

**Mentor Appointment Scheduling Application**

**Software Requirement Specification**

– Ho Chi Minh, May 2022 –

**Table of Contents**

[I. Overview 3](#_heading=h.gjdgxs)

[1. Introduction 3](#_heading=h.30j0zll)

[2. System Functions 4](#_heading=h.1fob9te)

[3. Common Requirements 5](#_heading=h.3znysh7)

[II. Functional Requirements 6](#_heading=h.2et92p0)

[1. <<Feature Name 1>> 6](#_heading=h.tyjcwt)

[a. <<Function Name 1>> 6](#_heading=h.3dy6vkm)

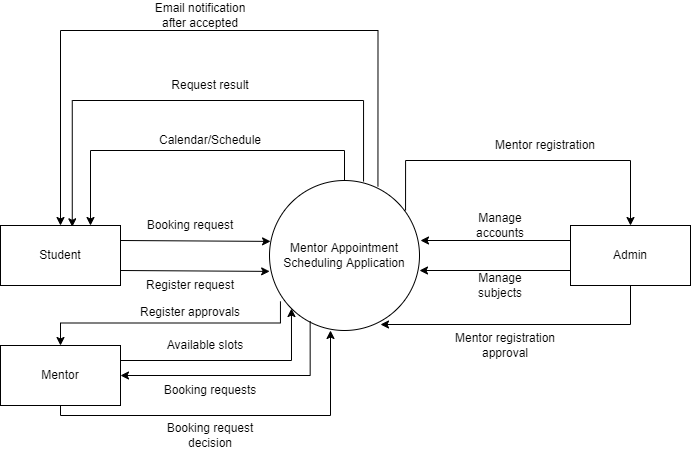
[b. <<Function Name 2>> 6](#_heading=h.1t3h5sf)

[2. <<Feature Name 2>> 6](#_heading=h.4d34og8)

# I. Overview

## 1. Introduction

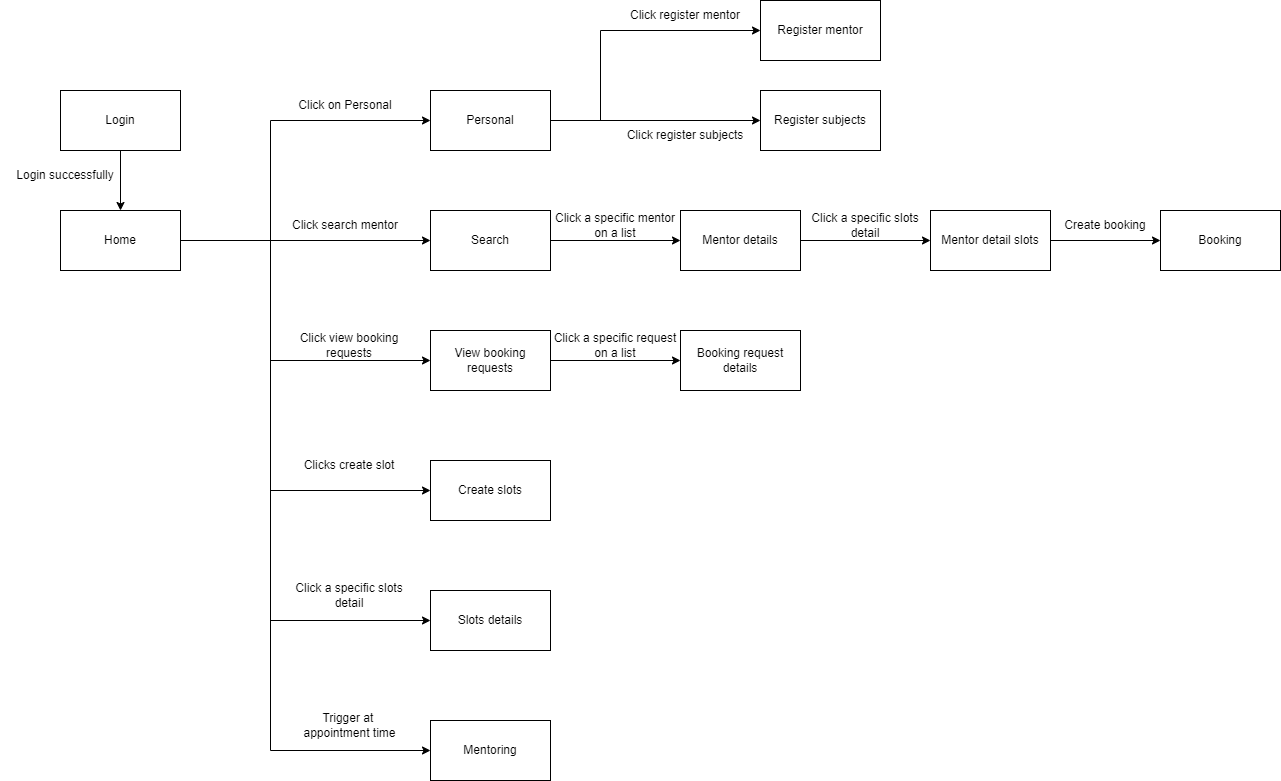
The Mentor Appointment Scheduling Application - MAS Application is a software system that helps the student who has problems with studying or career paths can make an appointment online with a mentor to ask for and retrieve advice from the mentor. This system is expected to help students improve their study quality and clarity in their career paths.

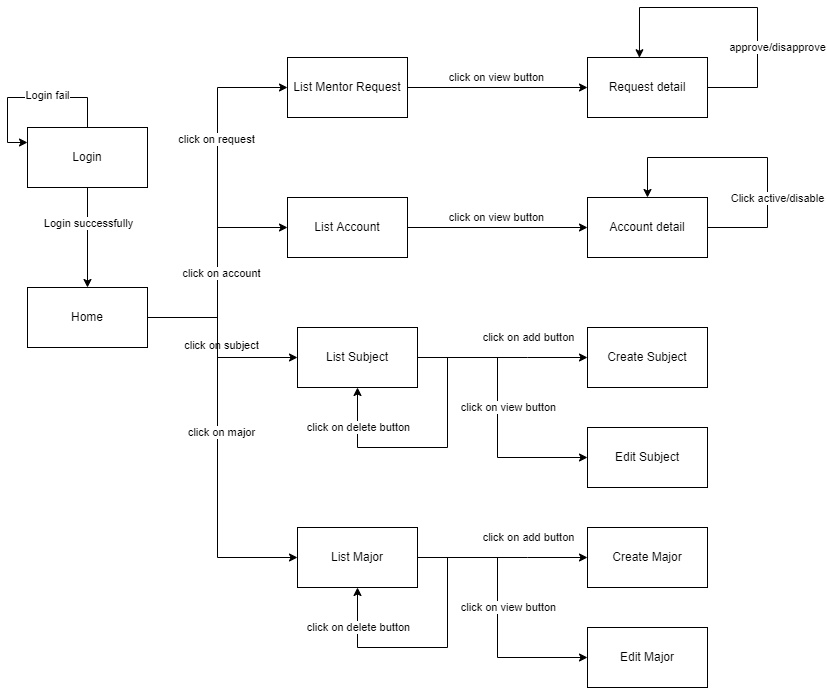


## 2. System Functions

#### a. Screen Flow

* User MAS web application screen flow:



* Admin MAS web application screen flow:

#### b. Screen Details

* User MAS web application screen details:

| **#** | **Feature** | **Screen** | **Description** |
| --- | --- | --- | --- |
| 1 | Login with Google | Login with Google | User login into MAS web application with Google account |
| 2 | View Schedule | Home | View schedule of the user of the current week to the next 3 weeks. |
| 3 | View Profile | Personal | View own student or mentor profile |
| 4 | Search Mentor | Home | Search mentor by name or by subject |
| 5 | View Mentor profile and available slot | Mentor details | View mentor available slots and mentor detail profile |
| 6 | View slot in detail | Mentor slot details | View details available slot. |
| 7 | Booking | Booking | Create a booking request and send it to mentor |
| 8 | View Booking Request | View booking requests | View booking request to available slots |
| 9 | Accept request | View booking requests | Accept a request to available slots |
| 10 | Deny request | View booking requests | Deny a request to available slots |
| 11 | Create available slot | Create available slot page | Create an new available slots |
| 12 | Update available slot | View an available slot details | Update an available slots |
| 13 | Remove available slot | Home | Remove an available slot |
| 14 | Register mentor | Register mentor | User send a mentor registration to admin |
| 15 | Register subjects | Register subjects | User after become a mentor can register to add more subjects |
| 16 | Mentoring | Mentor | User mentor other users |

* Admin MAS web application screen details:

| **#** | **Feature** | **Screen** | **Description** |
| --- | --- | --- | --- |
| 1 | Login with system account | Login | User login into MAS web application with system account |
| 2 | View statistic | Home | View statistics of the system |
| 3 | View all request | Request | View all user requests to become mentor |
| 4 | View request detail | Request | View request details |
| 5 | View all account | Account | View all account of user in system |
| 6 | View account detail | Account detail | View details of specific account |
| 7 | View all major | Major | View all subjects exist in system |
| 8 | Create new major | Create Major | Create new major into system |
| 9 | Edit major | Edit Major | View and edit specific major |
| 10 | View all subject | Subject | View all major exist in system |
| 11 | Create new subject | Create Subject | Create new major into system |
| 12 | Edit subject | Edit Subject | View and edit specific major |

#### c. User Authorization

| **Feature** | **Screen** | **Unauthenticated User** | **User** | **Admin** |
| --- | --- | --- | --- | --- |
| 1. Login with Google | Login |  | X | X |
| 2. View schedule | Home | X |  | X |
| 3. View profile | Personal | X |  | X |
| 3.1. Edit profile | Personal | X |  | X |
| 3.2. Disable subjects | Personal | X |  | X |
| 4. Register mentor | Register mentor | X |  | X |
| 5. Register subjects | Register subject | X |  | X |
| 6. Search mentor | Search | X |  | X |
| 6.1. View list mentor | Search | X |  | X |
| 7. View mentor slots | Mentor details | X |  | X |
| 8. View mentor slot details | Mentor details | X |  | X |
| 9. Booking | Booking | X |  | X |
| 10. View booking requests | View Booking requests | X |  | X |
| 11. View booking requests details | Booking request details | X |  | X |
| 12. Accept booking request | Booking request details | X |  | X |
| 13. Deny booking request | Booking request details | X |  | X |
| 14. Create a new available slot | Create slots | X |  | X |
| 15.1 View an available slots | Slots details | X |  | X |
| 15.2. Update an available slots | Slots details | X |  | X |
| 16. Delete an available slots | Slots details | X |  | X |
| 17. View question | Mentor | X |  | X |
| 18. Answer question | Mentor | X |  | X |
| 19. Login with system account | Login | X | X |  |
| 20. View statistic | Home | X | X |  |
| 21.1 View all request | Request | X | X |  |
| 21.2. View request detail | Request | X | X |  |
| 21.3. Approve/decline request | Request | X | X |  |
| 22.1 View all account | Account | X | X |  |
| 22.2 Active/disable account | Account | X | X |  |
| 23.1 View account detail | Account detail | X | X |  |
| 23.2 Active/disable account | Account detail | X | X |  |
| 24.1 View all subject | Subject | X | X |  |
| 24.2 Delete subject | Subject | X | X |  |
| 25 Create new subject | Create Subject | X | X |  |
| 26.1 Edit subject | Edit Subject | X | X |  |
| 26.2 Delete subject | Edit Subject | X | X |  |
| 27 View all major | Major | X | X |  |
| 28 Add Major | Create Major | X | X |  |
| 29.1 Delete Major | Edit Major | X | X |  |
| 29.2 Edit Major | Edit Major | X | X |  |
| 30 Rating and comment | History Detail | X |  | X |
| 31 View send appointment | Appointment details | x |  | x |
|  |  |  |  |  |
|  |  |  |  |  |

In which:

* Unauthenticated User: User has not logged into MAS web application
* User: User has logged into MAS web application by FPT Google account
* Admin: User has logged into Admin web application as Admin role

#### d. Non-Screen Functions

| **#** | **Feature** | **Description** |
| --- | --- | --- |
| 1 | Send email to user | Send email to user after booking request is accepted |
|  |  |  |

