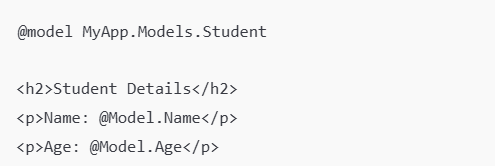
1. Conditional Statements



1. Loops

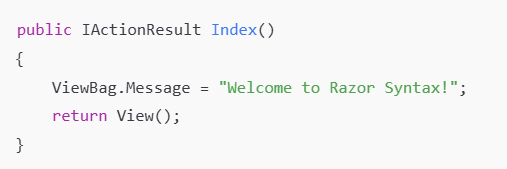


1. Working with Models



1. Using ViewBag and ViewData

Controller:



CSHTML View:



1. Using Layouts and Sections





1. Using Partial Views



1. Comments



## SQL Server

### Definition

SQL Server is a relational database management system (RDBMS) developed by Microsoft. It's widely used for storing and retrieving data in various applications, from small businesses to large enterprises.

### Syntax

- SQL (Structured Query Language) is used to communicate with and manipulate databases. SQL Server syntax elements:

+ SELECT Statement: Used to retrieve data from a database.

+ INSERT Statement: Used to insert new records into a table.

+ UPDATE Statement: Used to update existing records in a table.

+ DELETE Statement: Used to delete records from a table.

+ CREATE TABLE Statement: Used to create a new table in the database.

+ ALTER TABLE Statement: Used to modify an existing table (e.g., add or drop columns).

+ DROP TABLE Statement: Used to delete a table from the database.

+ JOIN Clause: Used to combine rows from two or more tables based on a related column between them.

+ WHERE Clause: Used to filter records based on a specified condition.

+ ORDER BY Clause: Used to sort the result set in ascending or descending order.

+ GROUP BY Clause: Used to group rows that have the same values into summary rows.

+ HAVING Clause: Used to filter group rows after the GROUP BY clause.

## Bootstrap

* Bootstrap is an HTML, CSS, and JavaScript framework that allows users to easily design websites according to a specific standard, creating mobile-friendly websites for devices such as mobile phones, iPads, and tablets.
* Bootstrap includes basic elements such as typography, forms, buttons, tables, navigation, modals, image carousels, and more. Within Bootstrap, there are additional components and JavaScript support for designing your responsive layouts more easily, conveniently, and quickly.
* Bootstrap is one of the most widely used frameworks globally for building websites.
* Bootstrap has established its own standard and is highly favored by users. Therefore, we often hear a very common phrase "Design according to Bootstrap standards."

# System Design

## Function

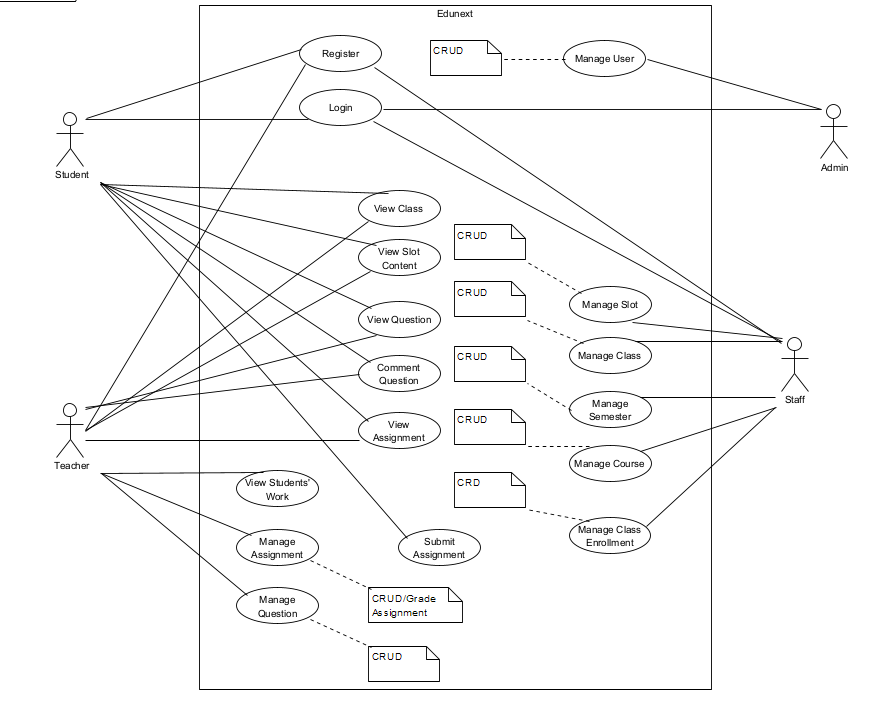
* *Student-1*
* *View Slots Content of a Class*
* *View/Comment Question*
* *View/Submit Assignment*
* *Teacher-2*
* *View Class Enrollment*
* *View Slots Content of a Class*
* *View/Comment/Create/Update/Delete Question*
* *View/Grade/Create/Update/Delete Assignment*
* *Staff-3*
* *View/Create/Update/Delete Course*
* *View/Create/Update/Delete Semester*
* *View/Create/Update/Delete Class*
* *View/Create/Update/Delete Slot*
* *Admin-4*
* *View/Create/Update/Delete User*

*Chú ý login cần vrf user deleted và các bảng khác: login, enrollments*

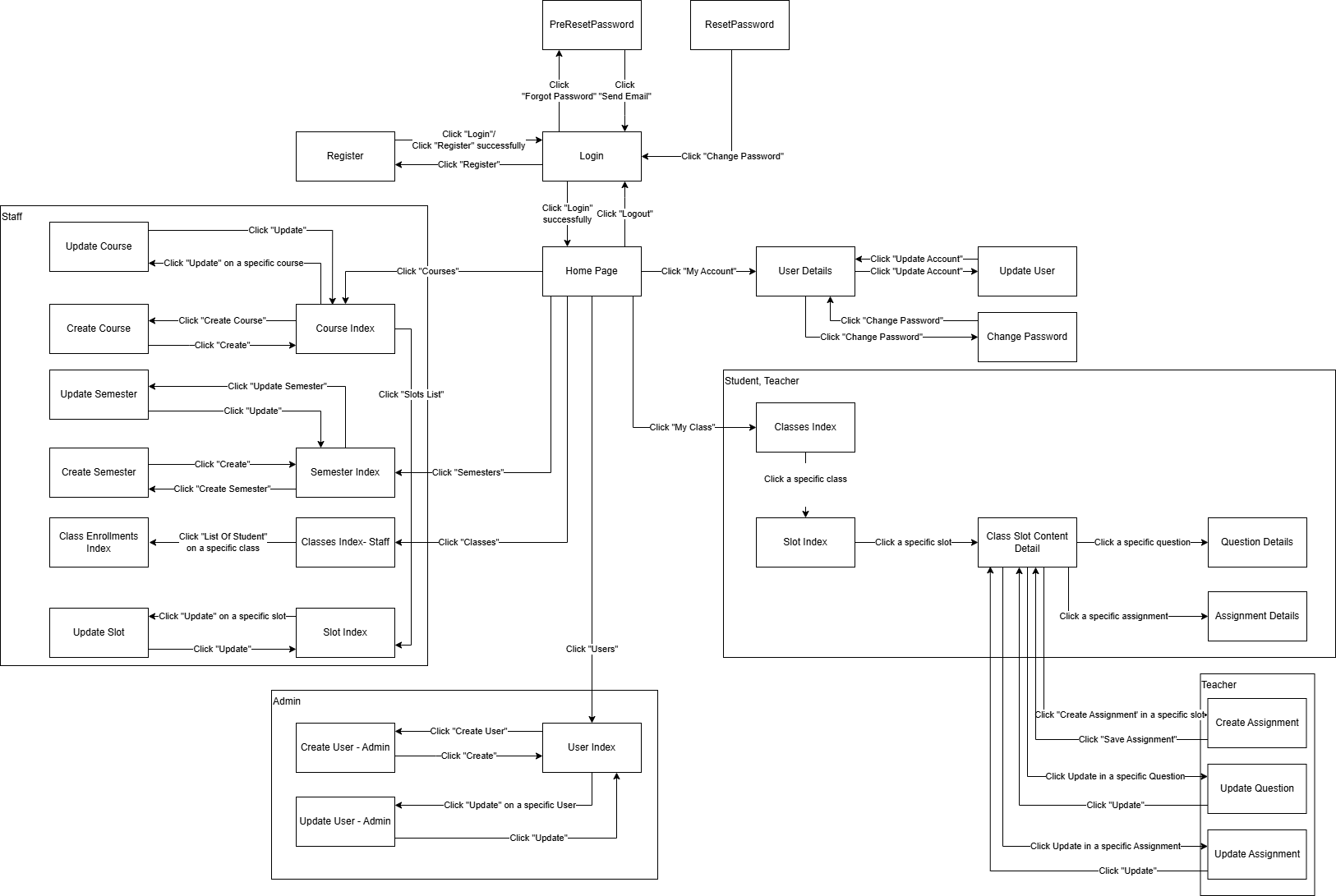
*enrollment check ton tai neu isdeleted->update thanh true*

