# **CE201 Team MVP Report**

Team Number:

Module Code: CE201  
  
 GitHub Repository URL: [*https://github.com/timsyf/CE201*](https://github.com/timsyf/CE201)

GitHub Project URL: [*https://github.com/users/timsyf/projects/1/views/1*](https://github.com/users/timsyf/projects/1/views/1)

Team Members:

· *Lee Wei Ting Bernard PRID: LEEWE10803*

· *Timothy Soon Yu Feng PRID: SOONT31607*

· *Hagman Lim Sheng Yuan PRID:LIMHA88502*

· *Ying Li*

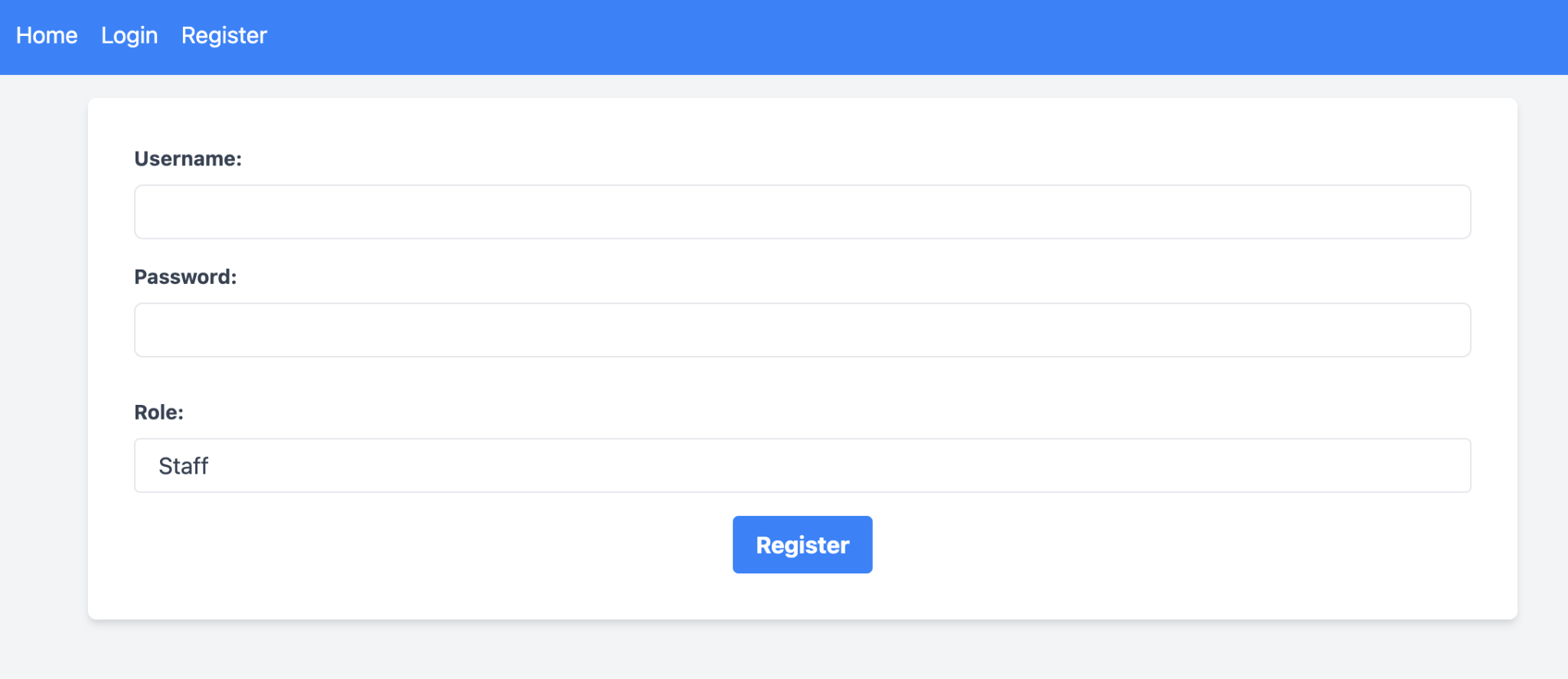
*The (Minimum Viable Product) MVP was decided based on the user stories. The product is a tracking system that needs authentication, tracks training hours , courses and provides a report.*

*Hence, we decided that the most important feature will be authentication,the tracking of the training hours , courses and reports.*