# II. Functional Requirements

1. Admin web application

## 1. Login

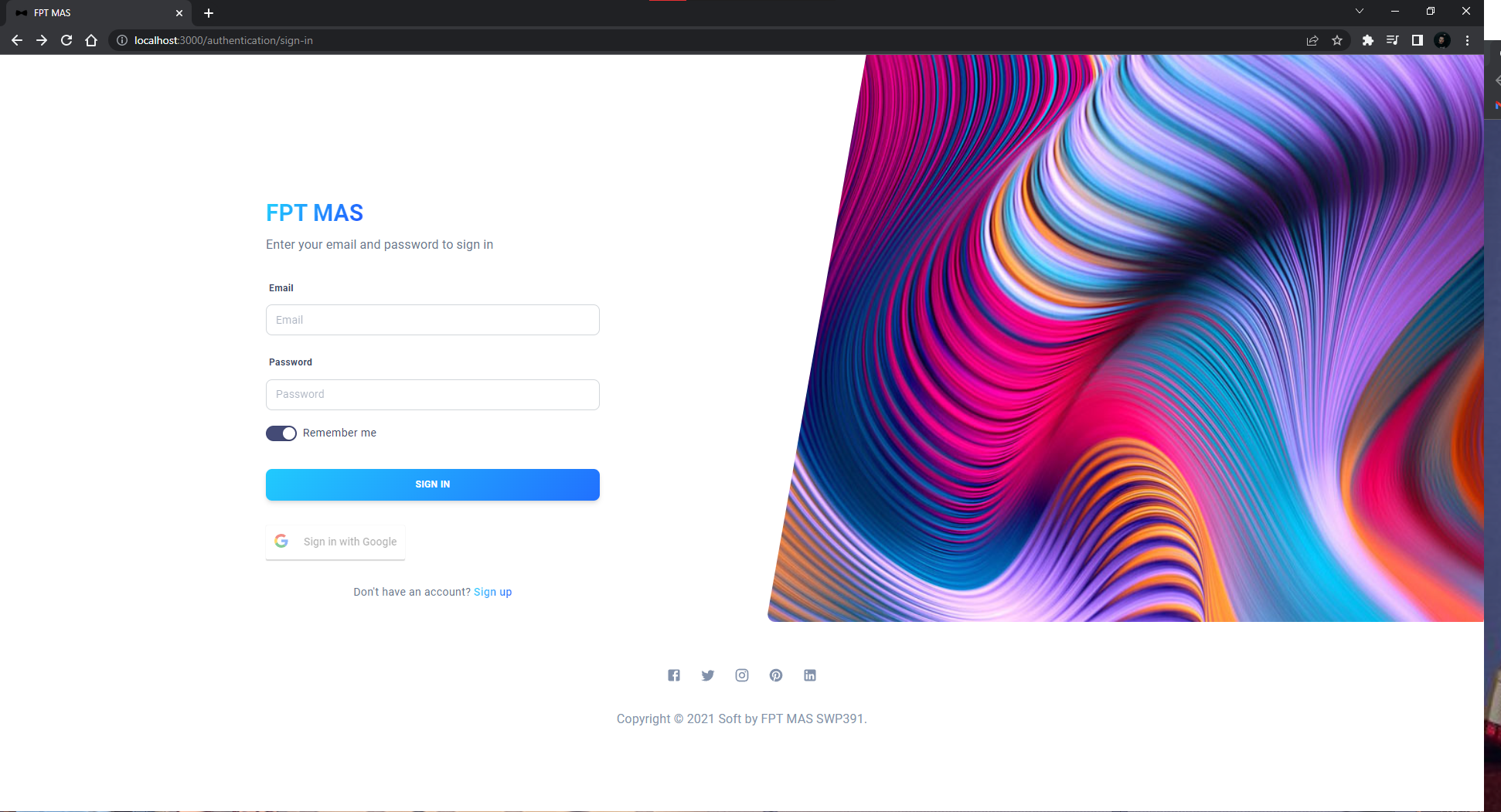
### a. Login

**● Function trigger:** Unauthenticated User request login into system by clicks on “Login” button

**● Function description:** Unauthenticated User login to system to verify role, give them

access to more features in the system

**● Screen layout:**

****

**● Function details:**

* User enters 2 fields:
* Username: Email format
* Password: include at least a number, an uppercase letter and a special character
* User clicks “Dang Nhap” button”
* System generates a token and user account will keep the token to enter the web admin application

## 

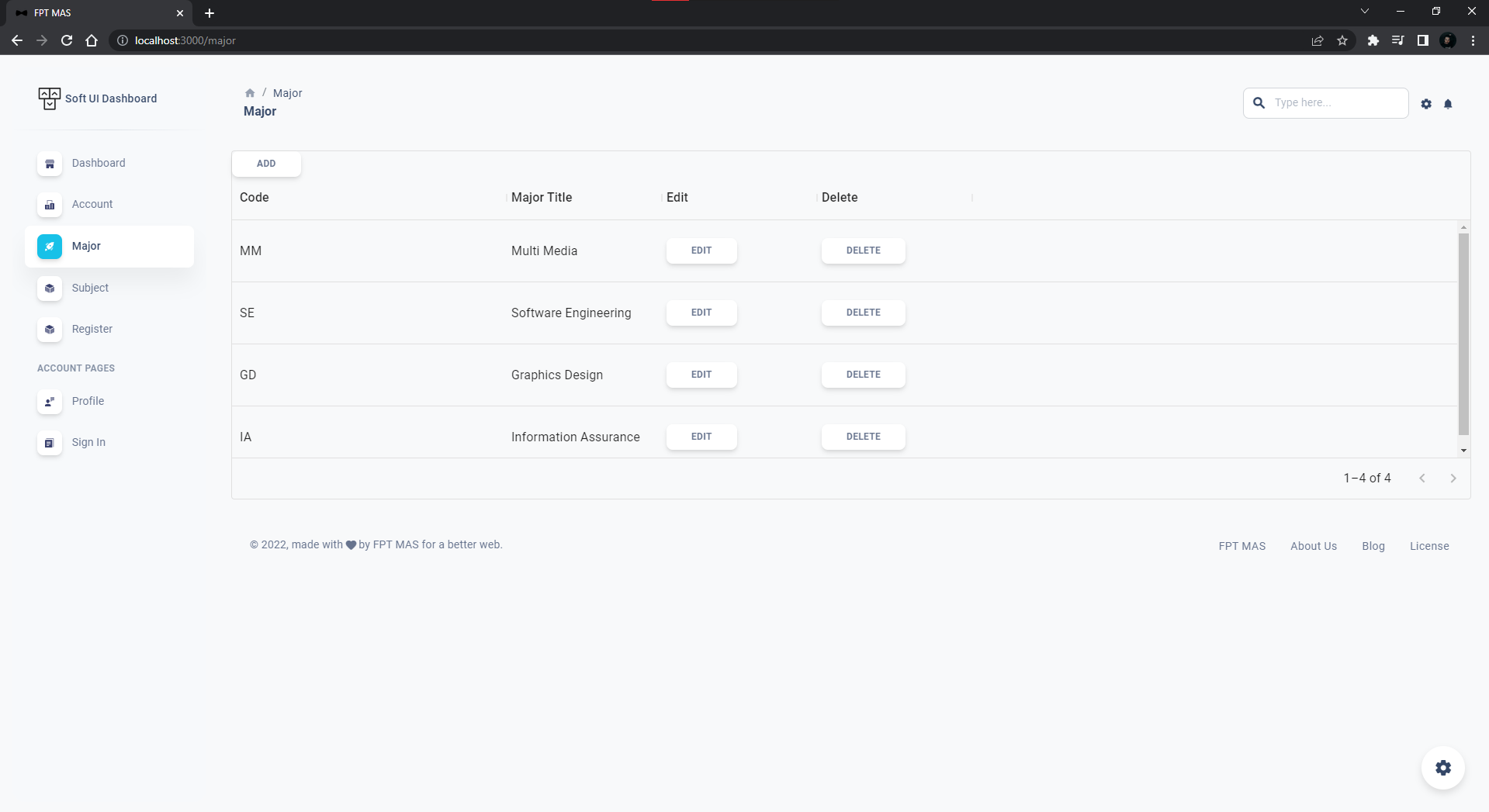
## 2. Manage majors.

### a. View all major

**● Function trigger:** Admin clicks “Major” on the left menu bar.

**● Function description:** Admin request view all major.

**● Screen layout:**

****

**● Function details:**

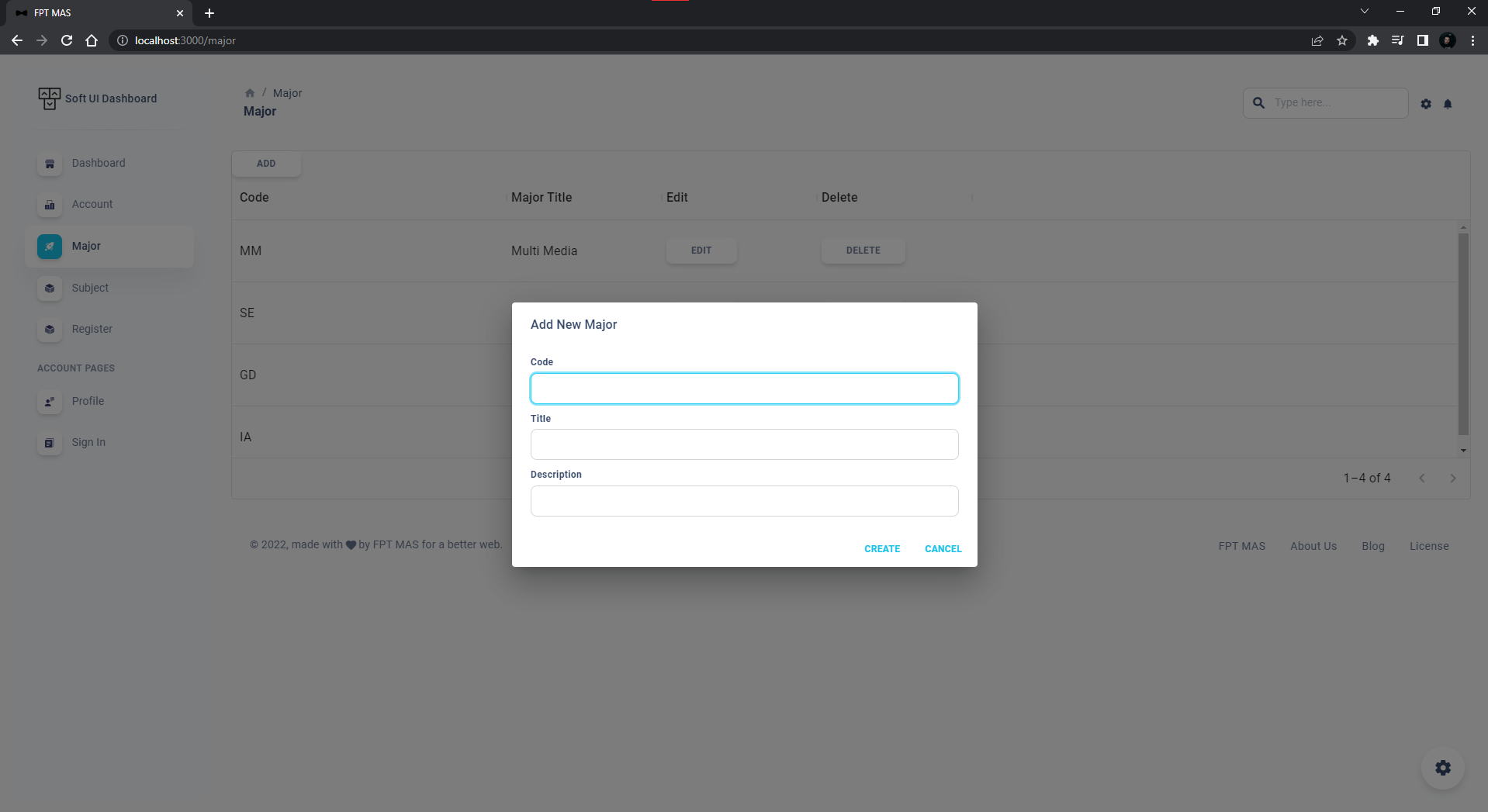
* System displays majors

### b. Create an new major

**● Function trigger:** Admin clicks “Create” on the Create major dialog.

**● Function description:** Admin request create a new major for the application

**● Screen layout:**

****

**● Function details:**

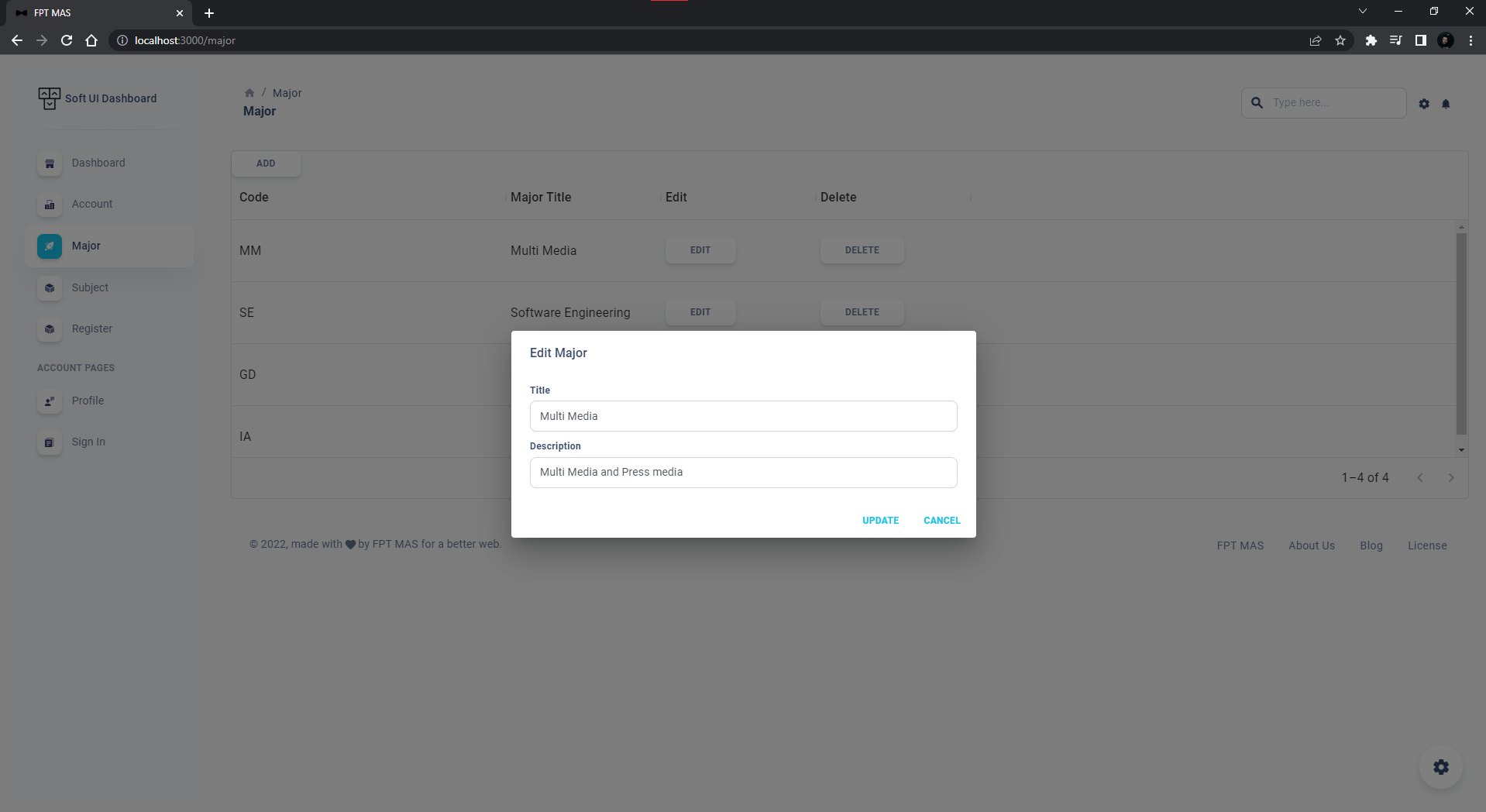
* User clicks plus icon on the Major screen
* Systems pops up a create new major dialog
* User enters below fields:
* Major title: the length from 5 to 10
* Major description: the length from 10 to 100
* User clicks “Create” button
* Systems saves the new major and close the pops up