*class teacher cung can active*

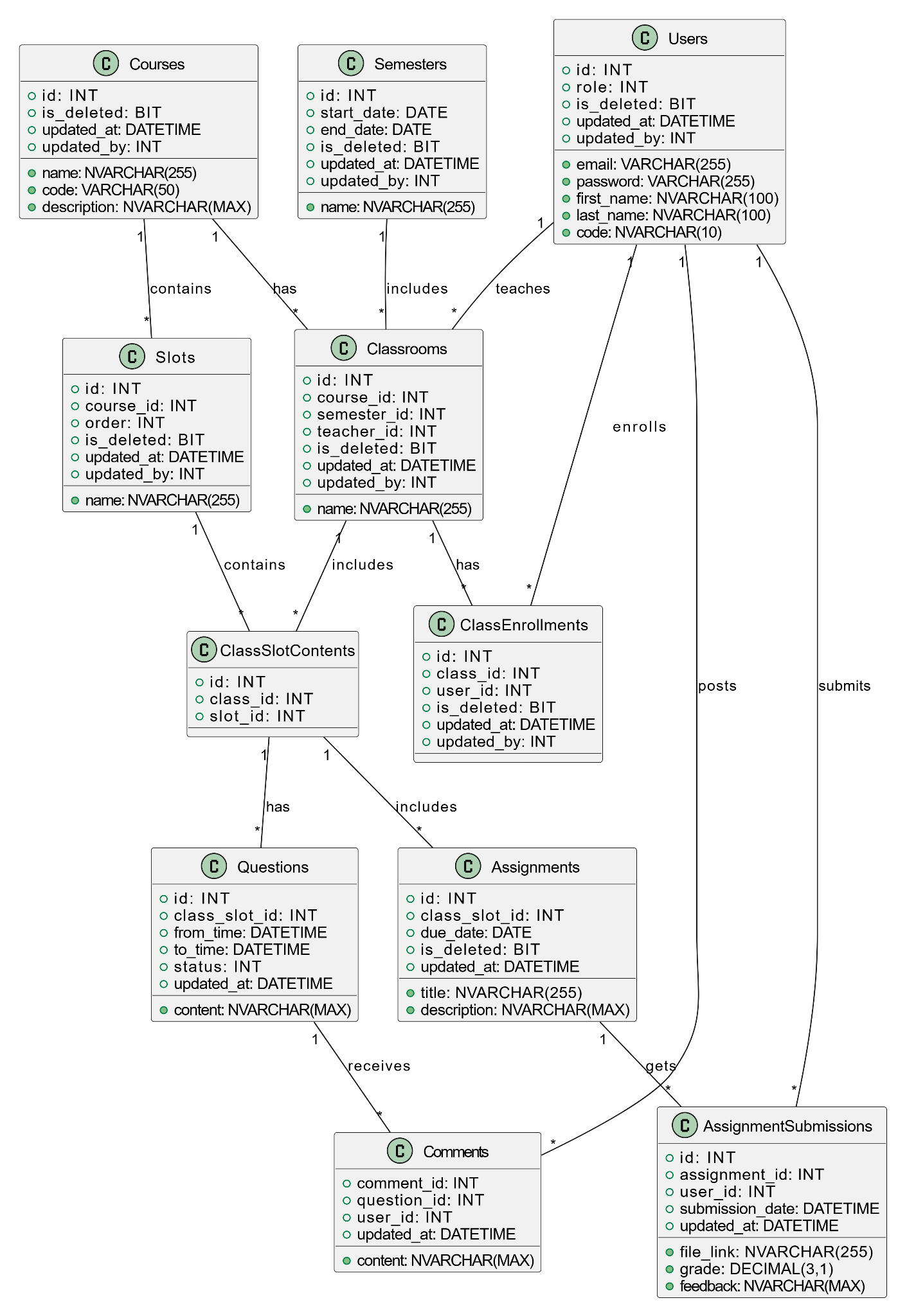
## Use Case Diagram



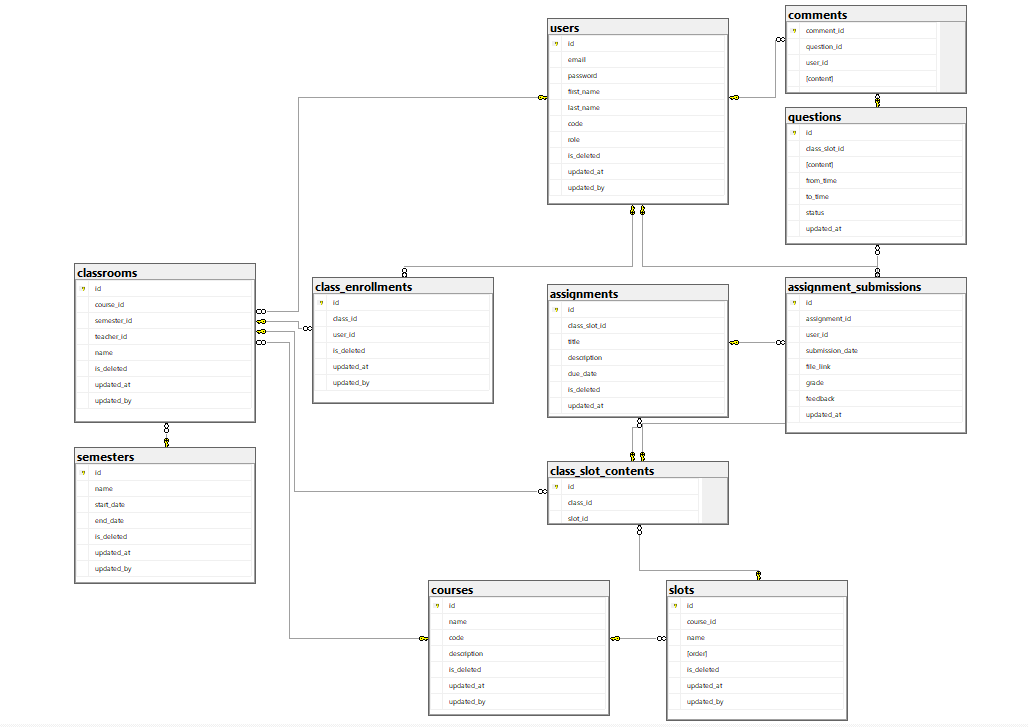
## Screen Flow



## Class Diagram

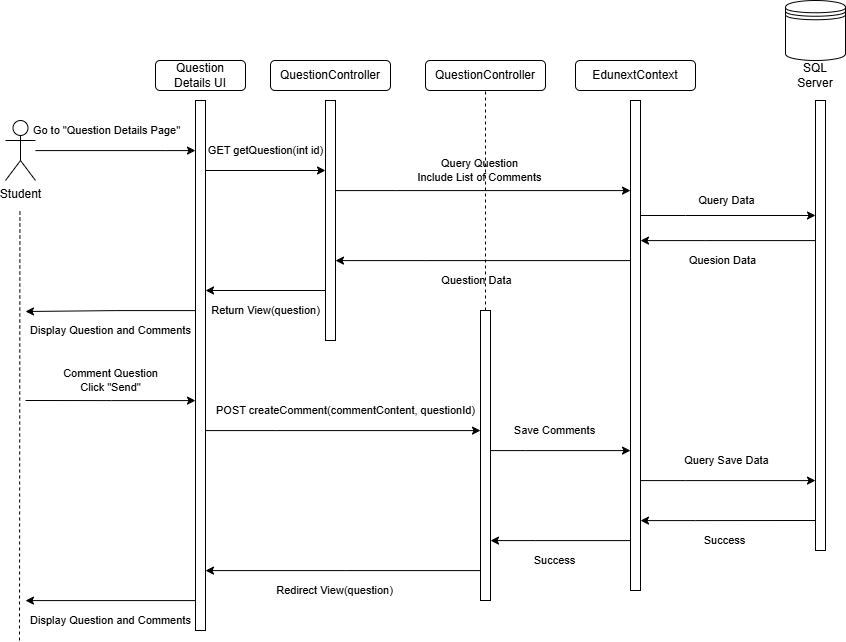


## Entity Relationship Diagram



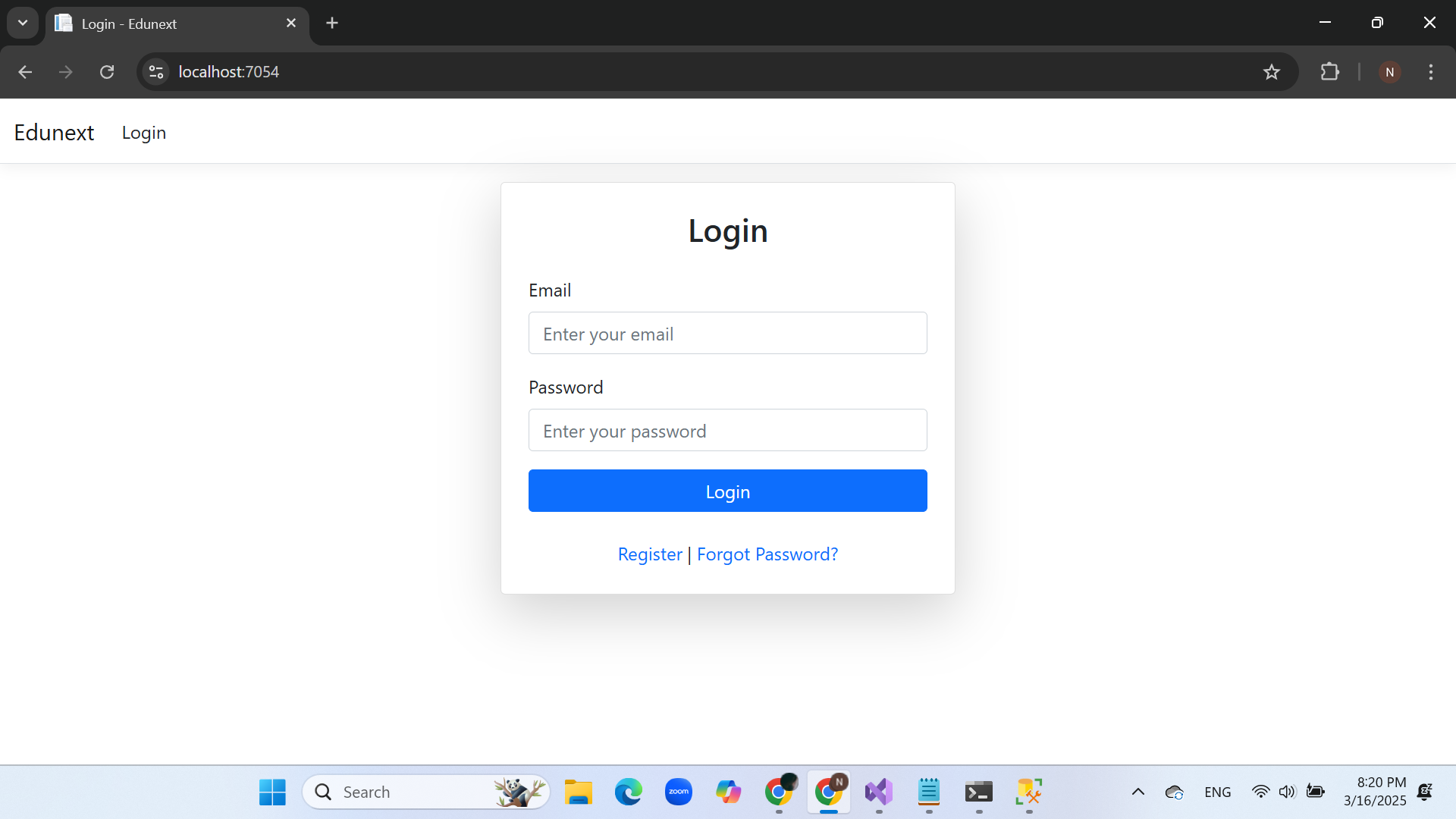
## Sequence diagram

* Comment Question

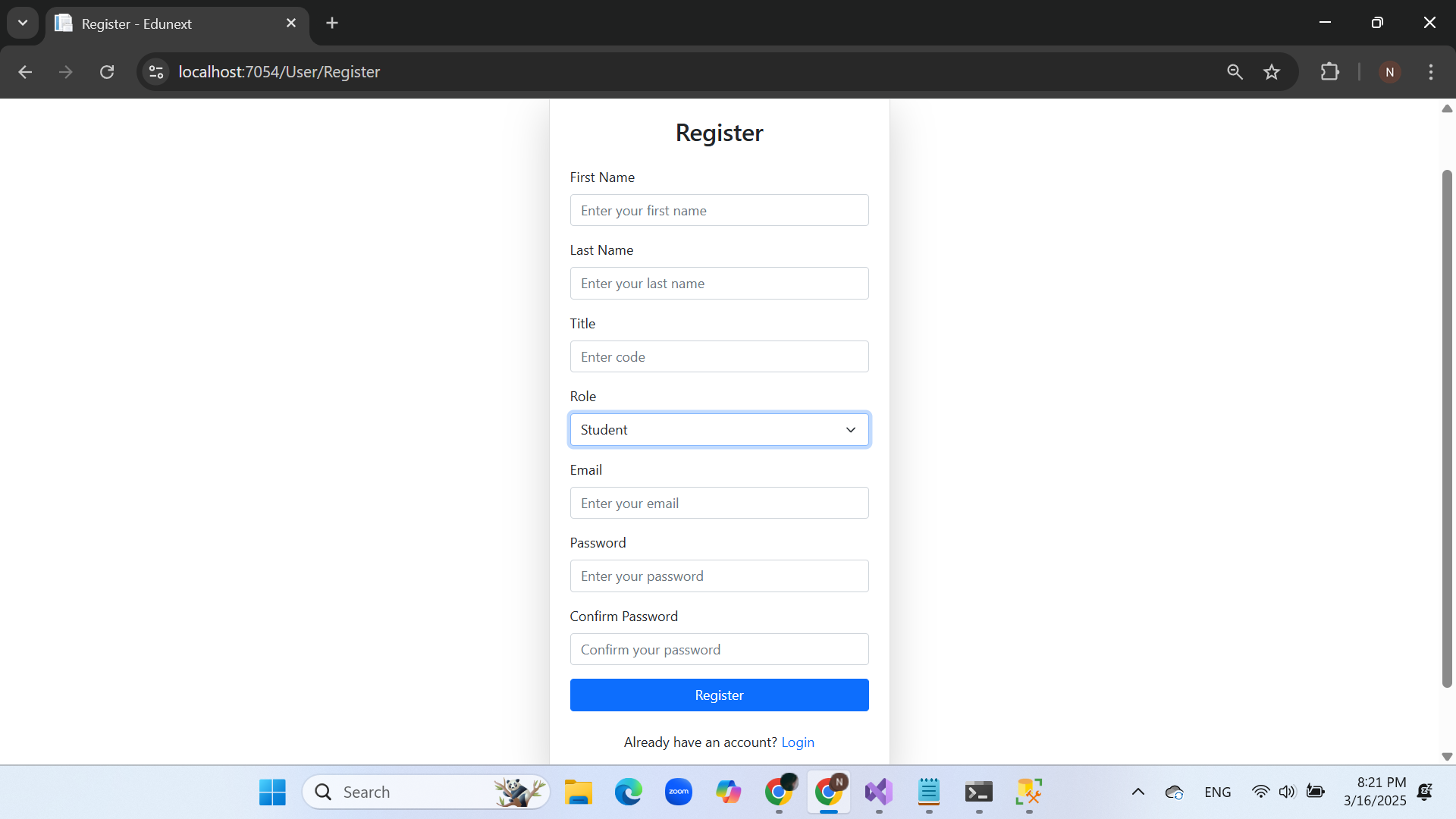


# Website Design

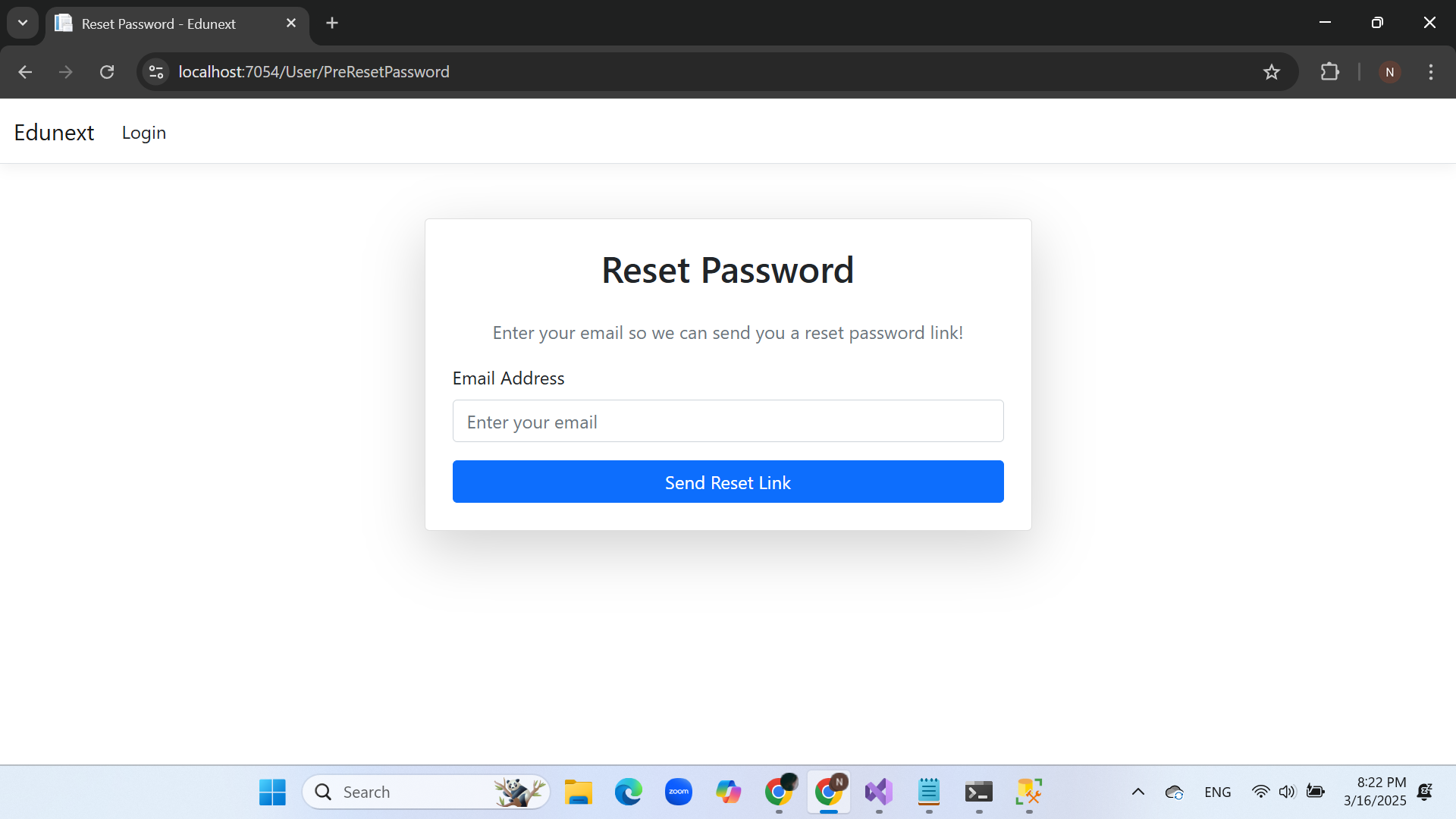
## Login interface



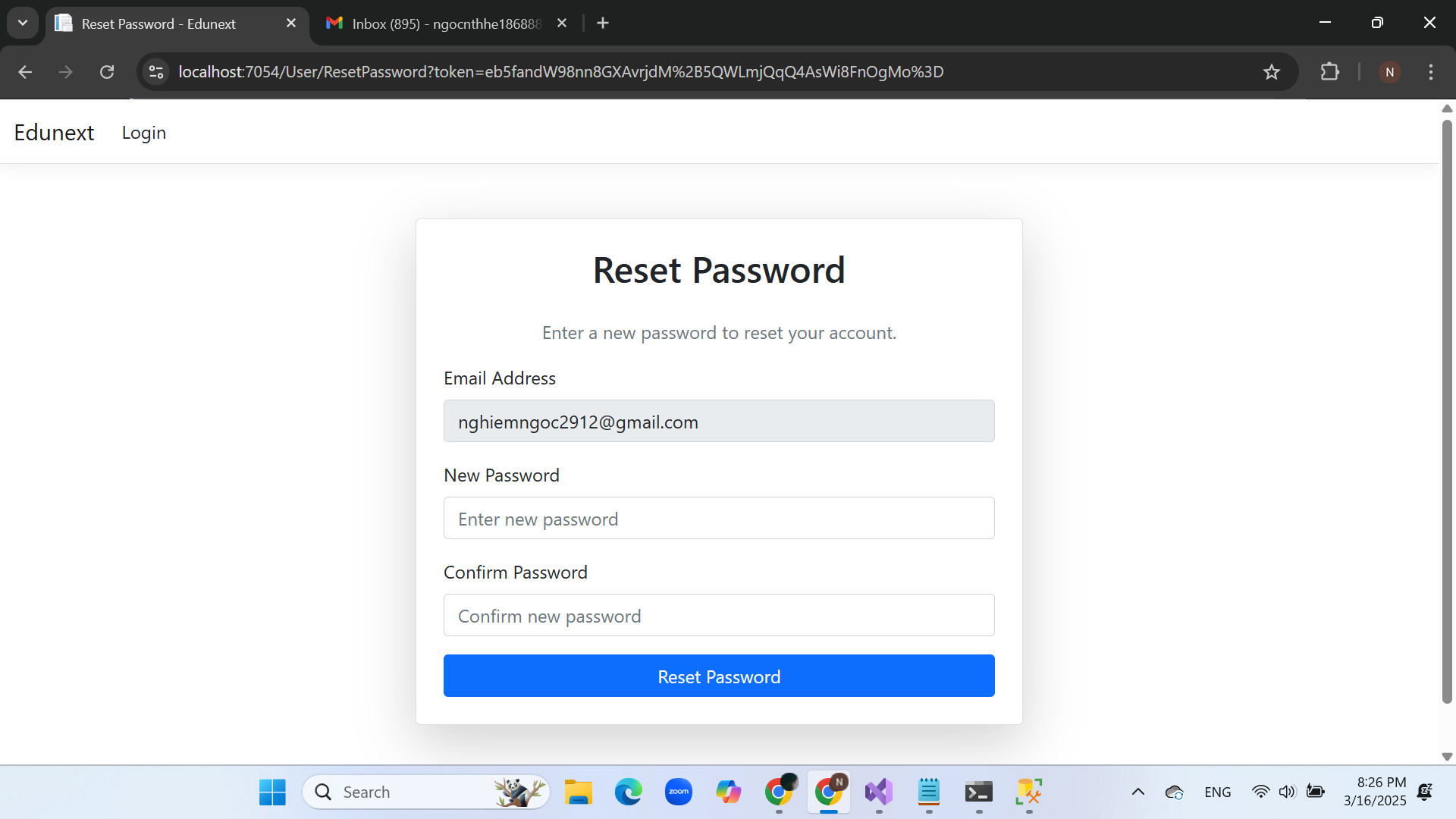
## Register interface



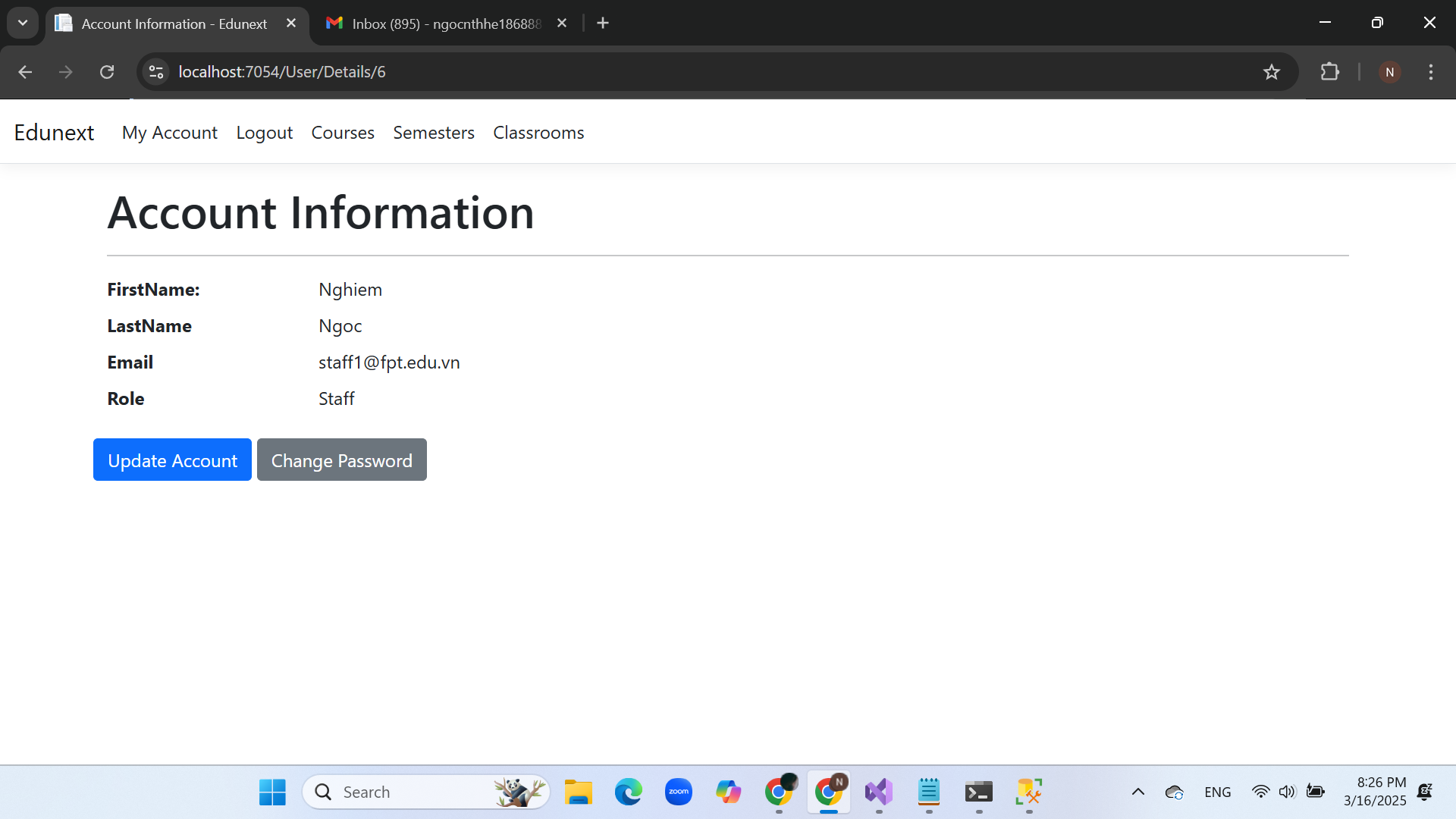
## PreResetPassword interface



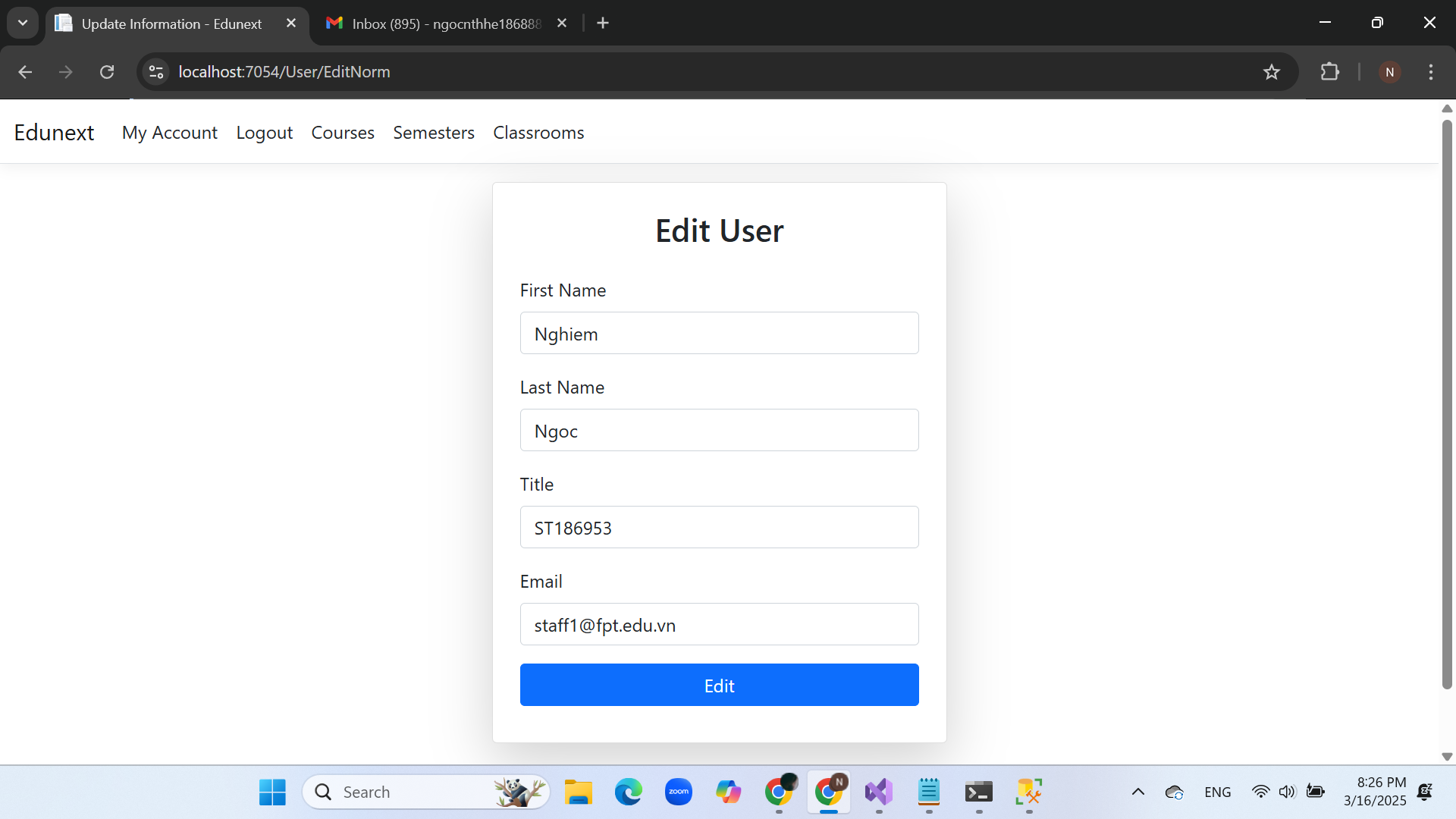
## ResetPassword interface

****

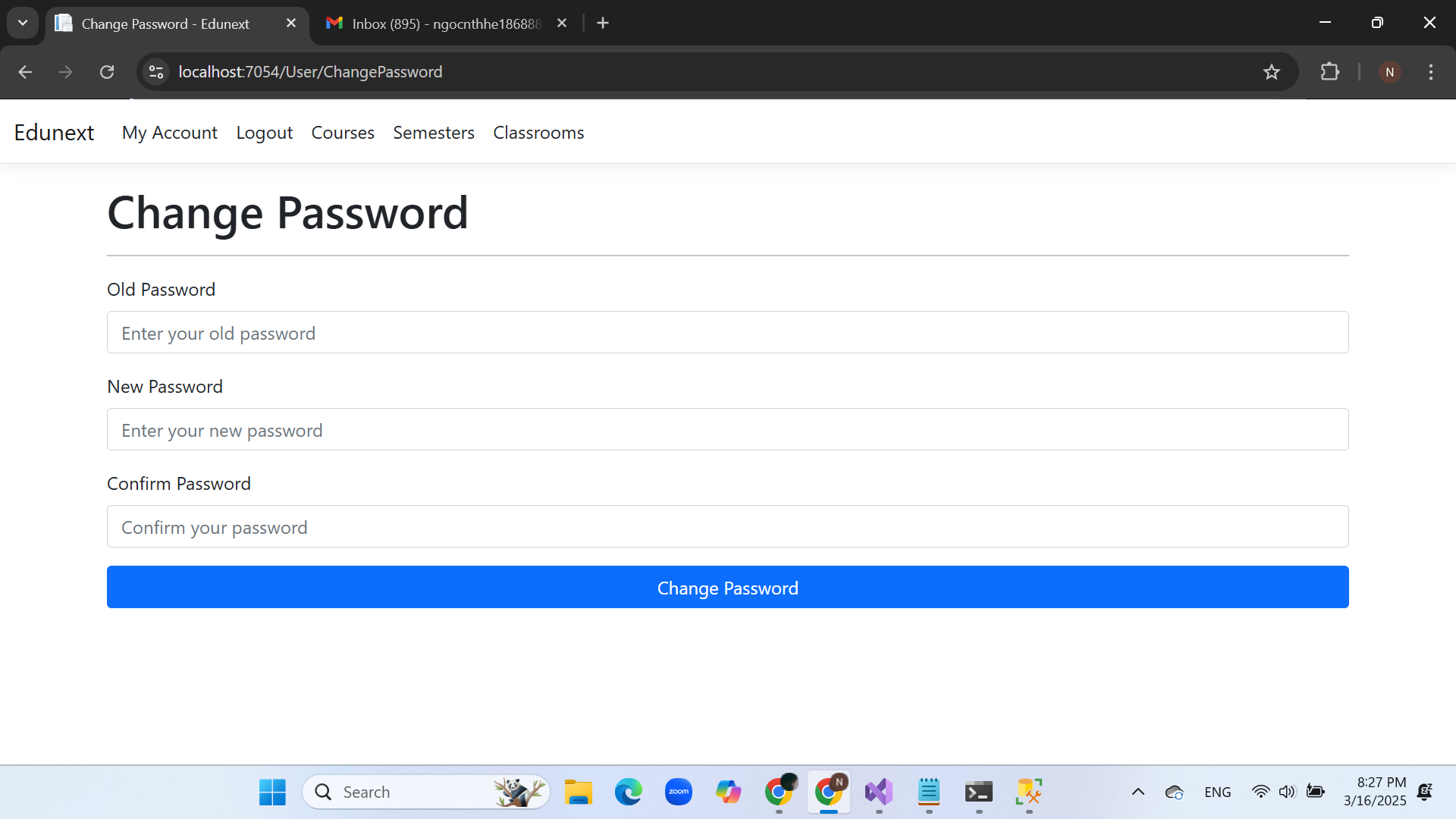
## User Details interface



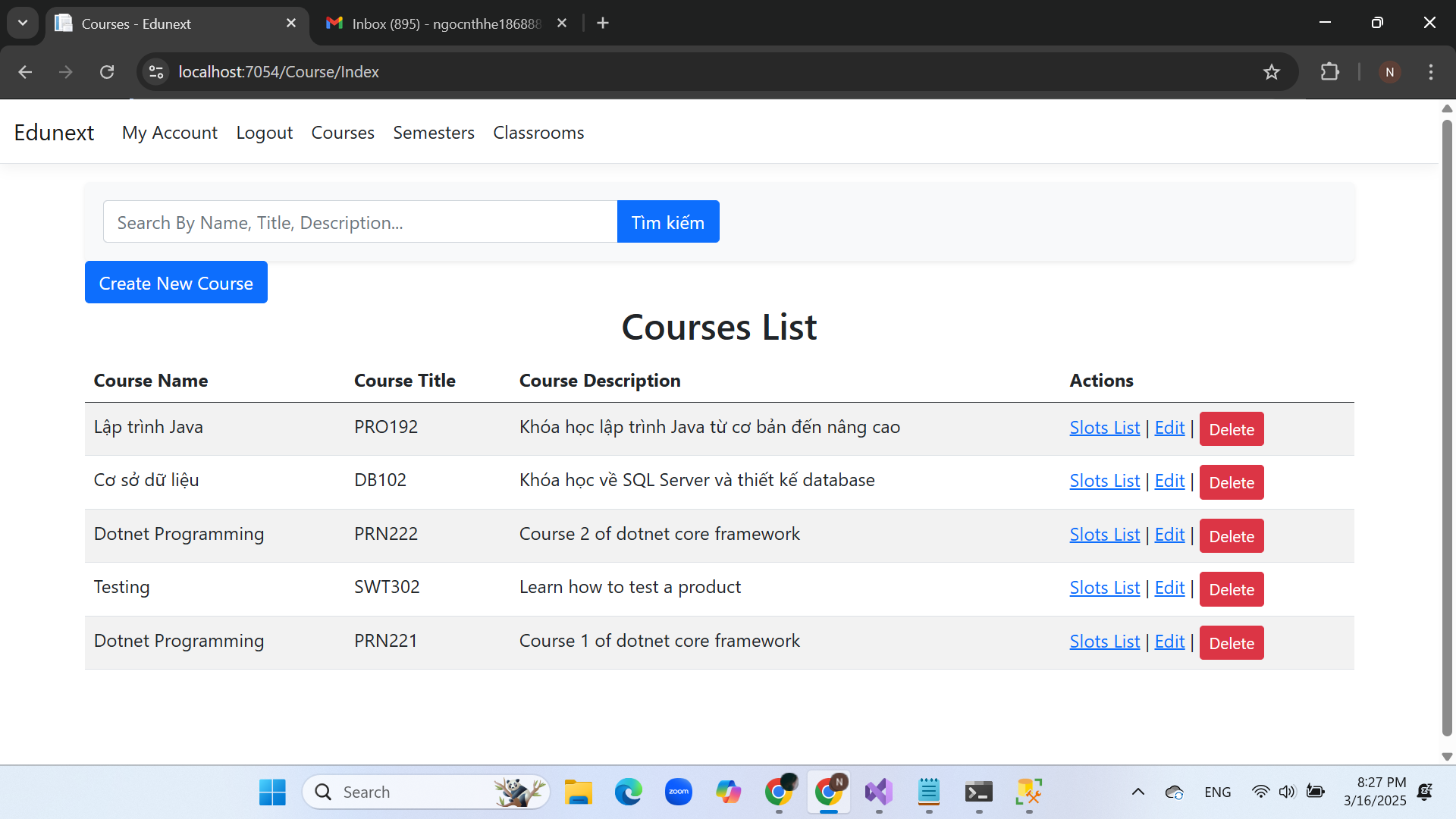