### c. Update a major

**● Function trigger:** Admin clicks “Update” on the major detail dialog.

**● Function description:** Admin requests update major for the application

**● Screen layout:**

****

**● Function details:**

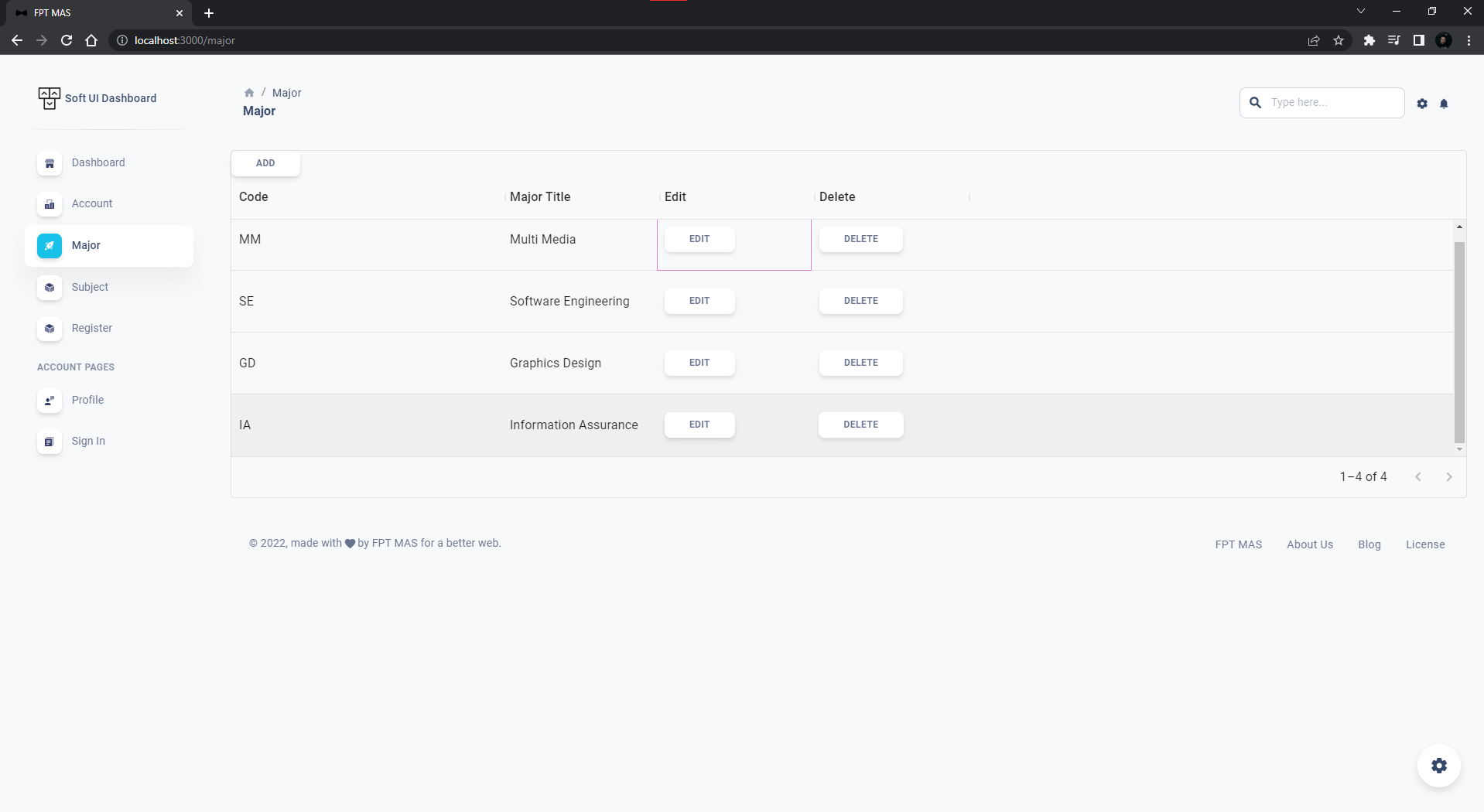
* User clicks on a specific update button on a major table.
* Systems pops up a major details dialog
* User updates below fields:
* Major title: the length from 5 to 10
* Major description: the length from 10 to 100
* User clicks “Update”
* Systems saves the new update major and close the pops up.

### d. Delete a major

**● Function trigger:** Admin clicks “Dong y” on the confirm delete dialog

**● Function description:** Admin requests delete a major for the application

**● Screen layout:**

****

**● Function details:**

* User clicks on a specific delete button on a major table
* Systems pops up a confirm delete dialog
* User clicks “Delete” to delete a major or clicks “Cancel” to not to delete.
* Systems changes the major status isActive to “false”.
* Systems reloads the major table.

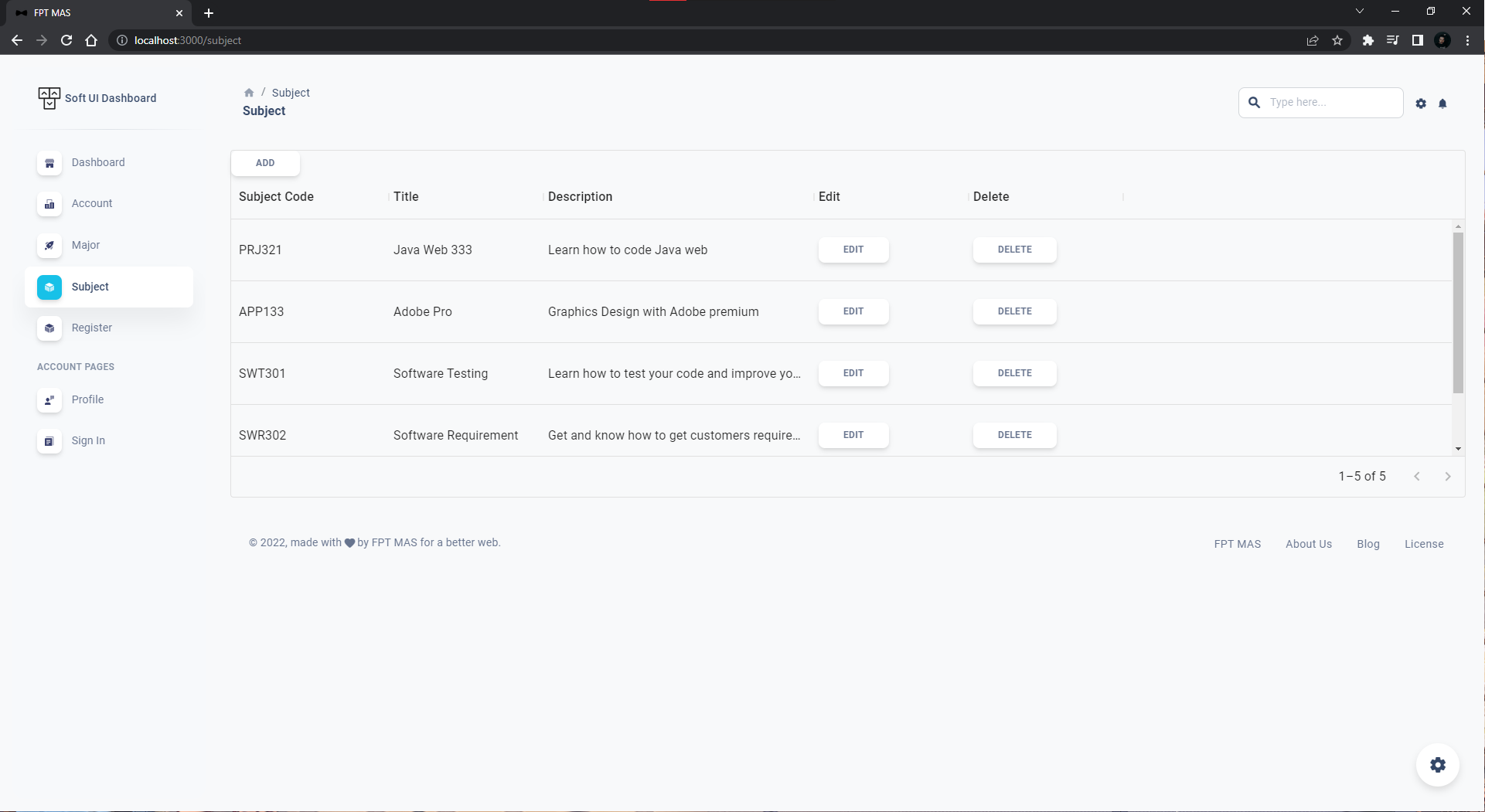
## 3. Manage subjects.