## Update User interface



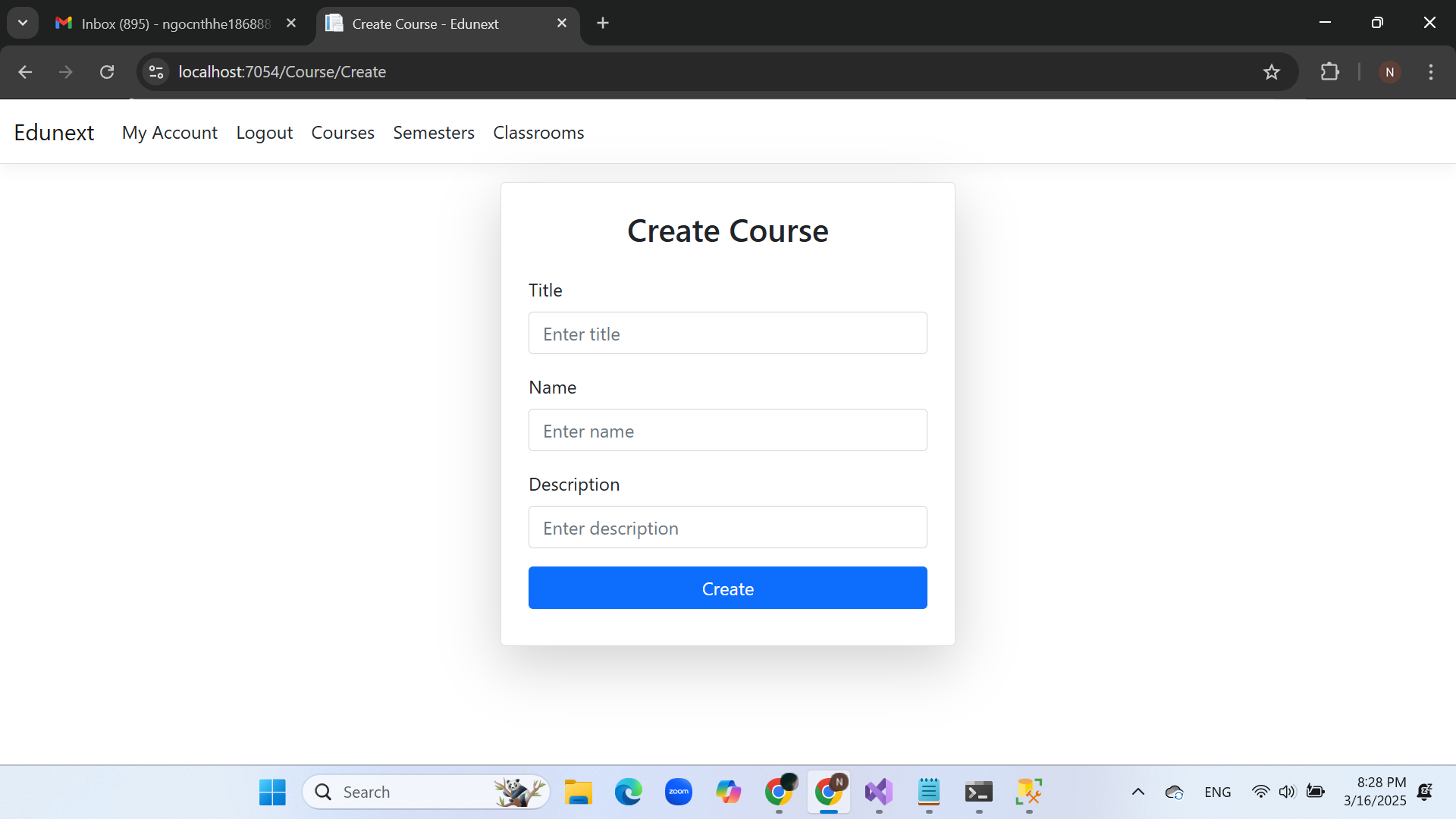
## Change Password interface



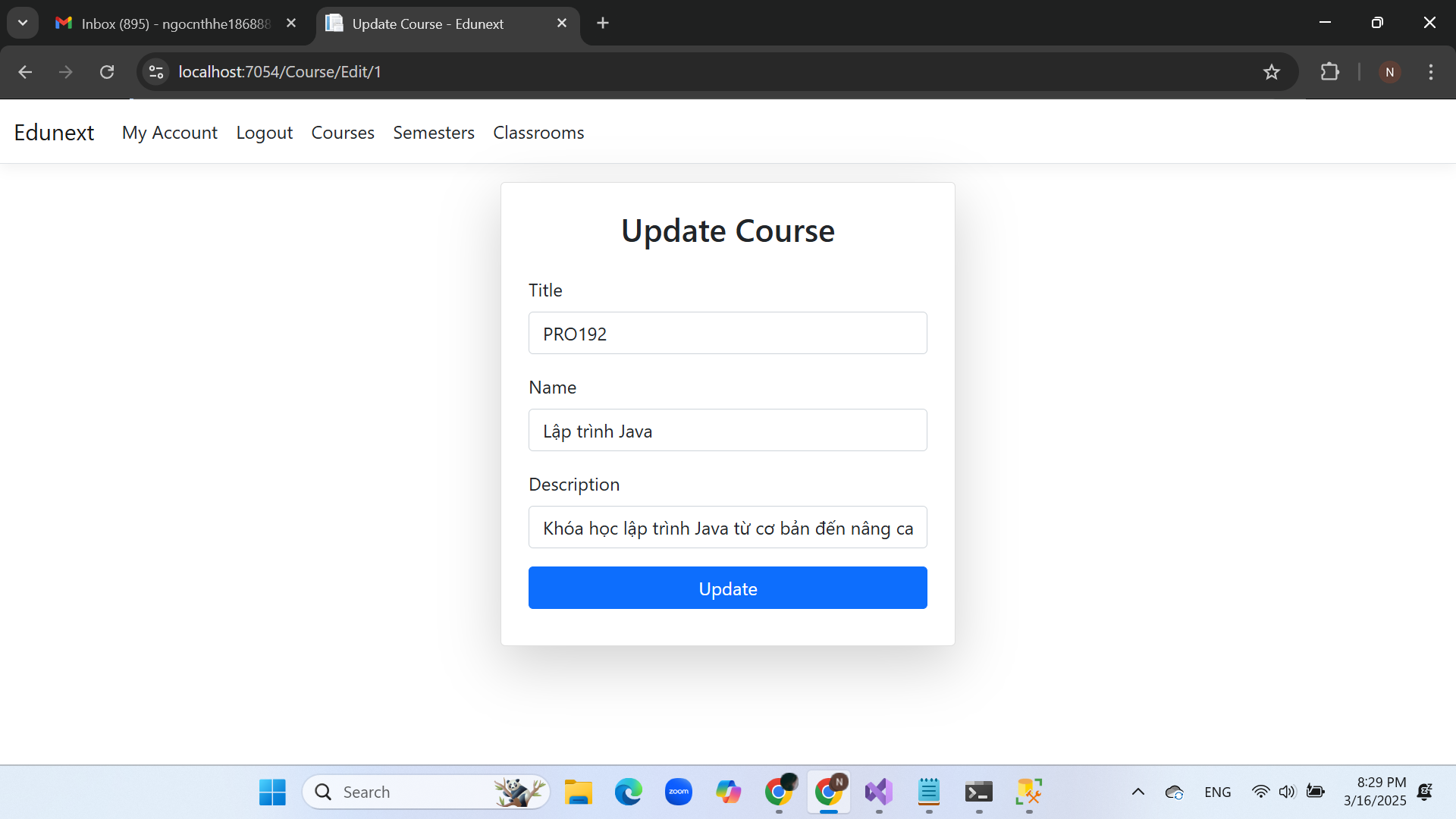
## Course Index interface



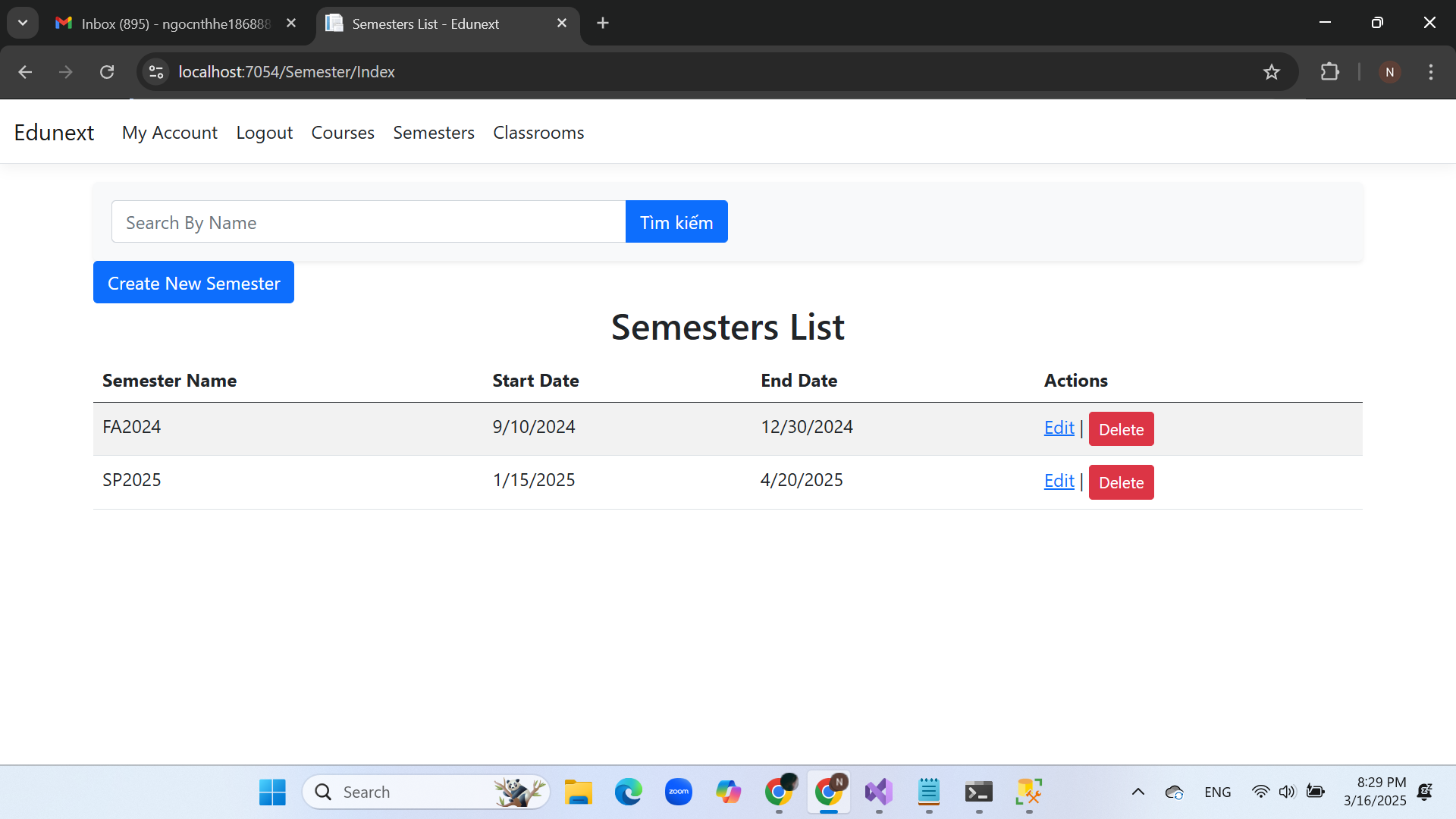
## Create Course interface



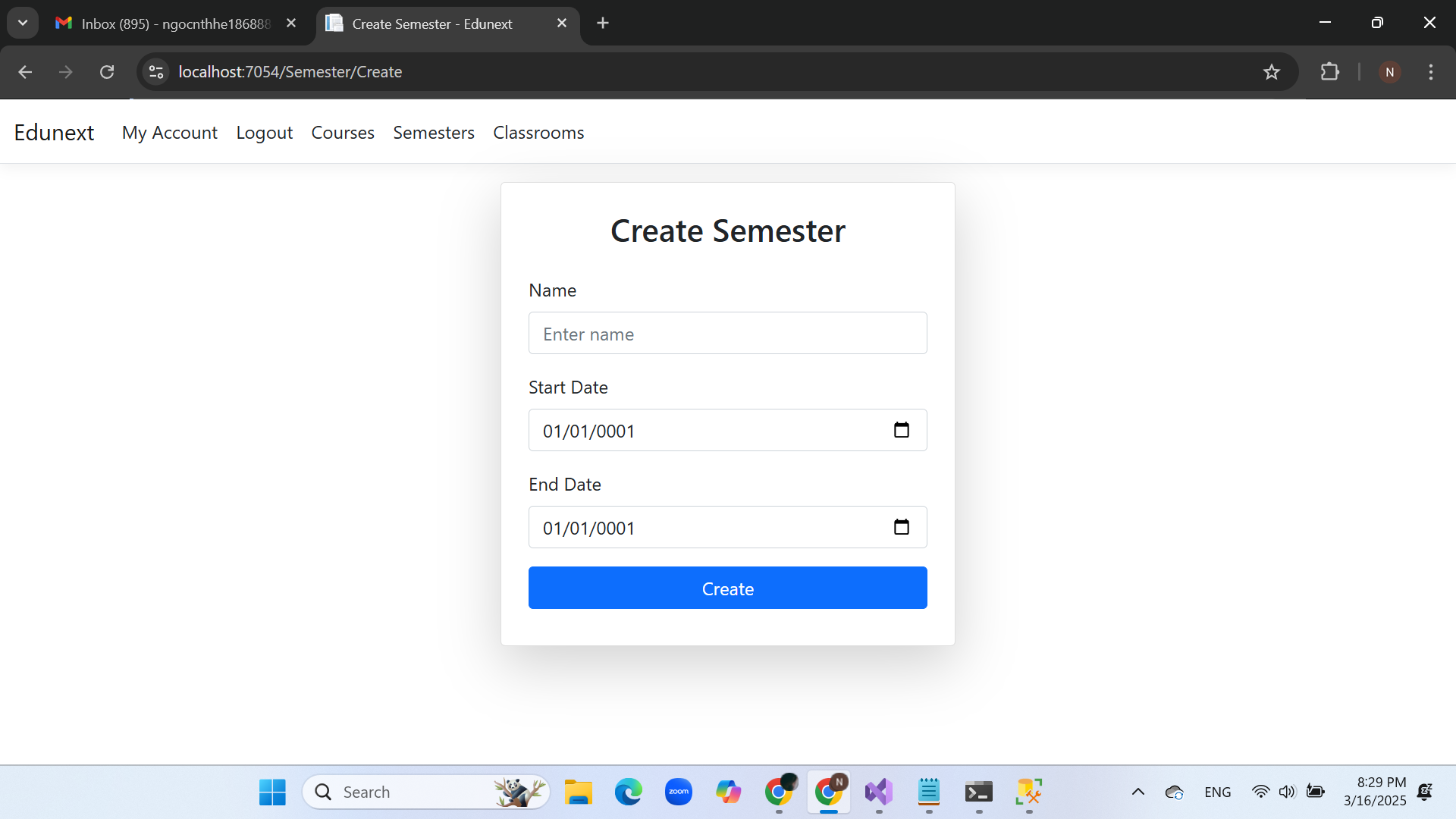
## Update Course interface



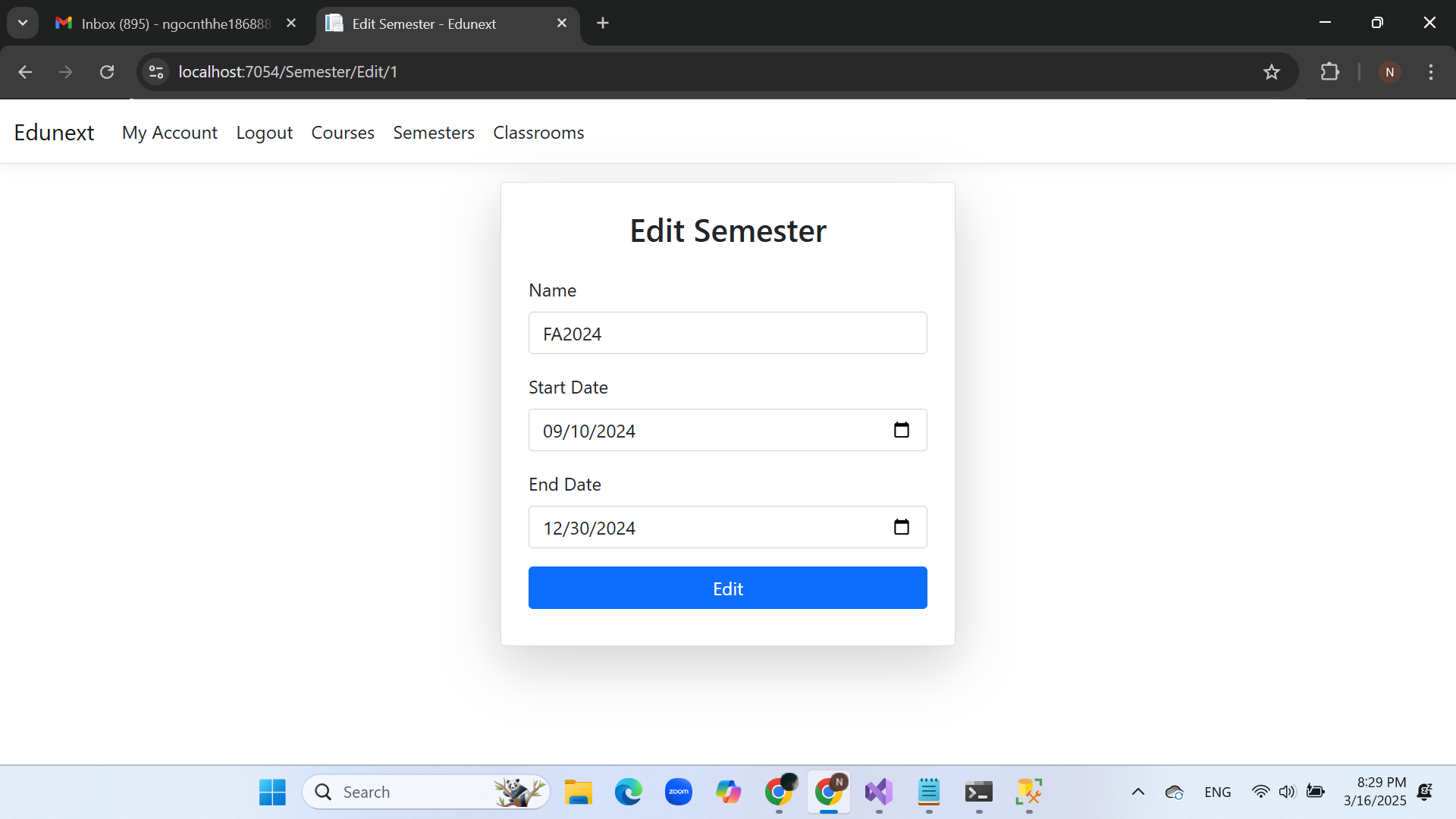
## Semester Index interface



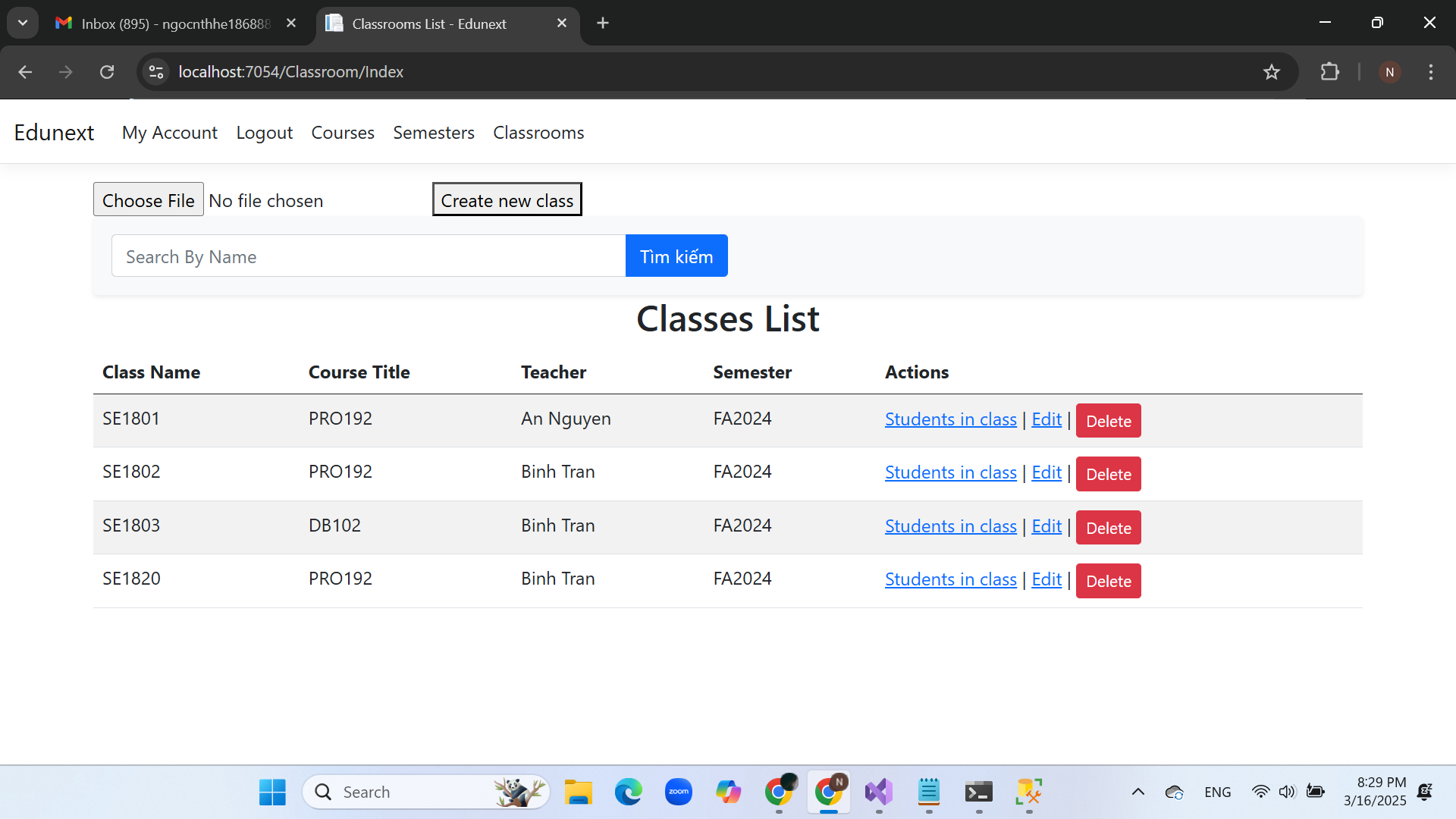