### a. View all subject

**● Function trigger:** Admin clicks “Subjects” on the left menu bar.

**● Function description:** Admin request view all Subjects.

**● Screen layout:**

****

**● Function details:**

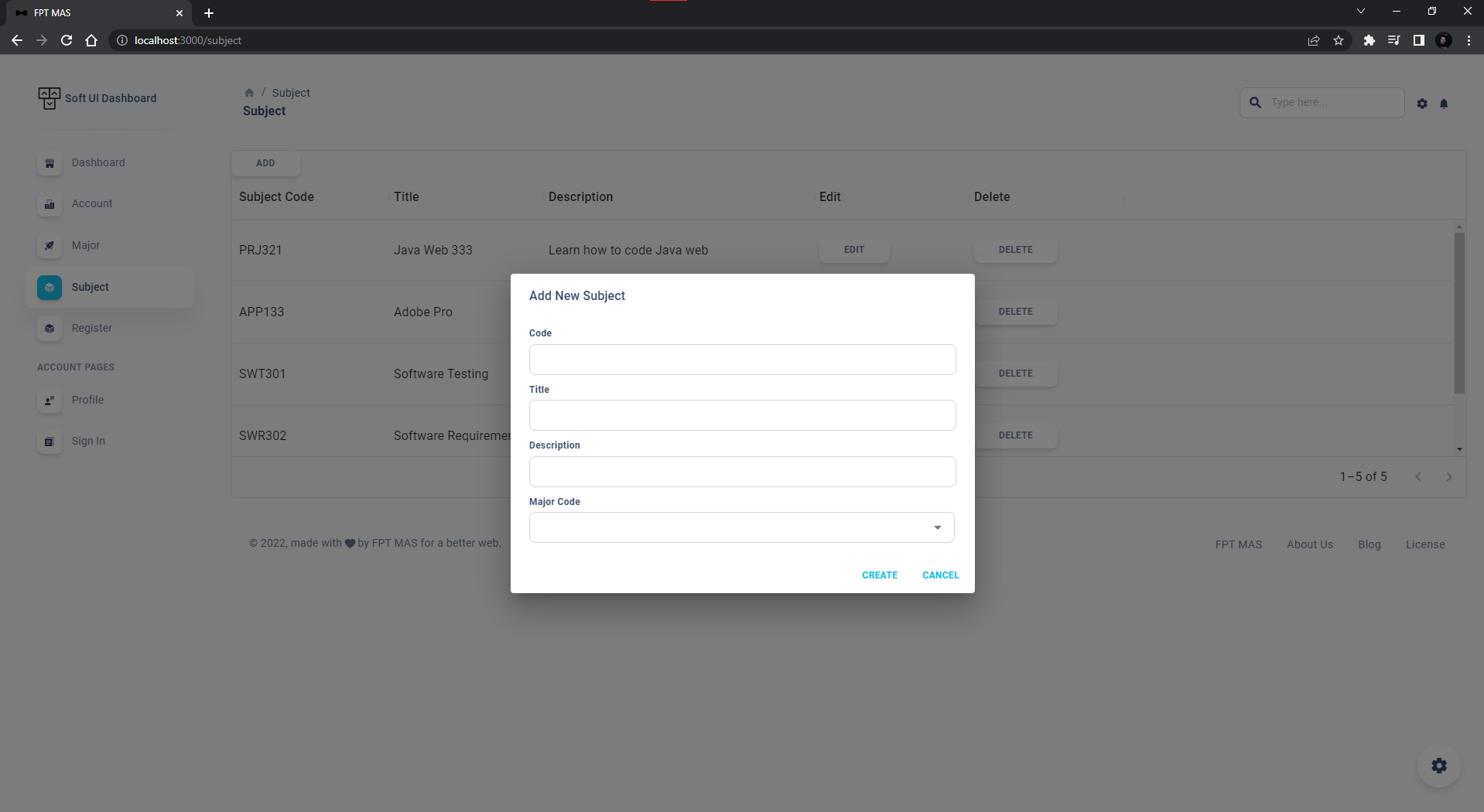
* System displays subjects

### b. Create an new subject

**● Function trigger:** Admin clicks “Create” on the Create subject dialog.

**● Function description:** Admin request create a new subject for the application

**● Screen layout:**

****

**● Function details:**

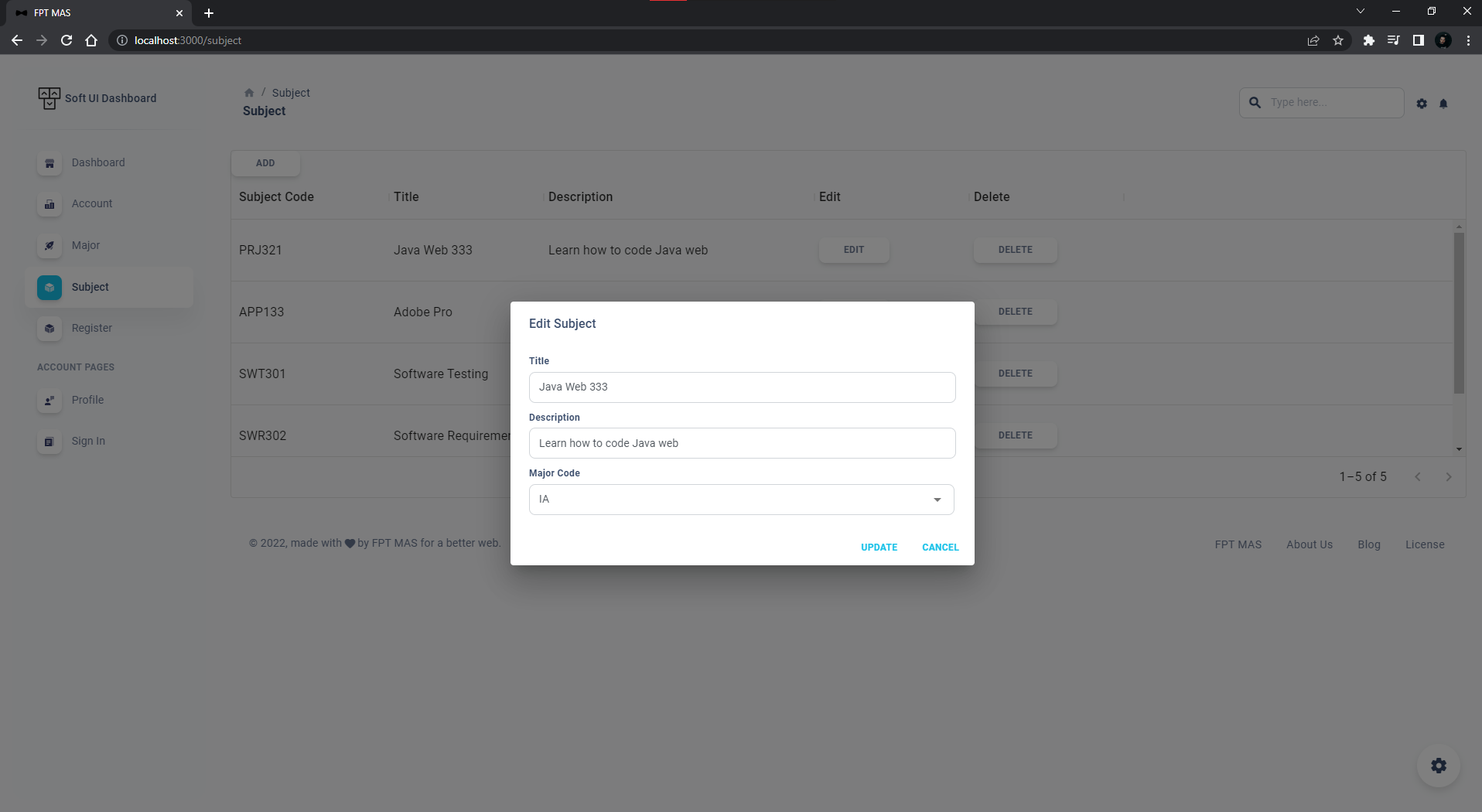
* User clicks plus icon on the subject screen
* Systems pops up a create new subject dialog
* User enters below fields:
* Subject title: the length from 5 to 10
* Subject description: the length from 10 to 100
* Subject major: not null.
* User clicks “Tao moi”
* Systems saves the new subject and close the pops up

### c. Update a subject

**● Function trigger:** Admin clicks “Update” on the subject detail dialog.

**● Function description:** Admin requests update subject for the application

**● Screen layout:**

****

**● Function details:**

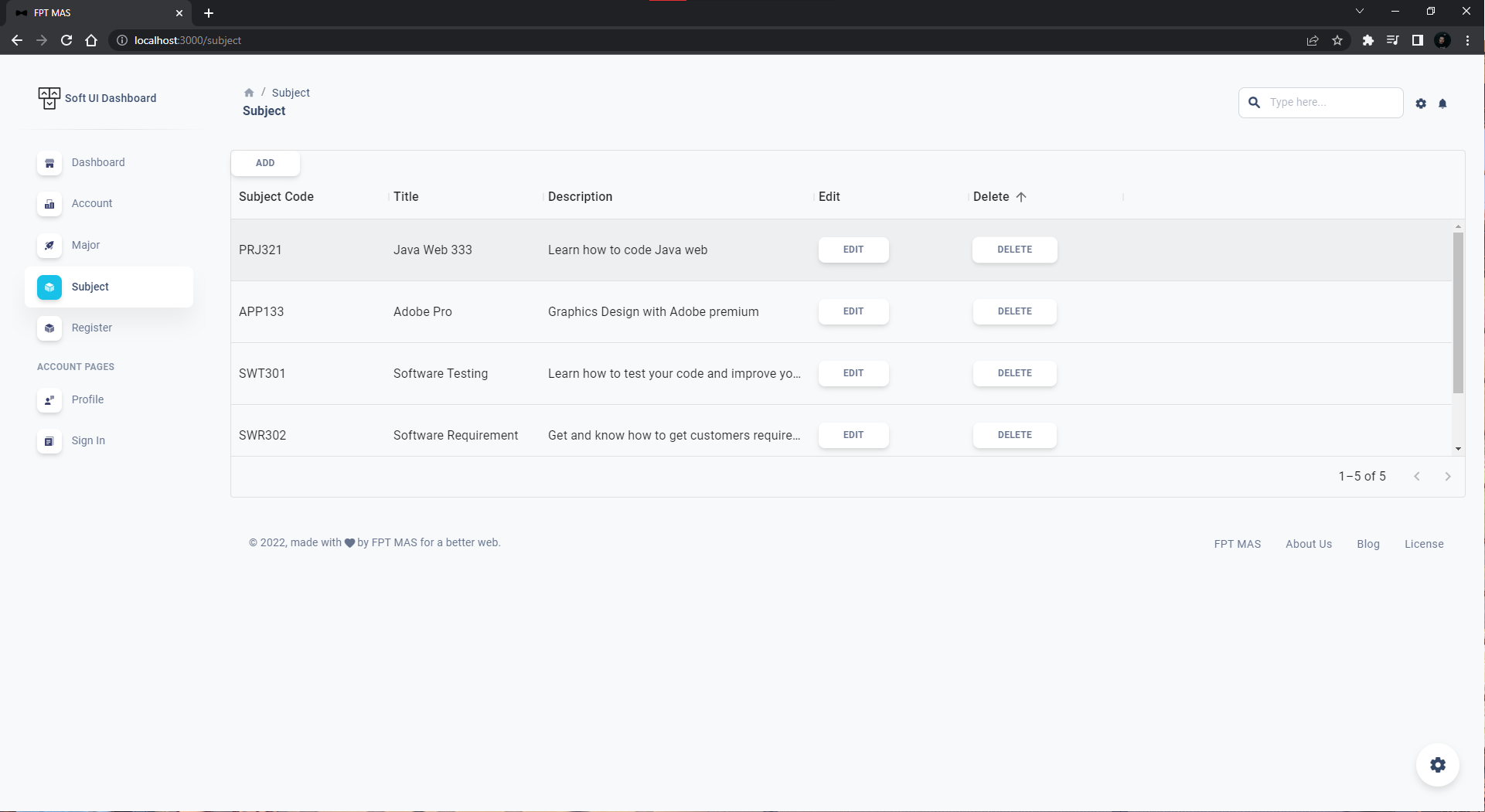
* User clicks on a specific update button on a subject table.
* Systems pops up a subject details dialog
* User updates below fields:
* Subject title: the length from 5 to 10
* Subject description: the length from 10 to 100
* Subject major: not null
* User clicks “Update”
* Systems saves the new update subject and closes the pops up.

### d. Delete a subject

**● Function trigger:** Admin clicks “Delete” on the confirm delete dialog

**● Function description:** Admin requests delete a subject for the application

**● Screen layout:**

****

**● Function details:**

* User clicks on a specific delete button on a subject table
* Systems pops up a confirm delete dialog
* User clicks “Delete” to delete a subject or clicks “Cancel” to not to delete.
* Systems changes the subject status isActive to “false”.
* Systems reloads the subject table.

## 4. Manage accounts.

### a. View all accounts

**● Function trigger:** Admin clicks “Account” on the left menu bar.

**● Function description:** Admin request view all Account.

**● Screen layout:**

**● Function details:**

* System displays Accounts

### b. Disable an account

**● Function trigger:** Admin clicks on a specific disable button on the account table.

**● Function description:** Admin request to disable an account

**● Screen layout:**

**● Function details:**

* User clicks on a specific disable button on the account table
* System pops up a disable confirm dialog
* User clicks “Disable” to disable an account or clicks “Cancel” to not to disable an account
* Systems saves the isActive status of the disable account to false.

## 5. Manage register requests.

### a. View all requests

**● Function trigger:** Admin clicks “Register” on the left menu bar.

**● Function description:** Admin request view all register requests.

**● Screen layout:**

**● Function details:**

* System displays all register requests

### b. View register requests detail

**● Function trigger:** Admin clicks on a specific request on the register request table.

**● Function description:** Admin request to view register request details

**● Screen layout:**

**● Function details:**

* User clicks on a specific requests on register request table
* System pops up a request detail dialog

### c. Accept register requests

**● Function trigger:** Admin clicks Accept button

**● Function description:** Admin request to accept register request.

**● Screen layout:**

**● Function details:**

* User clicks on a specific requests on register request table
* System pops up a request detail dialog
* User clicks Accept button
* System saves the register request owner isMentor status to “True”
* System saves the register request status to “Accepted”

### d. Decline register requests

**● Function trigger:** Admin clicks Decline button

**● Function description:** Admin request to decline register request.

**● Screen layout:**

**● Function details:**

* User clicks on a specific requests on register request table
* System pops up a request detail dialog
* User clicks Decline button
* System saves the register request status to Decline

2. User web application

## 1. Login

### a. Login

**● Function trigger:** Unauthenticated User request login into system by clicks on “Login” button

**● Function description:** Unauthenticated User login to system to verify role, give them

access to more features in the system

**● Screen layout:**

**● Function details:**

* User enters 2 fields:
* Username: Email format
* Password: include at least a number, an uppercase letter and a special character
* User clicks “Dang Nhap” button”
* System generates a token and user account will keep the token to enter the user web application

### b. Logout

**● Function trigger:** User request to logout the user web application

**● Function description:** User logouts the system

**● Screen layout:**

**● Function details:**

* User clicks “Logout” button on the menu bar
* Systems removes the token from local storage and redirects to the login screen.

## 2. Register

**● Function trigger:** Unauthenticated User clicks “Register” button on Register screen

**● Function description:** Unauthenticated User request to register a new account to MAS user web application

**● Screen layout:**

**● Function details:**

* User enters below fields:
* Username: Email format
* Password: include at least a number, an uppercase letter and a special character
* Confirm password: exactly the same as password
* User clicks “Register” button”
* System saves a new account to database and redirect to login screen