## Create Semester interface



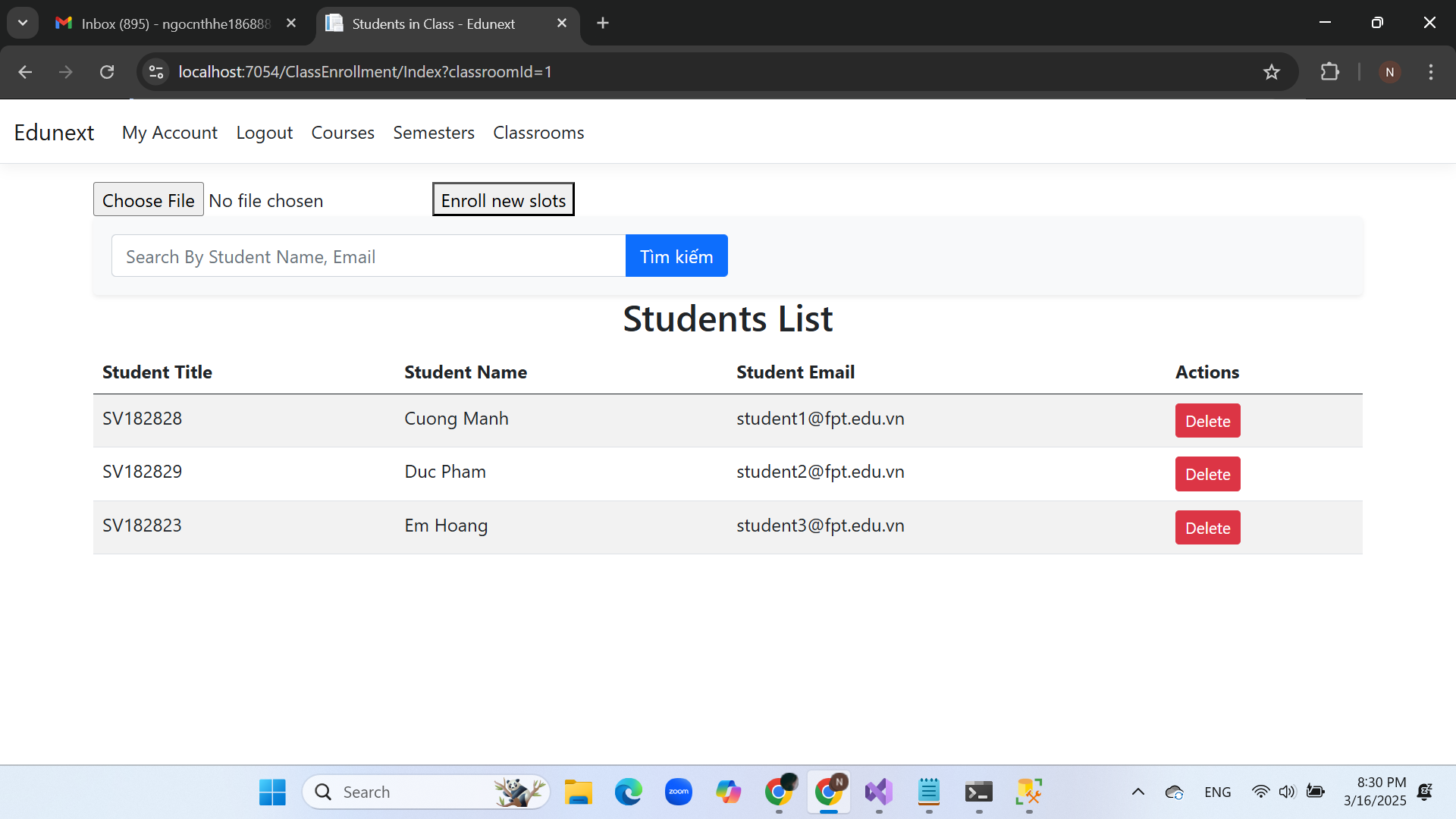
## Update Semester interface



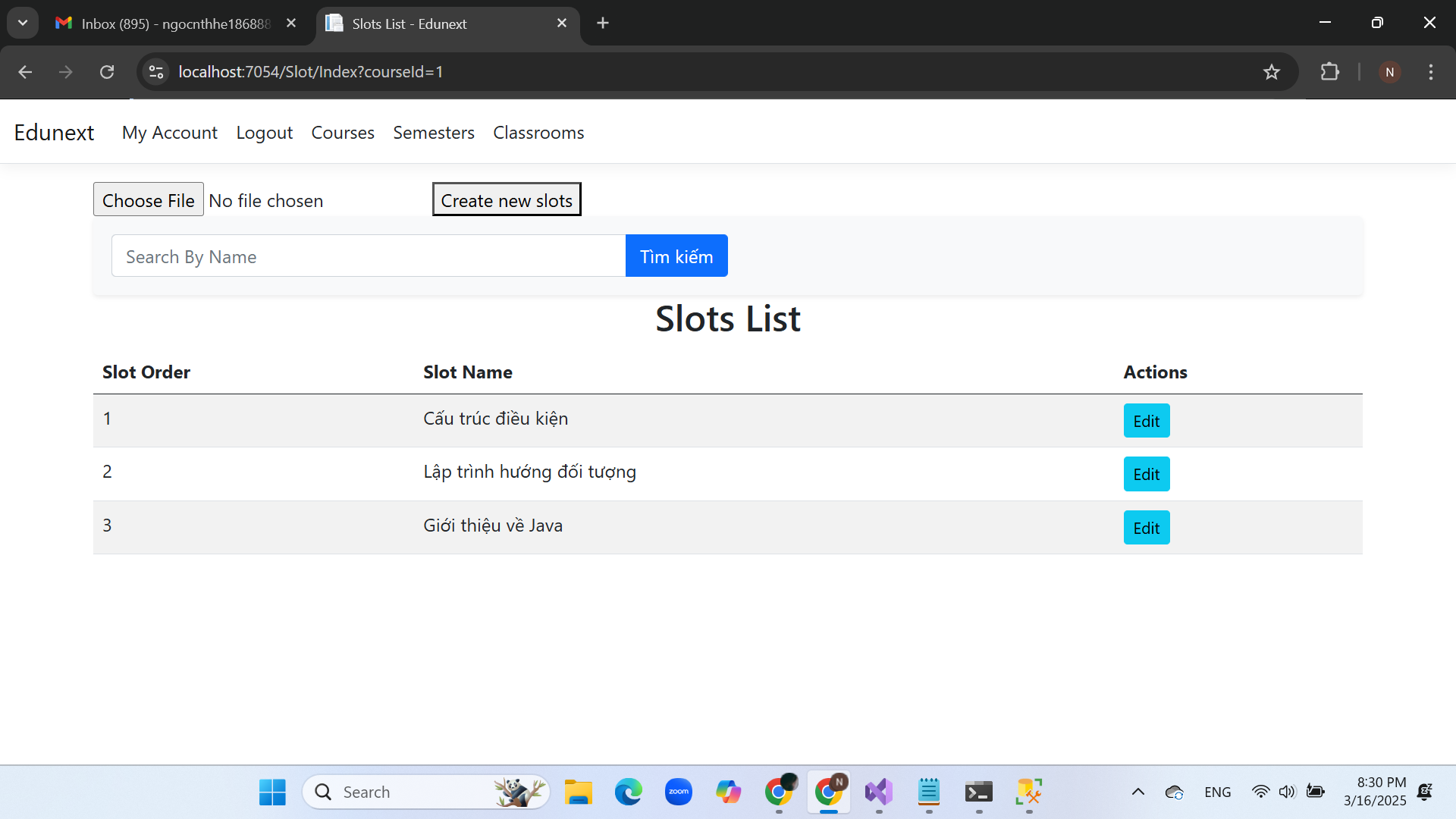
## Classes Index interface



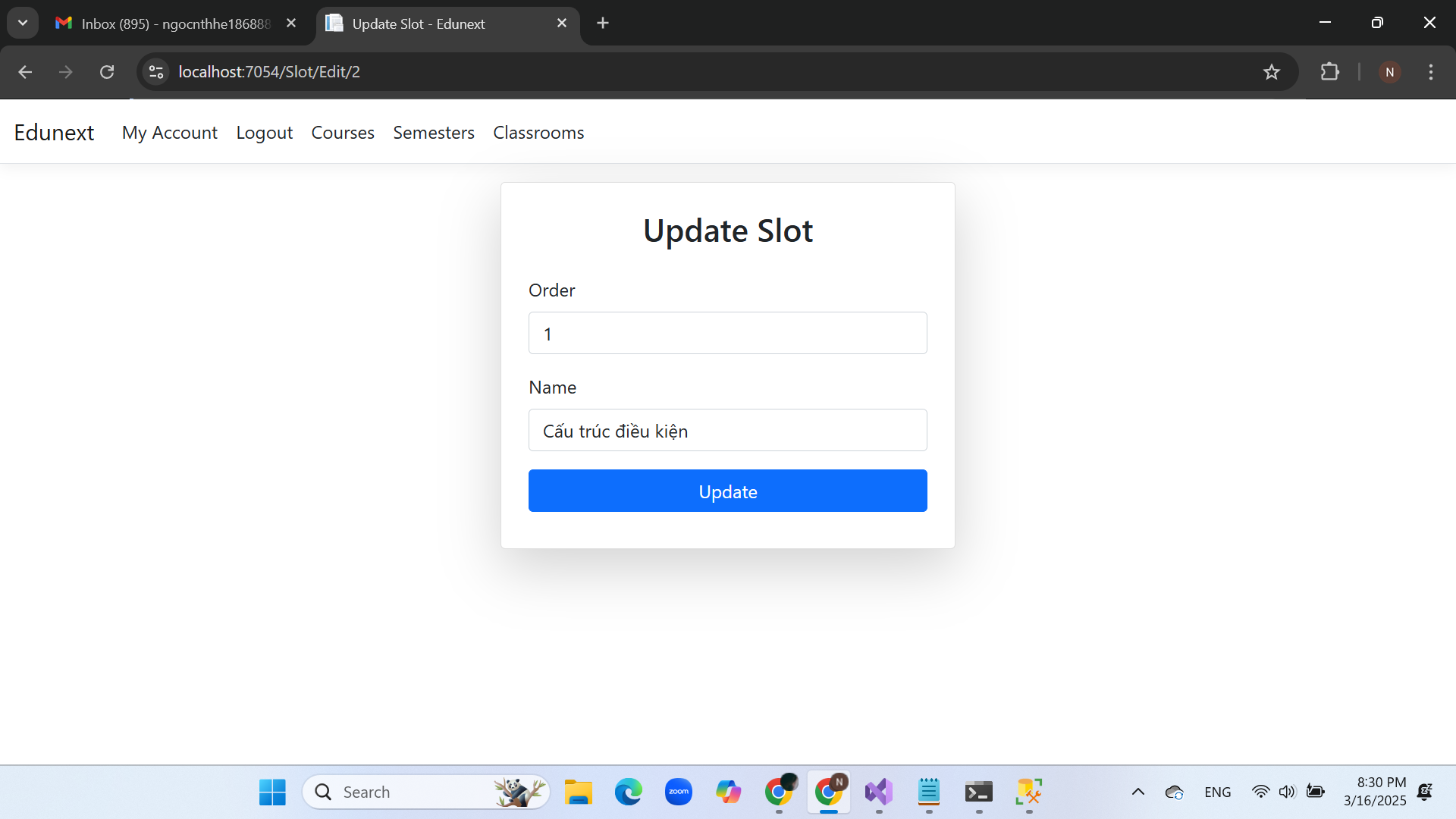
## Class Enrollments Index interface



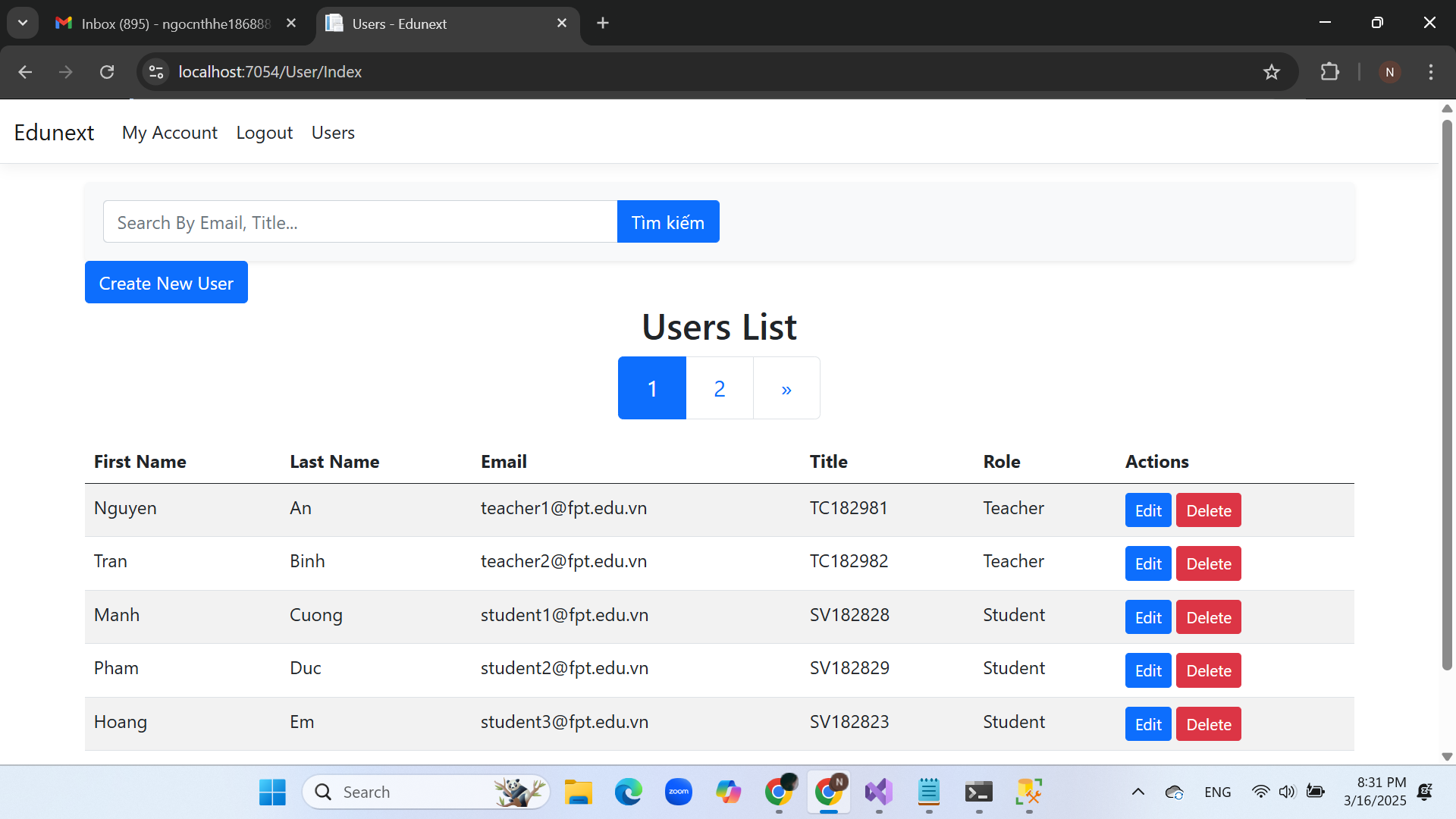
## Slot Index interface



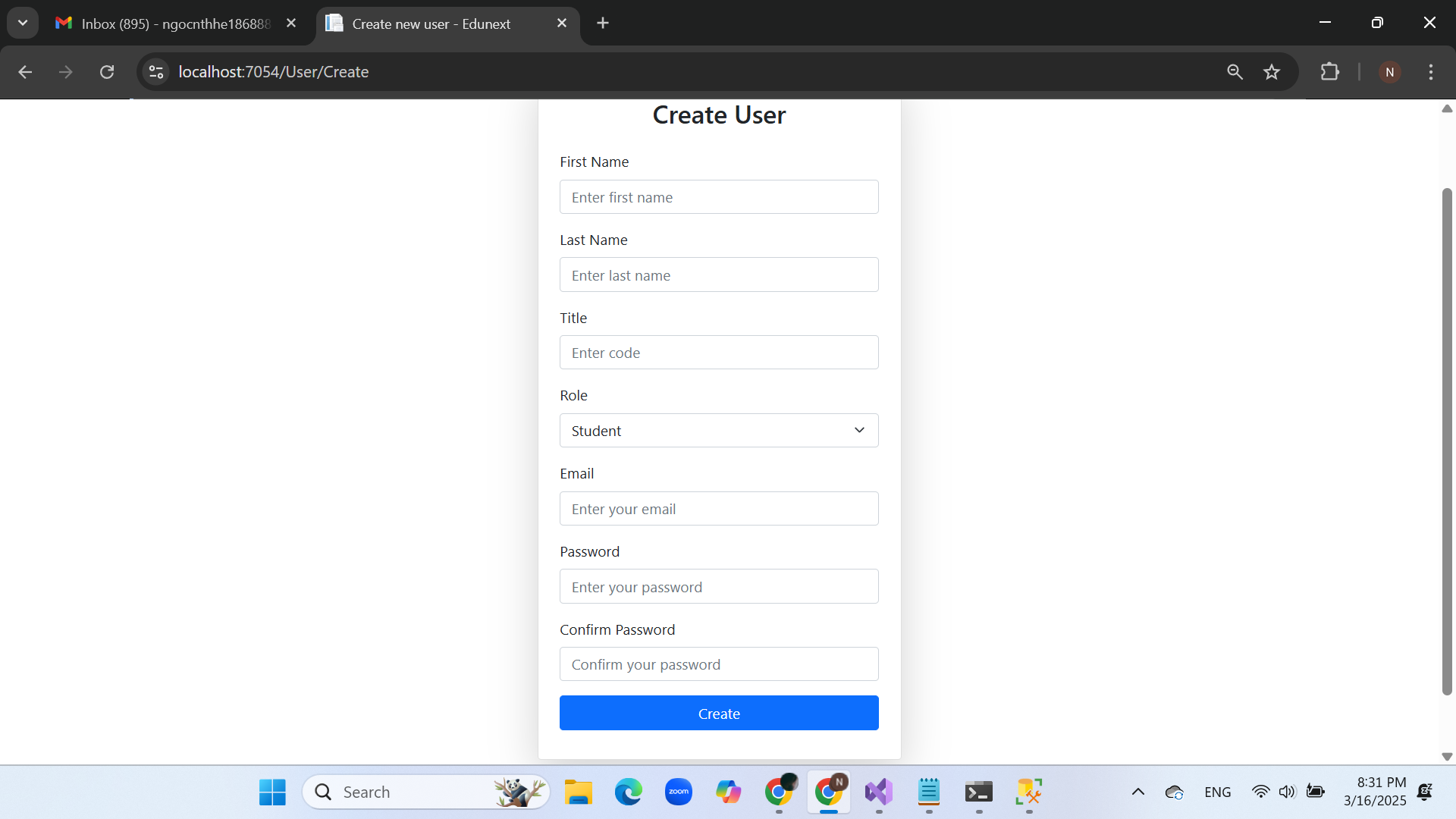