## 3. Forgot password

**● Function trigger:** Unauthenticated User clicks “Change password” button on Change password screen

**● Function description:** Unauthenticated User request to change account password

**● Screen layout:**

**● Function details:**

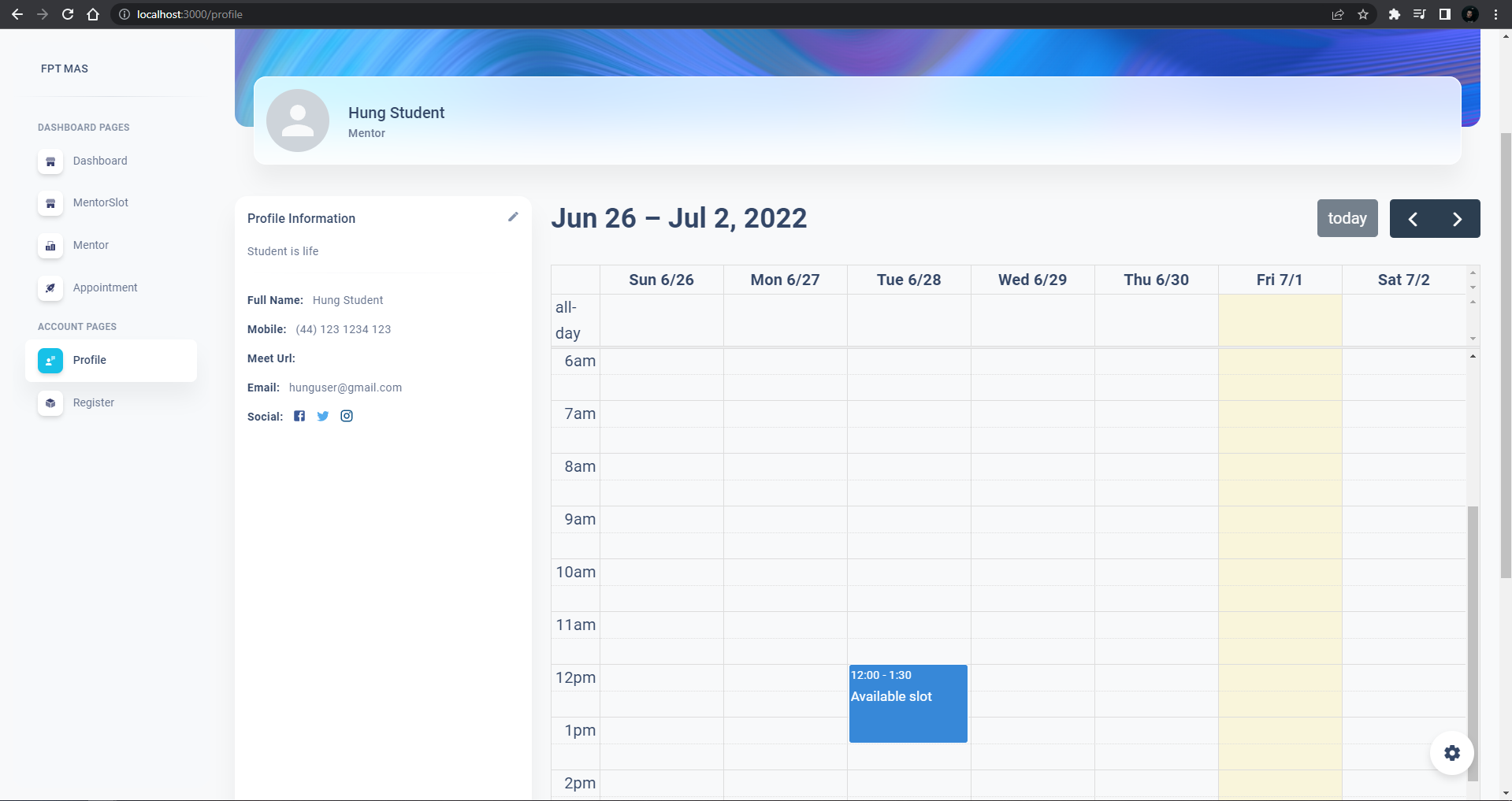
* User enters below fields:
* Username: Email format
* Password: include at least a number, an uppercase letter and a special character
* Confirm password: exactly the same as password
* User clicks “Register” button”
* System saves a new account to database and redirect to login screen

## 4. View own profile

**● Function trigger:** User clicks “Profile” on menu bar

**● Function description:** User requests to view own profile

**● Screen layout:**

****

**● Function details:**

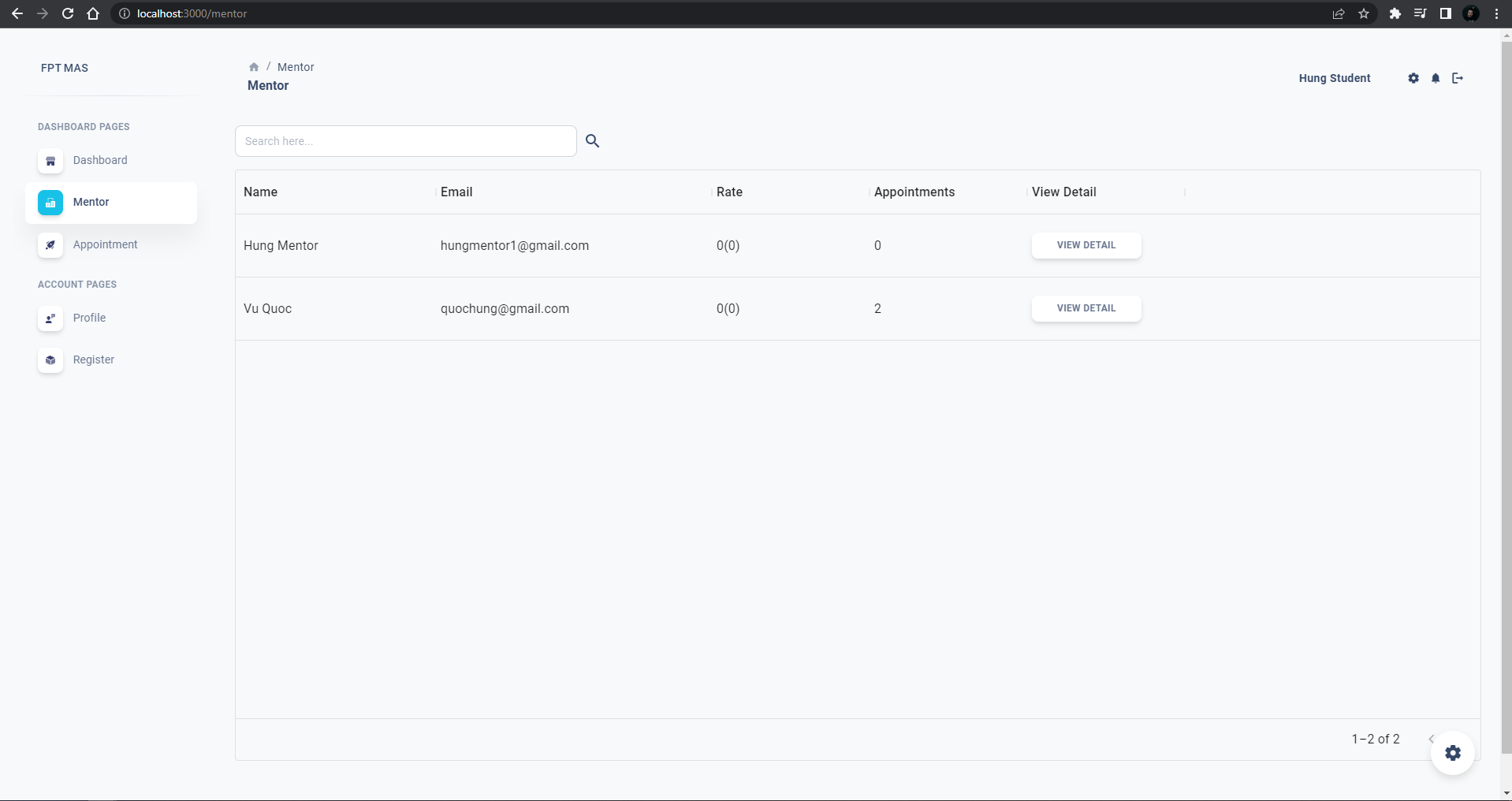
* System displays user profile details with calendar belows value:
* name
* meet url
* email
* Slots

## 5. View and Search mentor

**● Function trigger:** User clicks “mentor” on menu bar

**● Function description:** User requests to search mentor

**● Screen layout:**

****

**● Function details:**

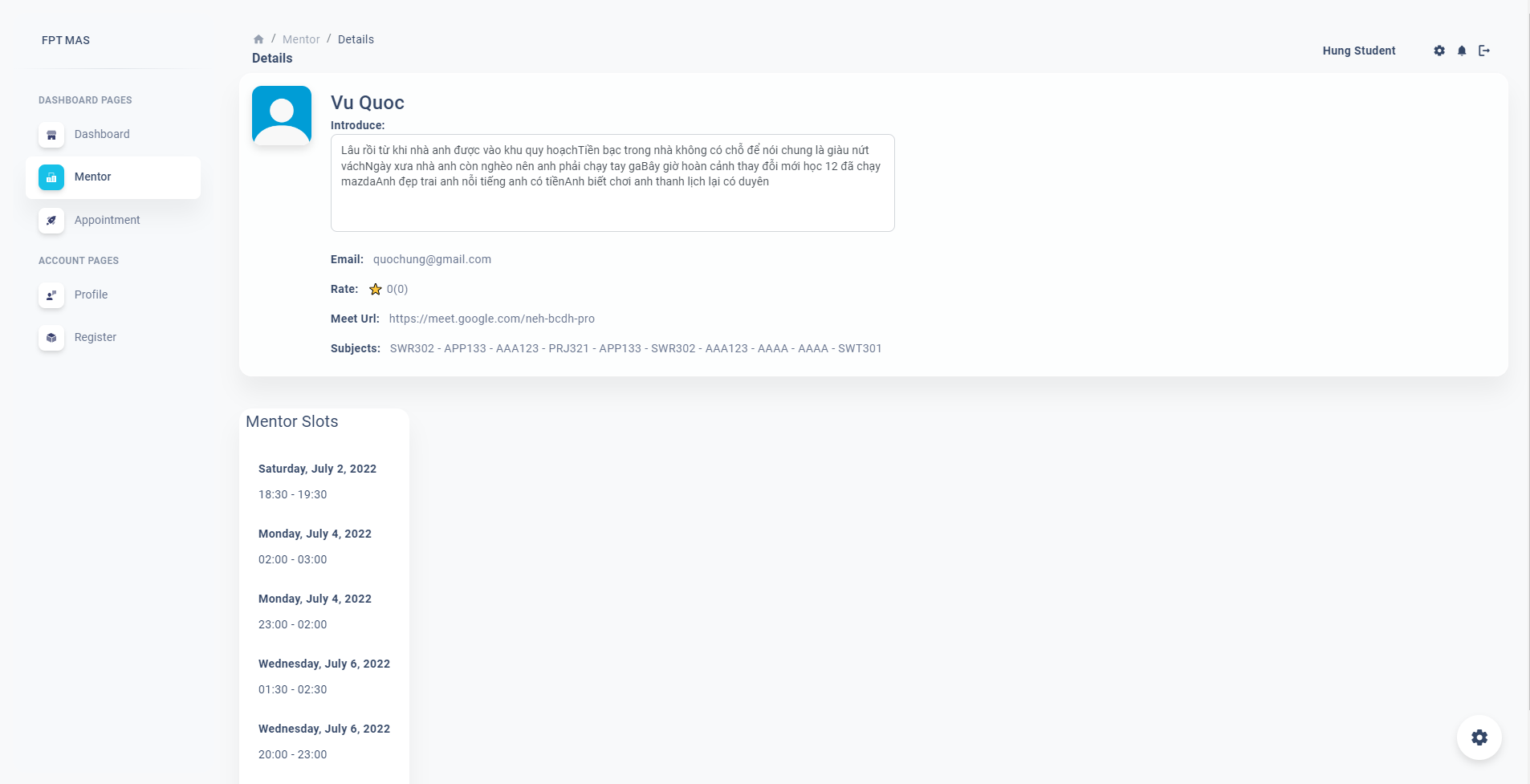
* User enters mentor name and press the search icon
* System displays mentor list

## 6. View mentor details

**● Function trigger:** User clicks a view detail on a specific mentor

**● Function description:** User requests to view detail

**● Screen layout:**

****

**● Function details:**

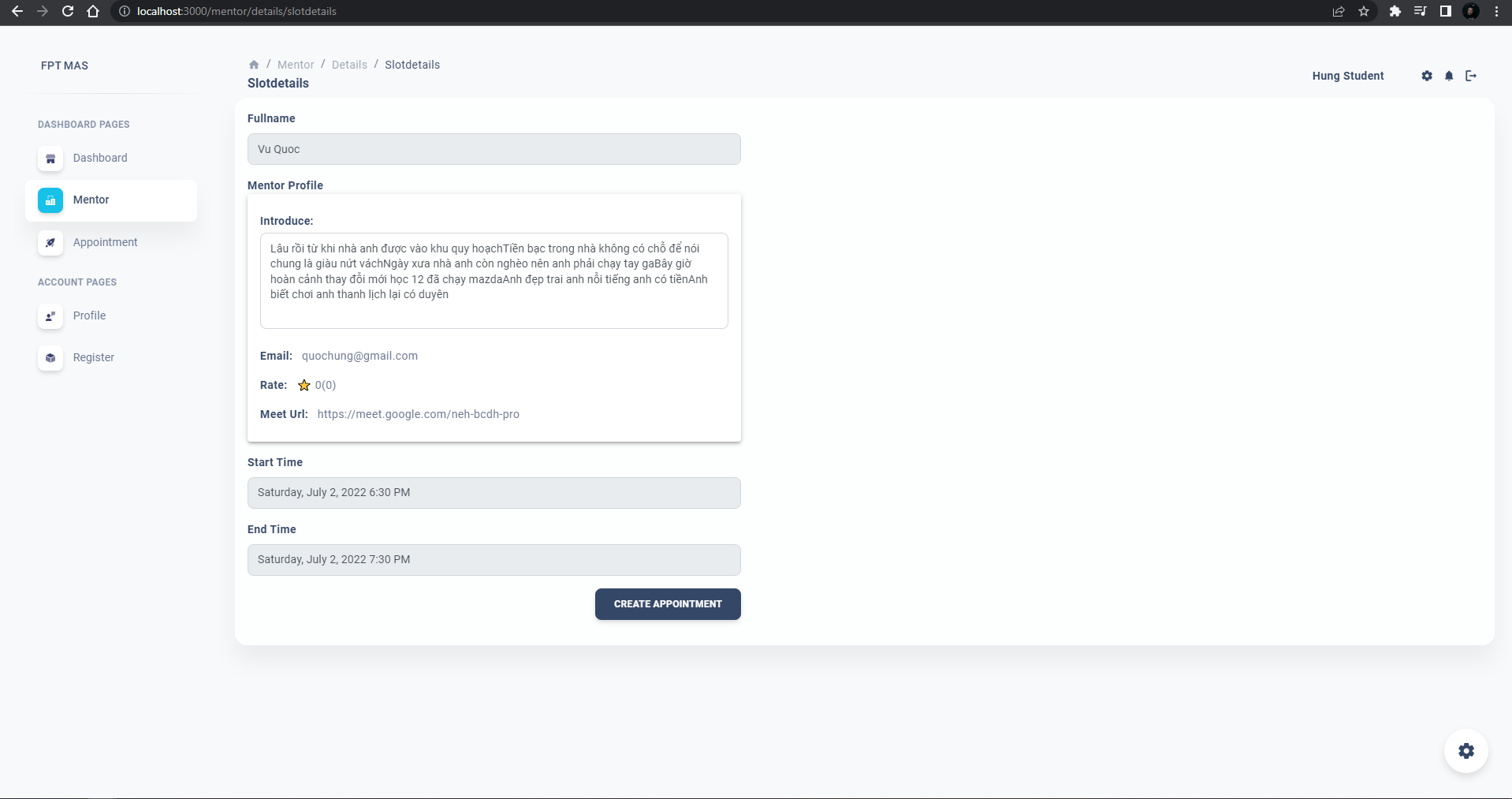
* System displays below fields:
* Name
* Introduce
* Rating
* Meet url
* Mentor Subjects
* Slots