## Update Slot interface



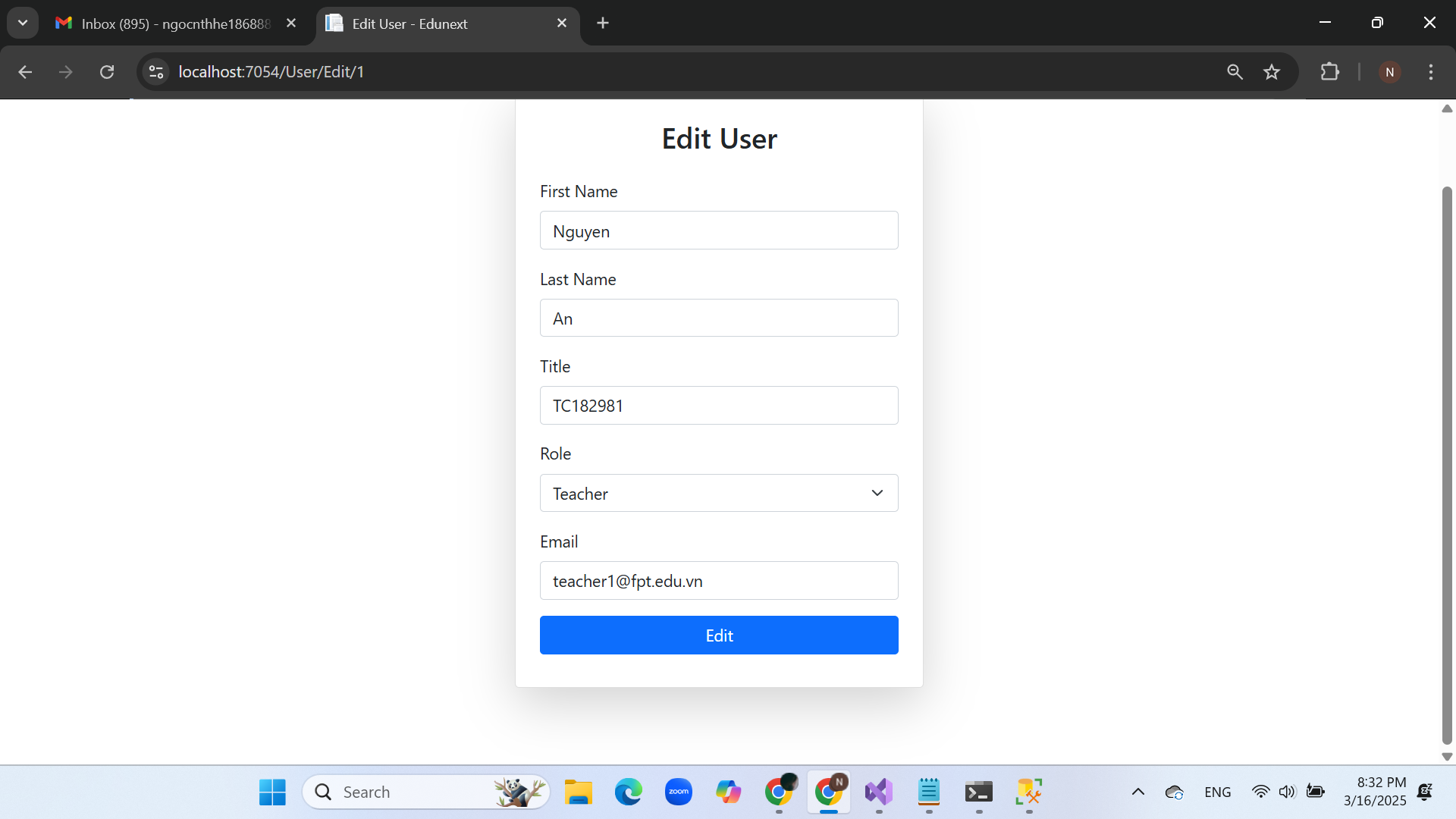
## User Index interface



## Create User interface



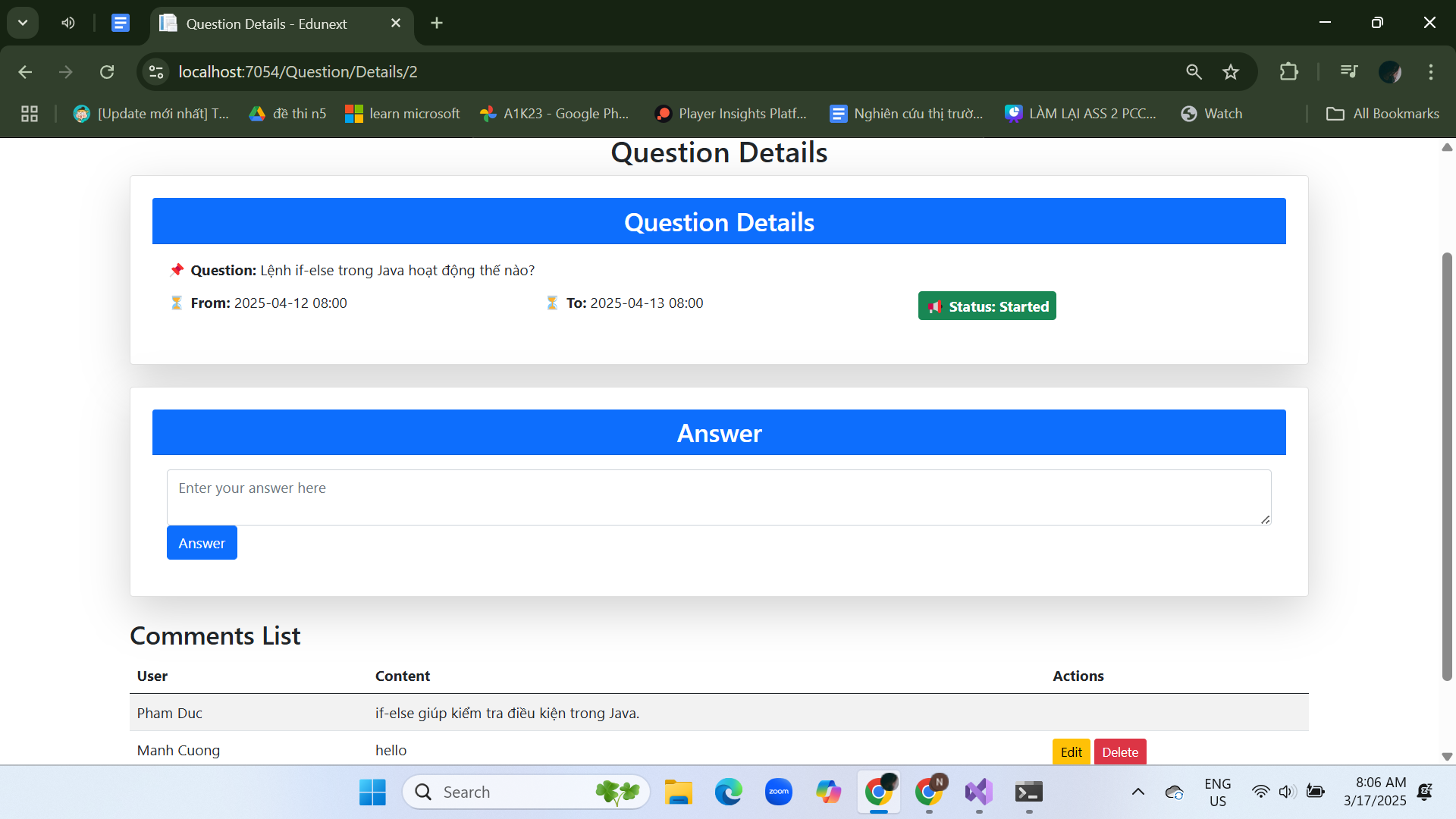
## Update User interface



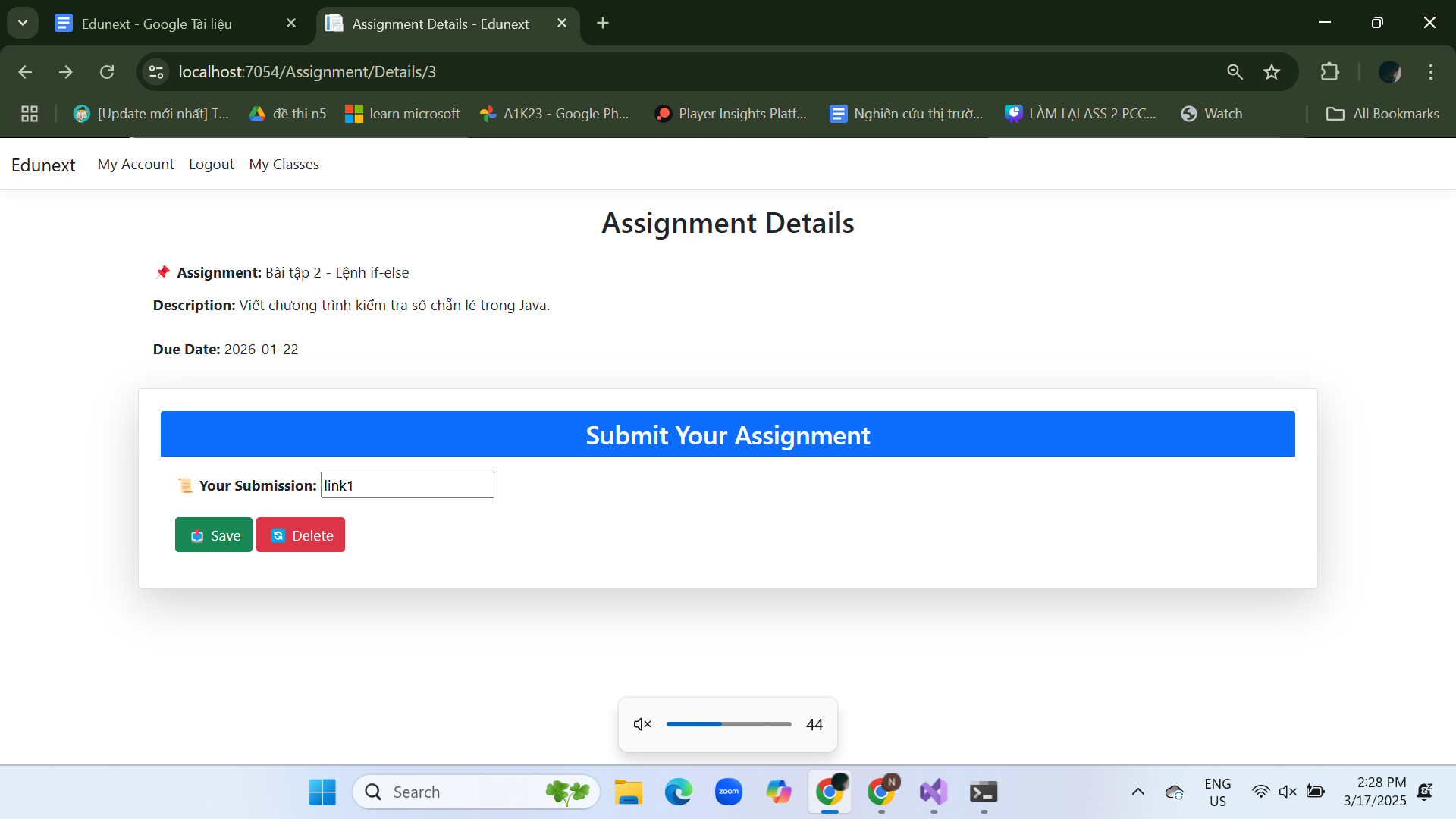
## Class Slot Content Detail interface



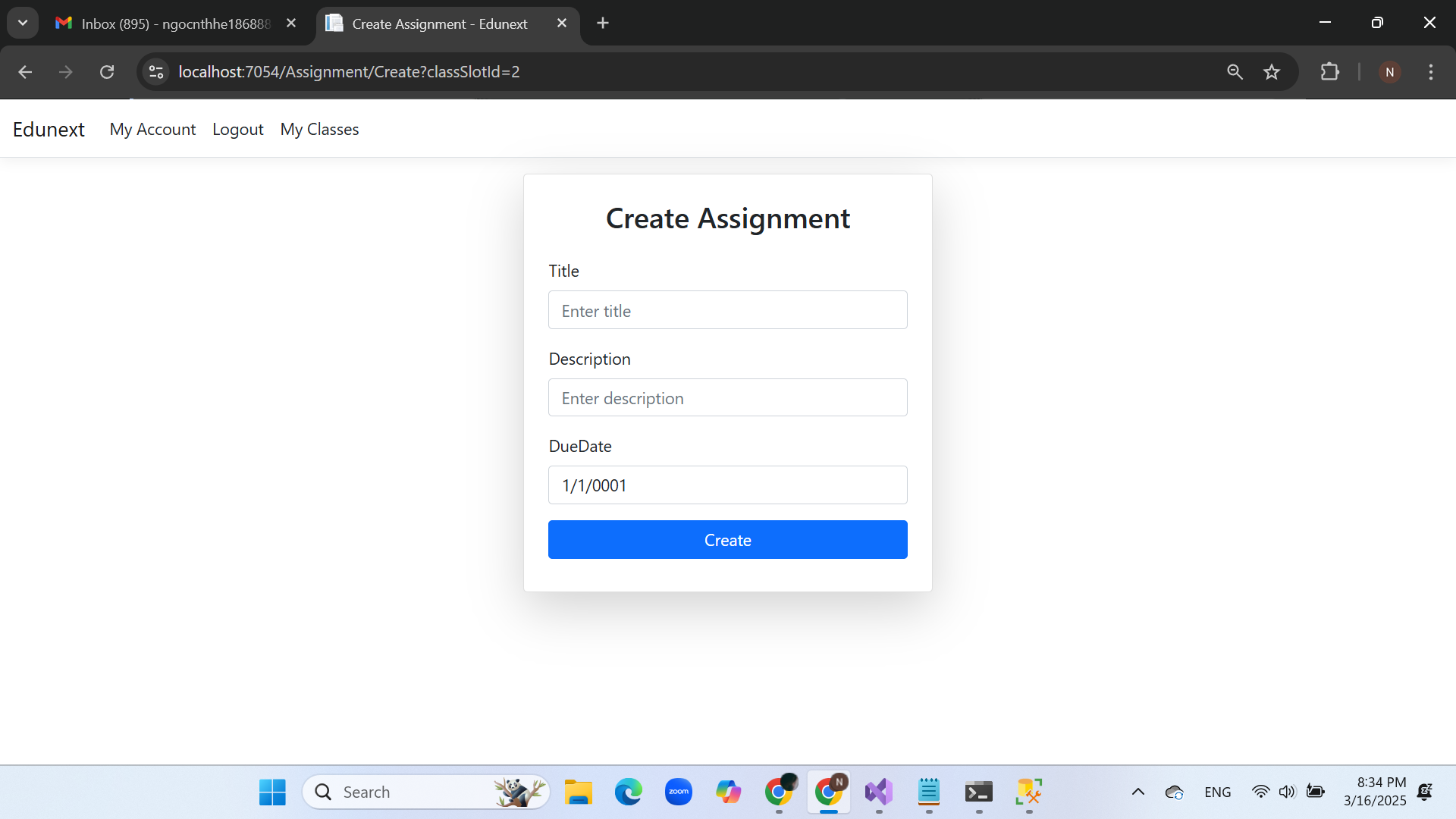
## Question Details Interface



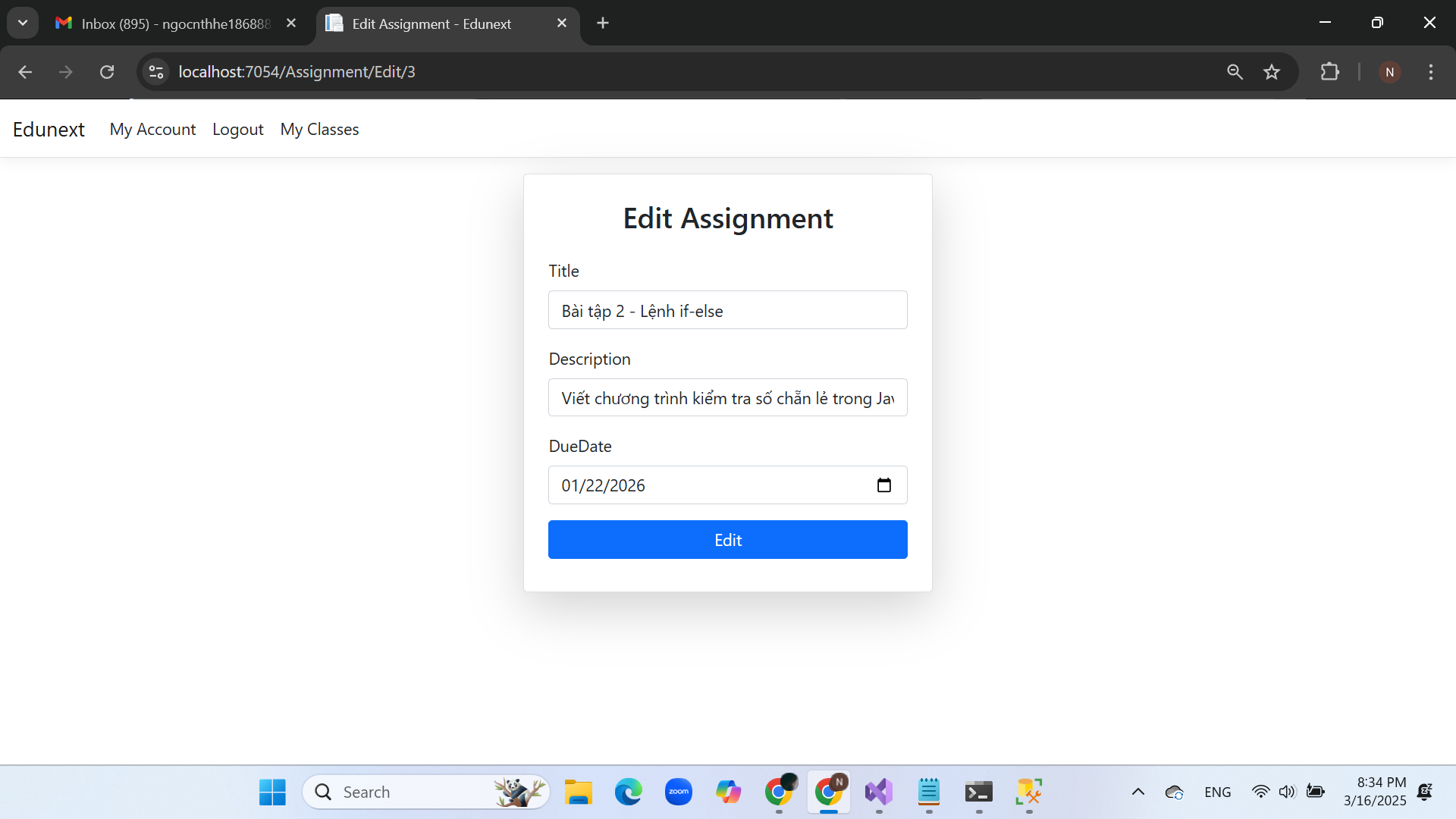