## 7. View slot details

**● Function trigger:** User clicks a specific mentor slot

**● Function description:** User requests to view slot detail

**● Screen layout:**

****

**● Function details:**

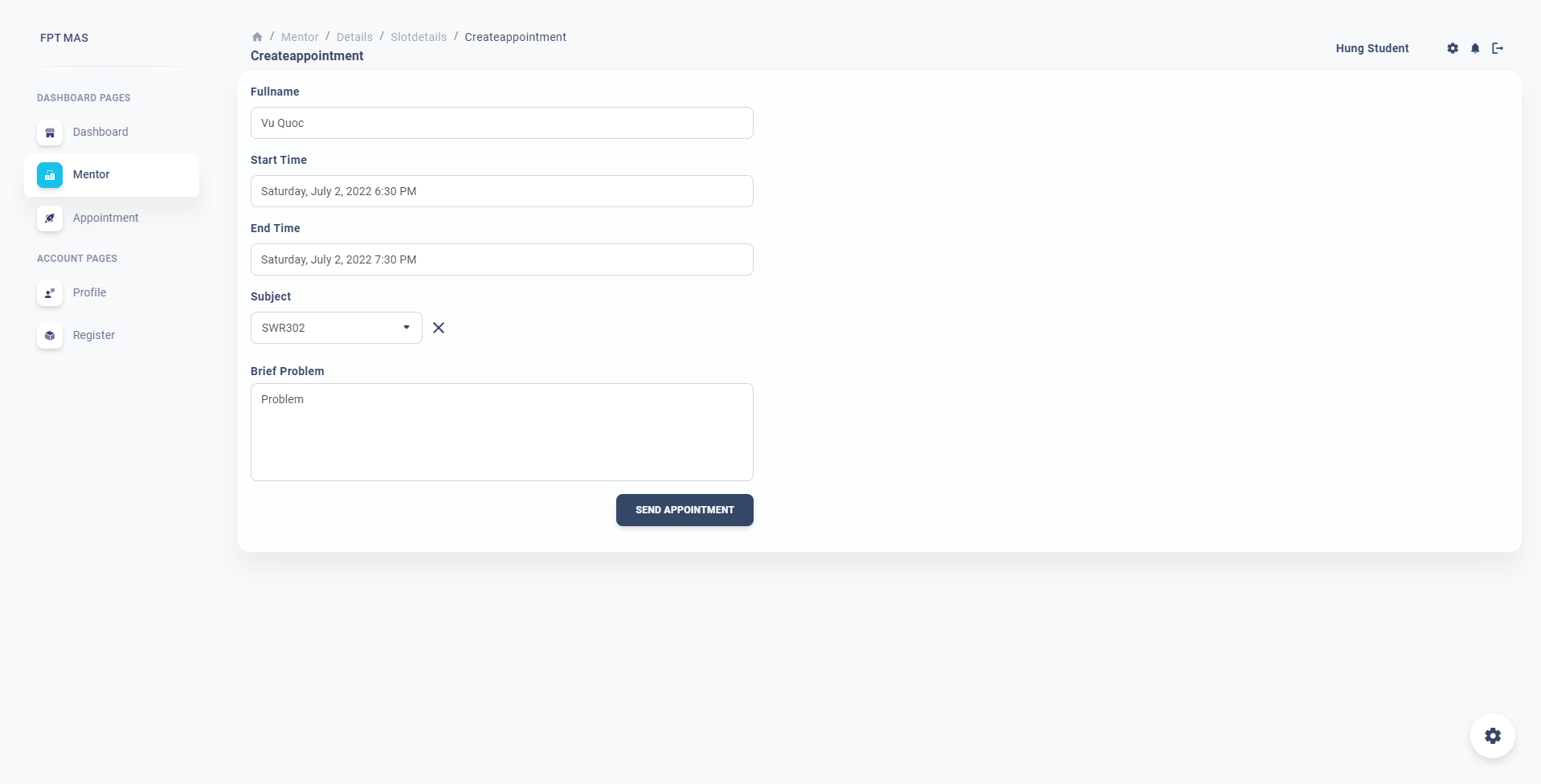
* System displays below fields:
* Mentor name
* Introduce
* Rating
* Meet url
* Start time
* End time

## 8. Create appointment

**● Function trigger:** User clicks a create appointment button

**● Function description:** User requests to create appointment

**● Screen layout:**

****

**● Function details:**

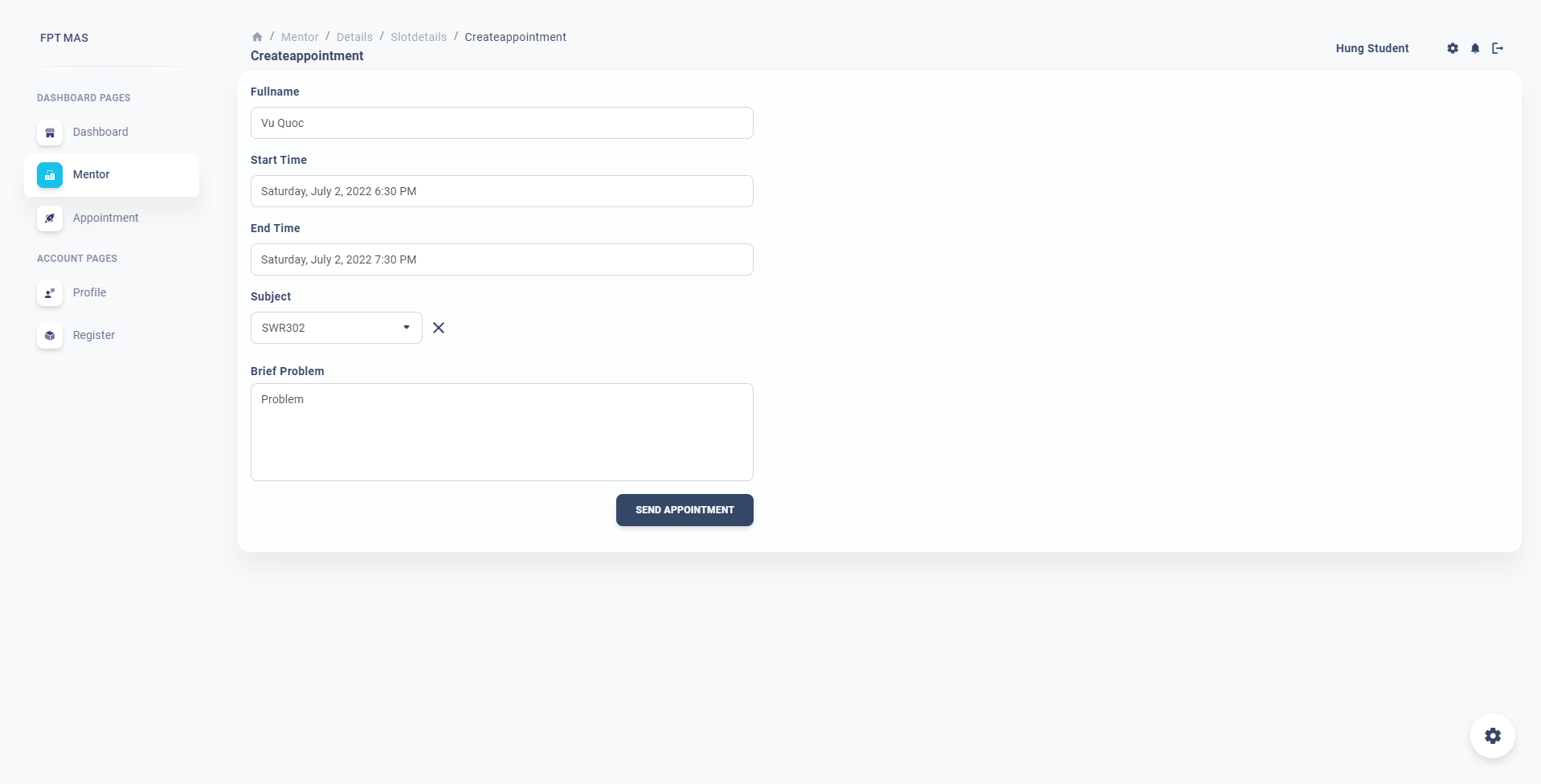
* System displays below fields:
* Mentor name
* Start time
* End time
* User enters below fields:
* Subject to mentor
* Brief problem

## 9. Send appointment

**● Function trigger:** User clicks a send appointment button

**● Function description:** User requests to send appointment

**● Screen layout:**

****

**● Function details:**

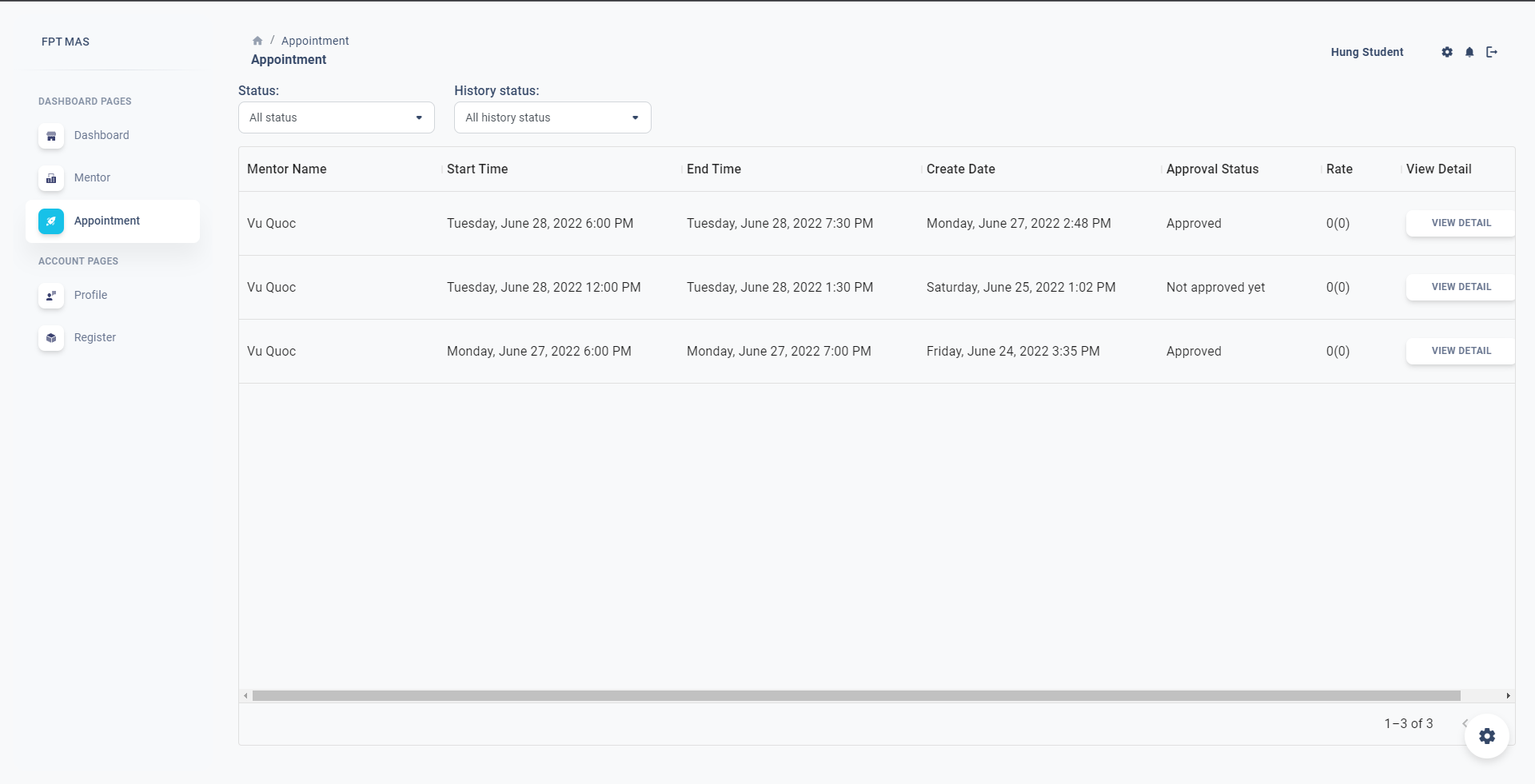
* System saves appointment status is “Not approved yet”
* System redirects to Appointment screen

## 10. View send appointment list

**● Function trigger:** User clicks appointment on menu bar

**● Function description:** User requests to view send appointment

**● Screen layout:**

****

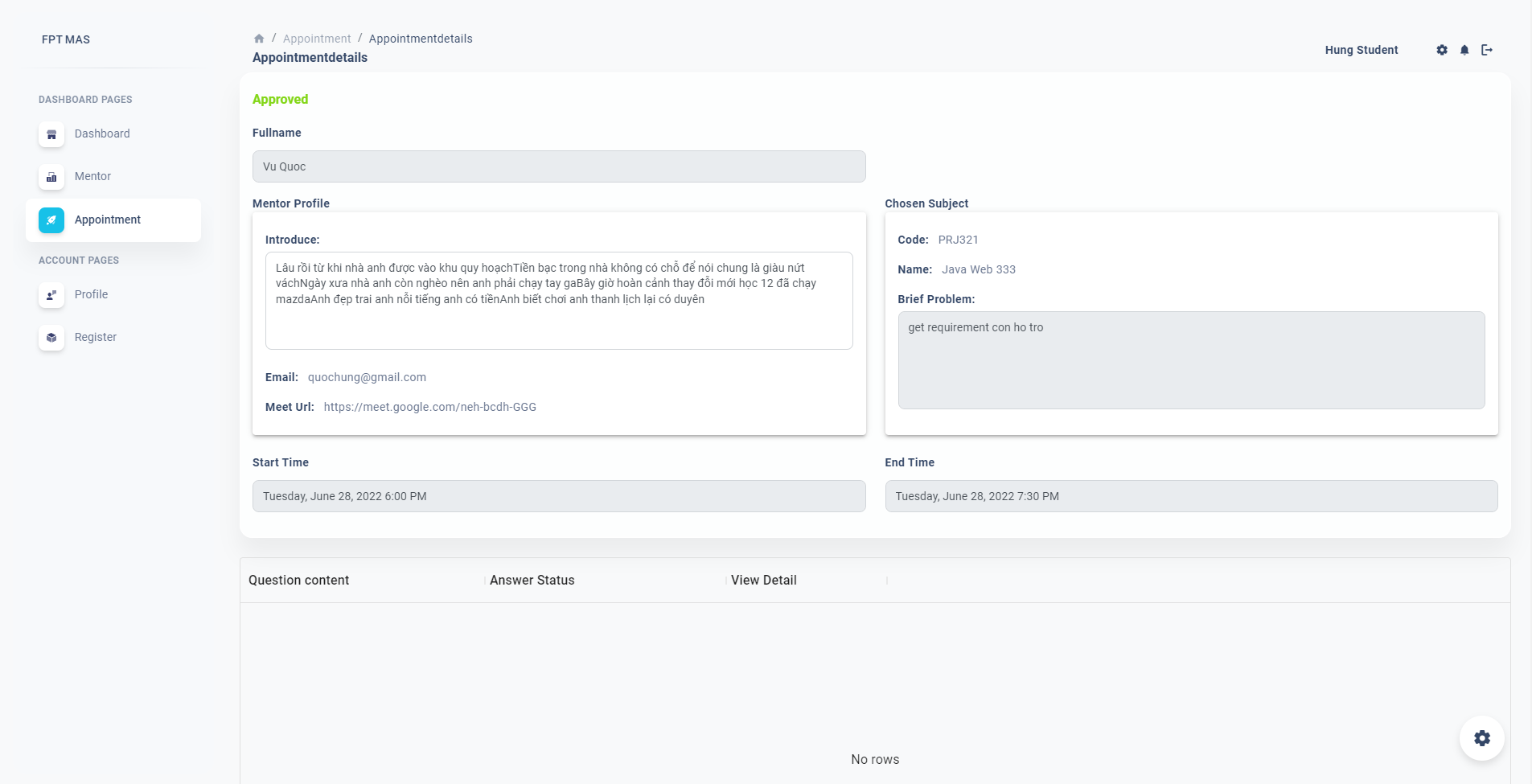
**● Function details:**

* System displays appointment list with status filter and status history filter

## 11. View appointment details

**● Function trigger:** User clicks on a specific appointment

**● Function description:** User requests to view send appointment details

**● Screen layout:**

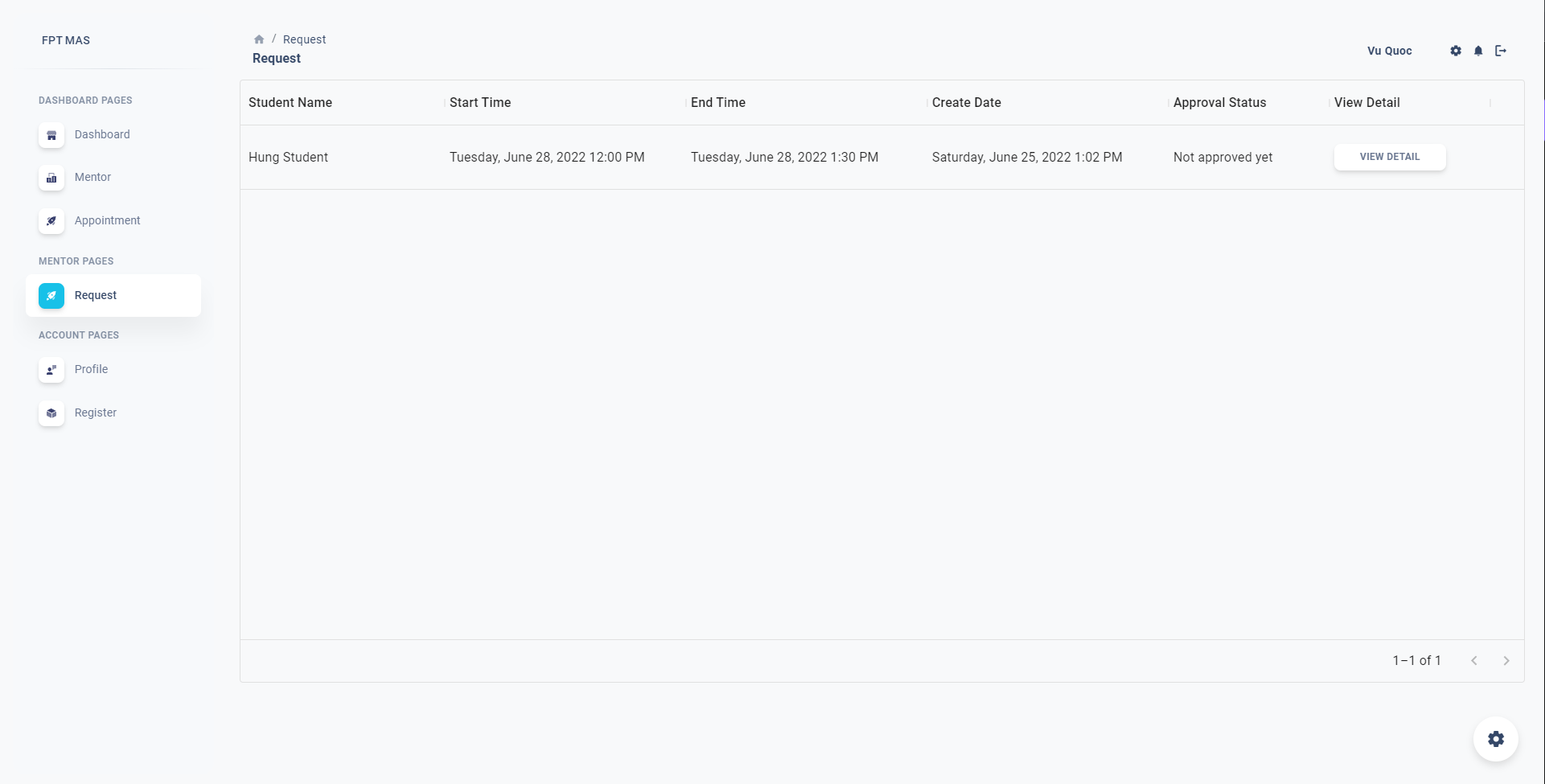
**● Function details:**

* System displays appointment details information

## 12. View appointment request list

**● Function trigger:** User(mentor) clicks request on menu bar

**● Function description:** User requests to view appointment requests

**● Screen layout:**

**● Function details:**

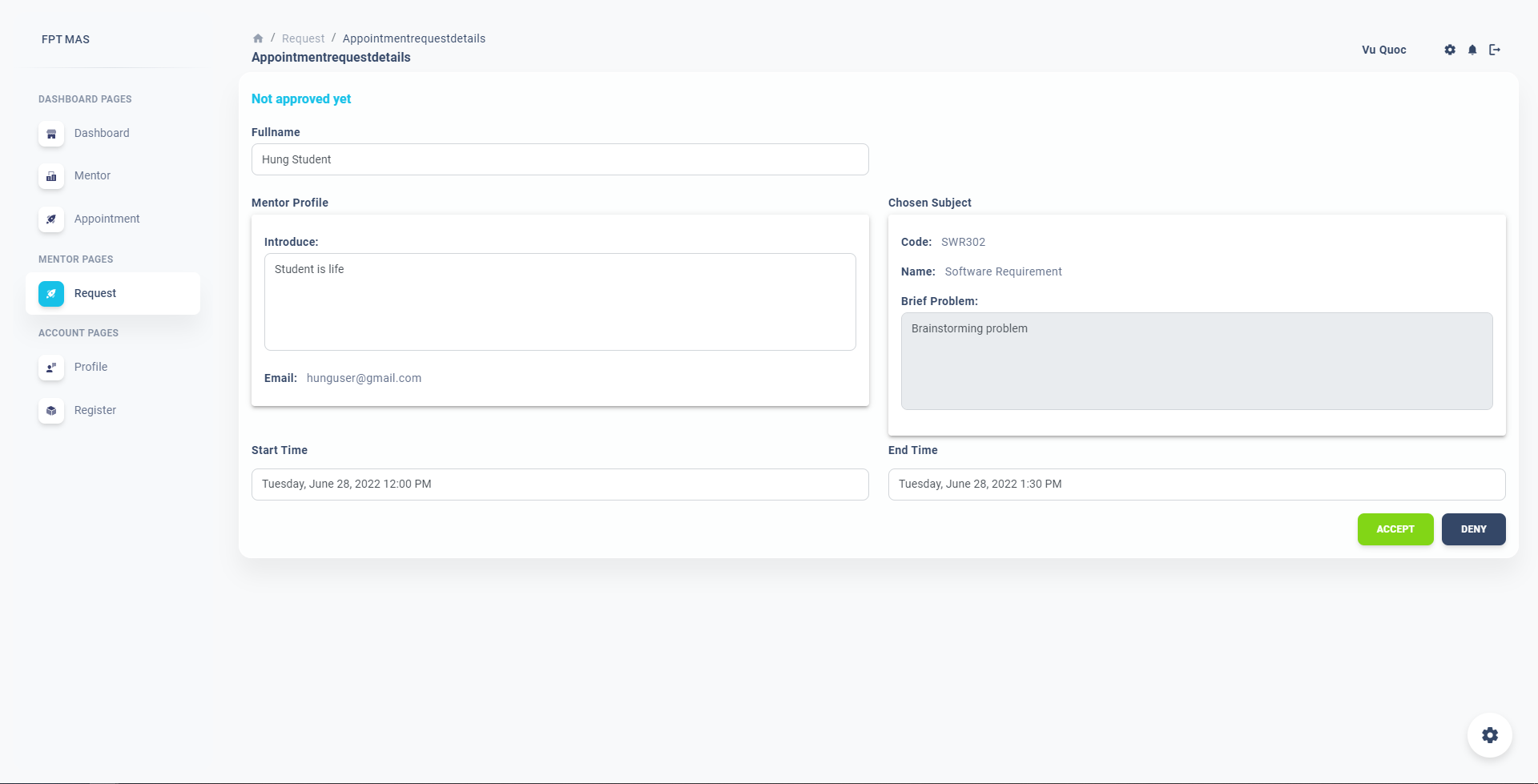
* System displays received appointment request

## 13. View appointment request details

**● Function trigger:** User(mentor) clicks a specific details on list

**● Function description:** User requests to view appointment requests details

**● Screen layout:**

****

**● Function details:**

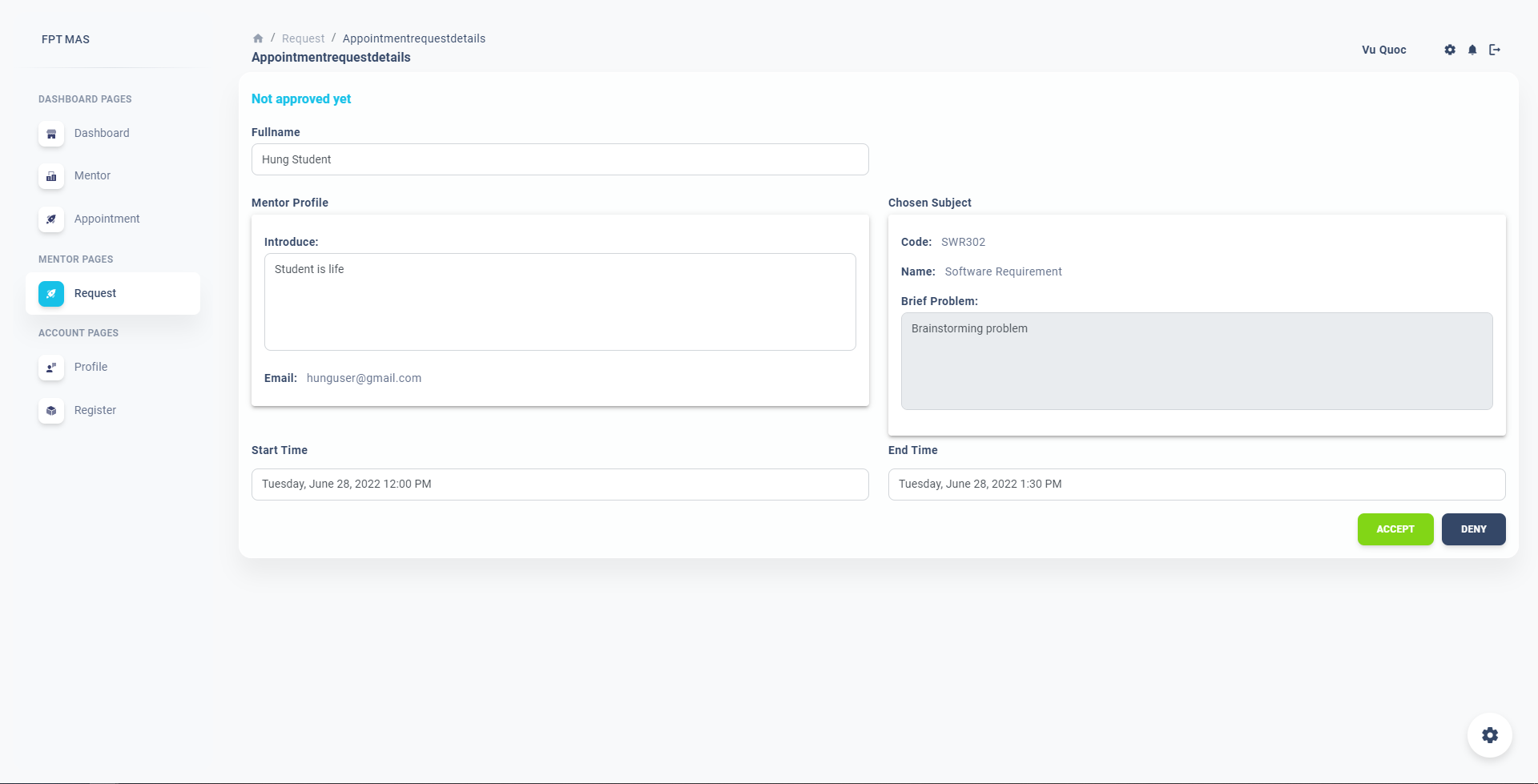
* System displays received appointment request details