## Assignment Details interface



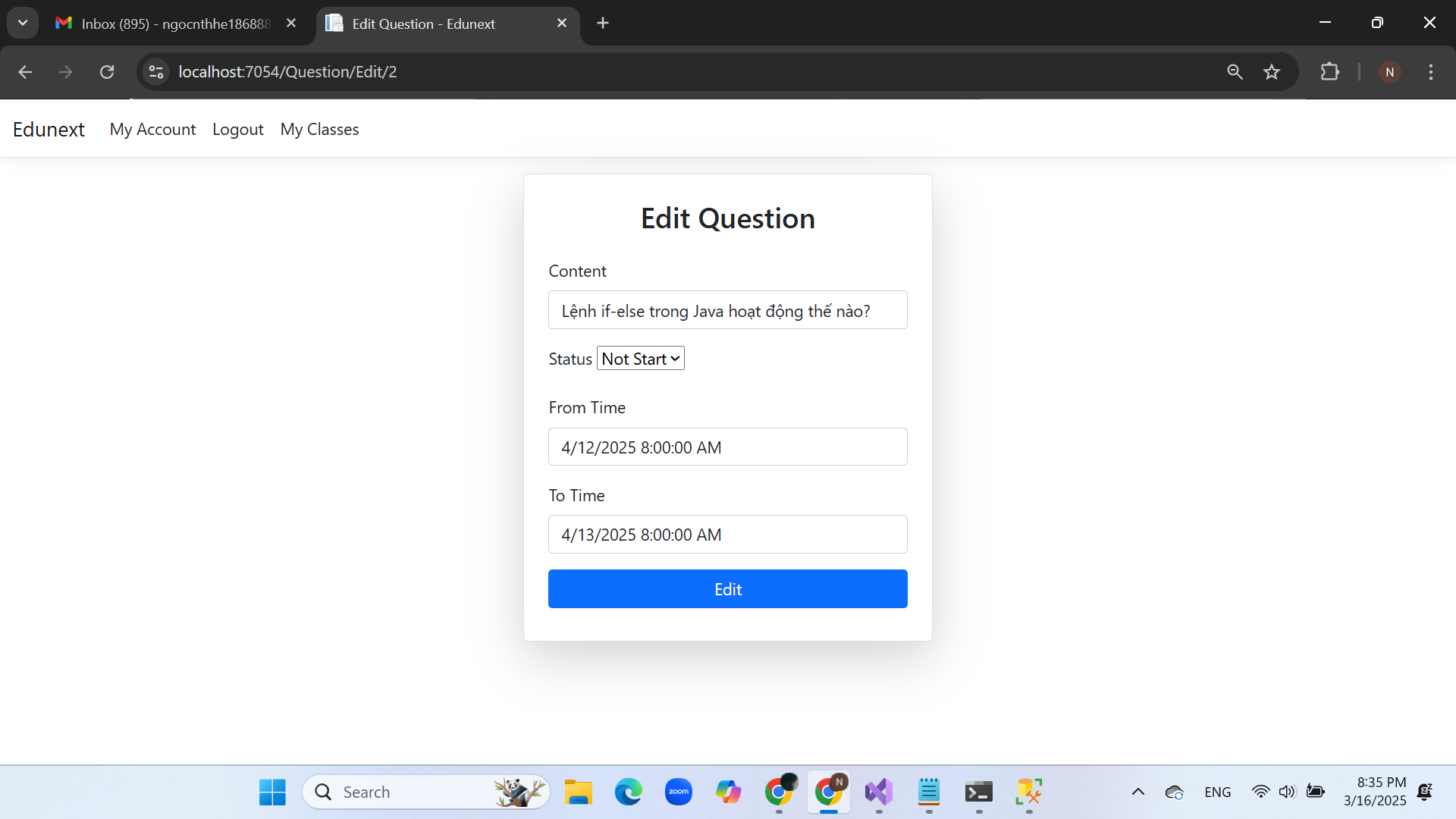
## Create Assignment interface



## Update Assignment interface



## Update Question interface



# Conclusion

**- Achievements:**

* The system partially meets the needs of digitalizing the e-learning process, providing a platform for students and teachers to engage in online learning effectively. This enhances accessibility, flexibility, and convenience for both parties while maintaining the quality of education.
* The system has a simple and modern interface, making it user-friendly and easy to navigate for both students and instructors.

**- Limitations:**

* The system still has several shortcomings in both functionality and interface, which may require further improvements to optimize the user experience and enhance learning effectiveness.