## 14. Accept appointment request

**● Function trigger:** User(mentor) clicks Accept button

**● Function description:** User requests to accept appointment requests

**● Screen layout:**

****

**● Function details:**

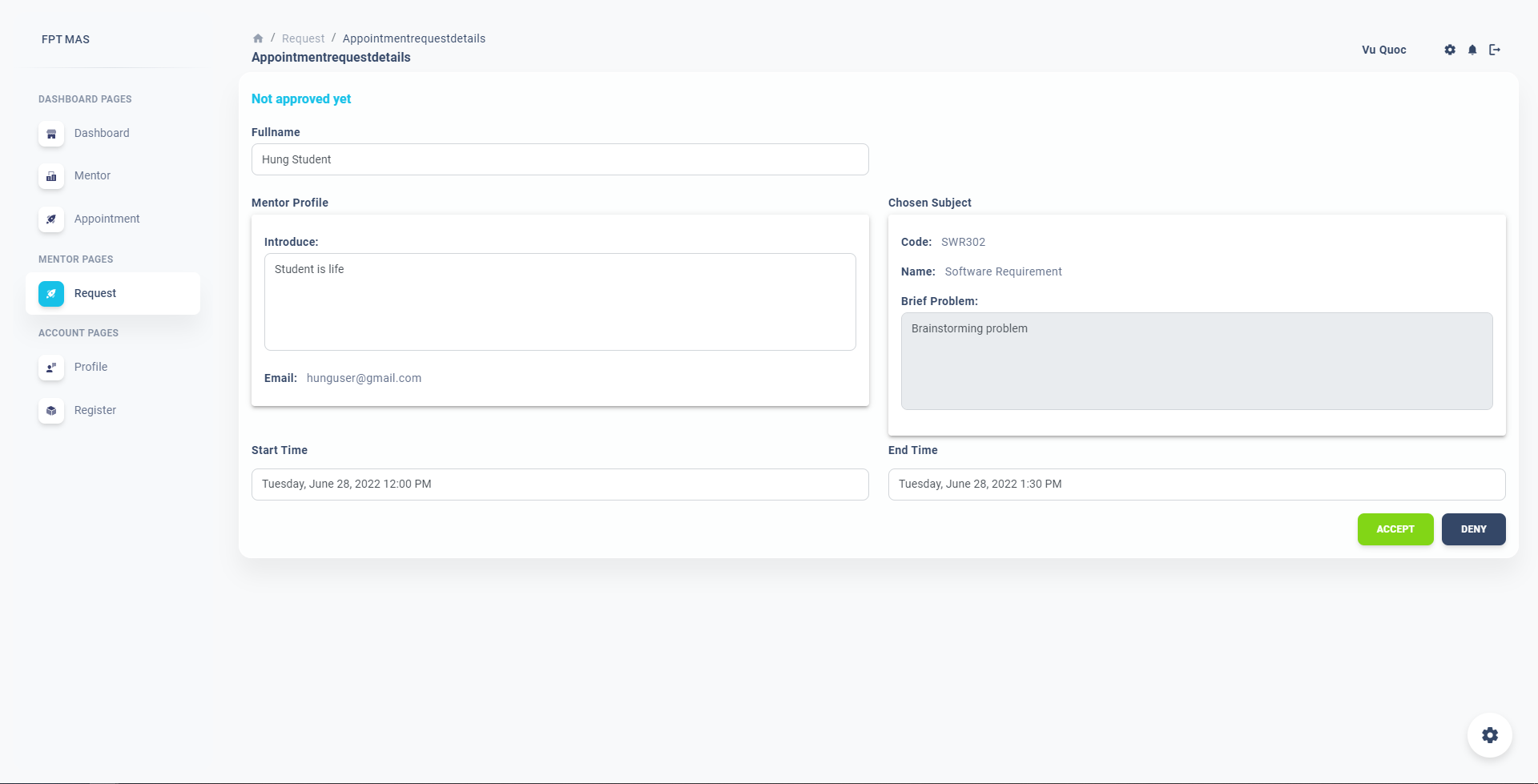
* System displays received appointment request details
* User clicks accept button
* System saves appointment status to Approved
* System redirects to dashboard

## 15. Deny appointment request

**● Function trigger:** User(mentor) clicks Deny button

**● Function description:** User requests to deny appointment requests

**● Screen layout:**

****

**● Function details:**

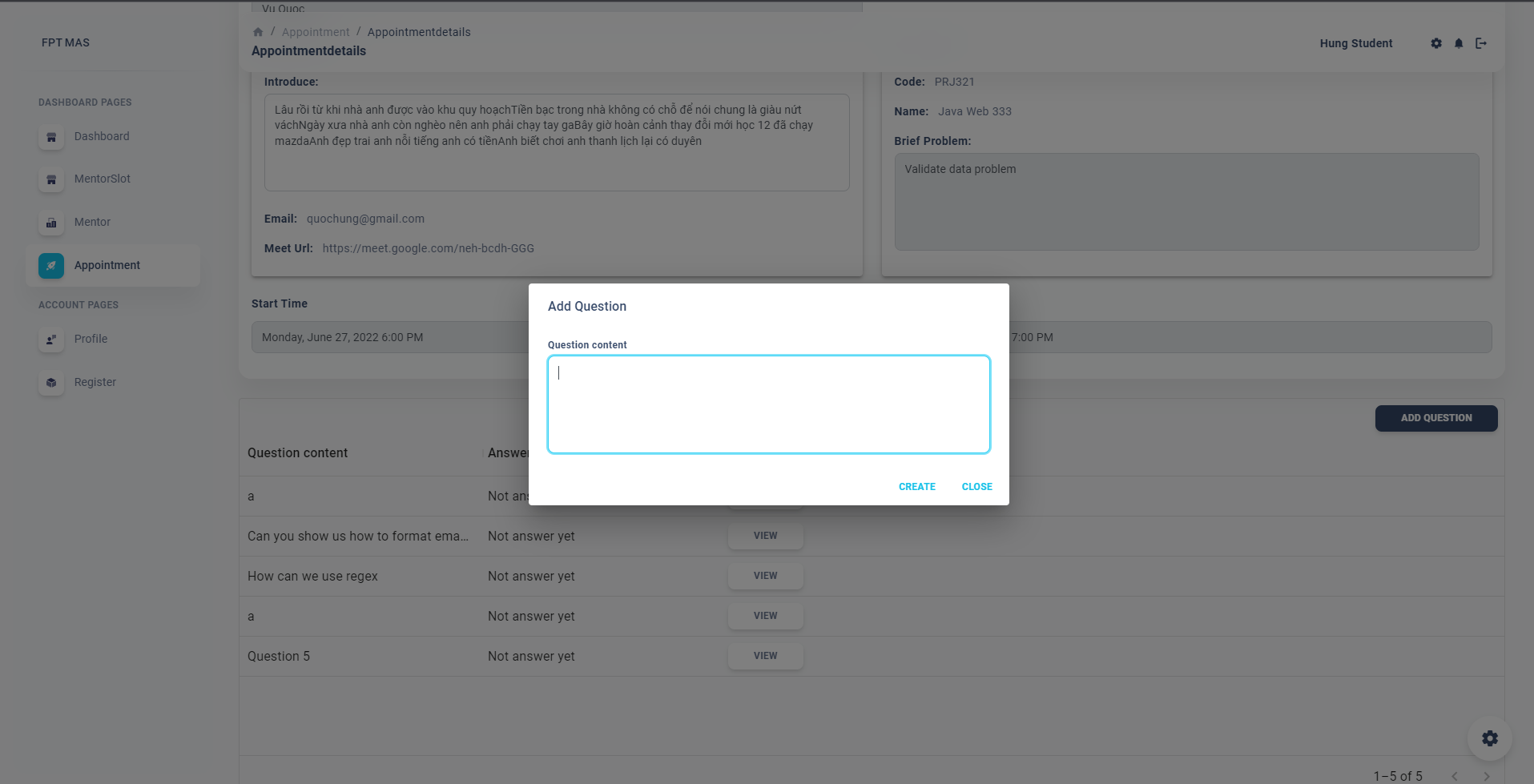
* System displays received appointment request details
* User clicks deny button
* System saves appointment status to Denied
* System redirects to dashboard

## 16. Add question

**● Function trigger:** User clicks Create button

**● Function description:** User requests to create a new question

**● Screen layout:**

****

**● Function details:**

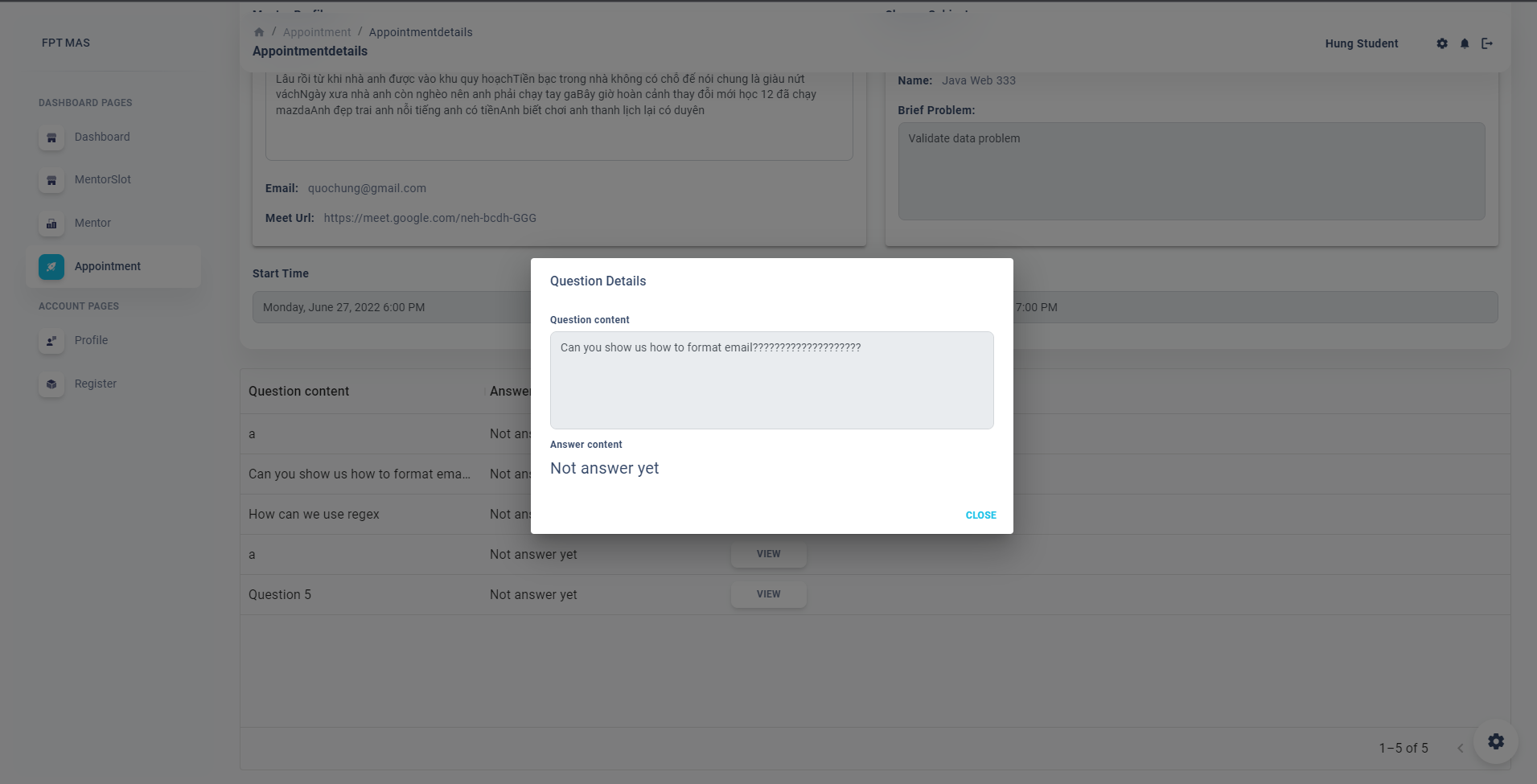
* User enters question content
* User clicks Create button
* System saves question with status not answer yet

## 17. View question

**● Function trigger:** User clicks View button

**● Function description:** User requests to view question details

**● Screen layout:**

****

**● Function details:**

* System display question details.

## 18. Answer question

**● Function trigger:** Mentor(user) clicks Save button

**● Function description:** User requests to answer question

**● Screen layout:**

**● Function details:**

* System display question details.
* User enter answer content
* User clicks Save button
* System saves question with answer and change status to answered