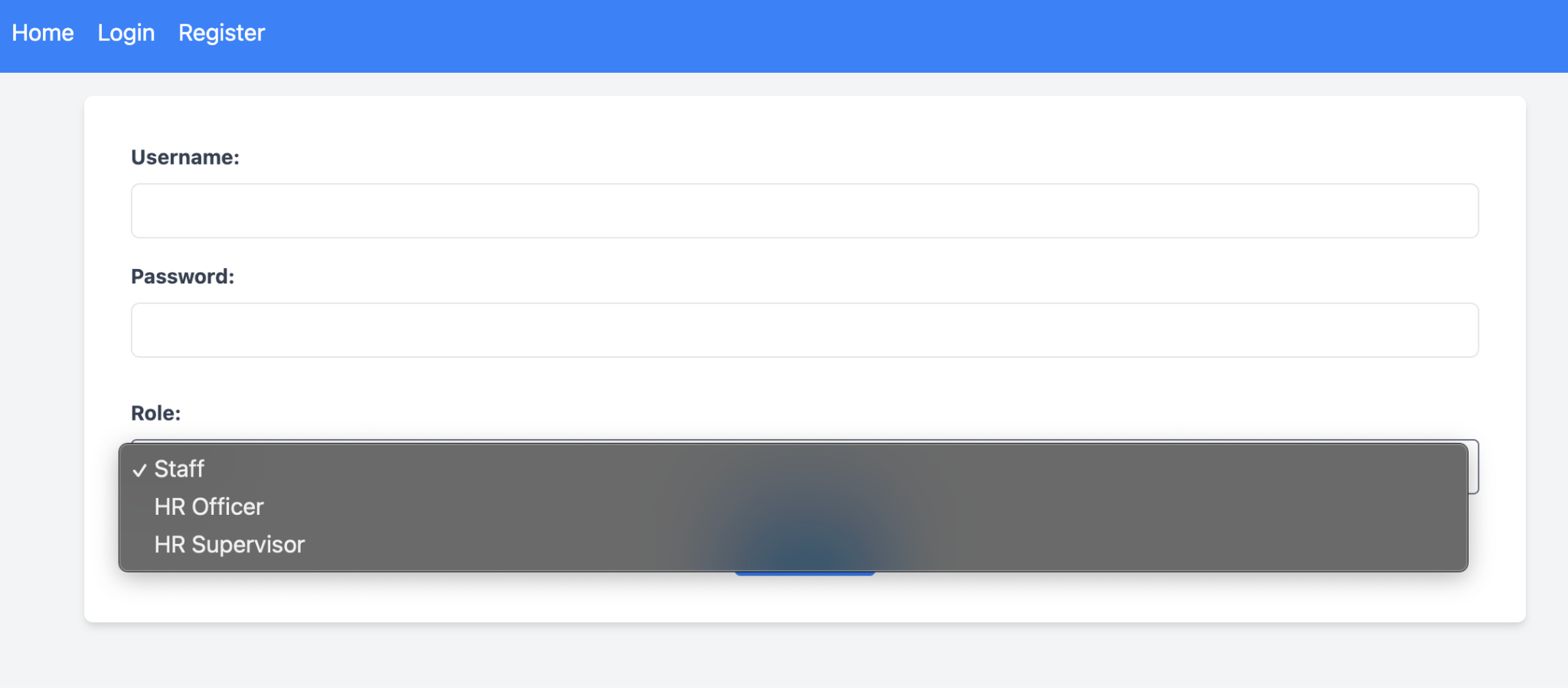
# **1.** **MVP Product Demonstration**

*Authentication*

1. *Click ‘Register’ and it will show Register Page*

****

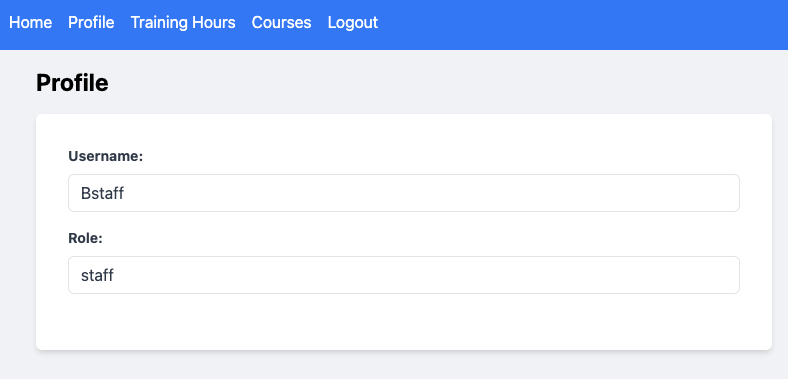
1. *Type in your Username and Password and select your Role (Staff, HR officer, HR supervisor) and click ‘Register’*

**

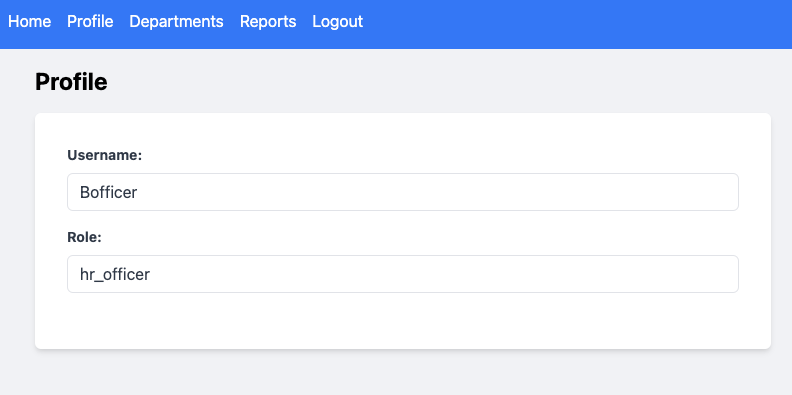
1. *You will come to login page, enter your username and password and click ‘Sign in’*

# 

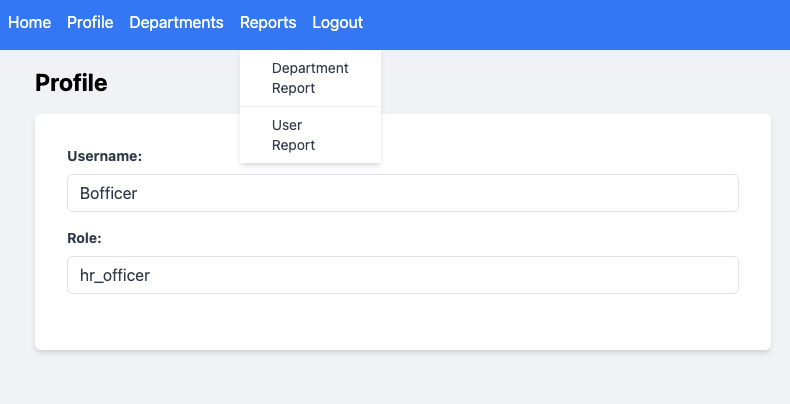
1. Login page - ‘Staff’ role



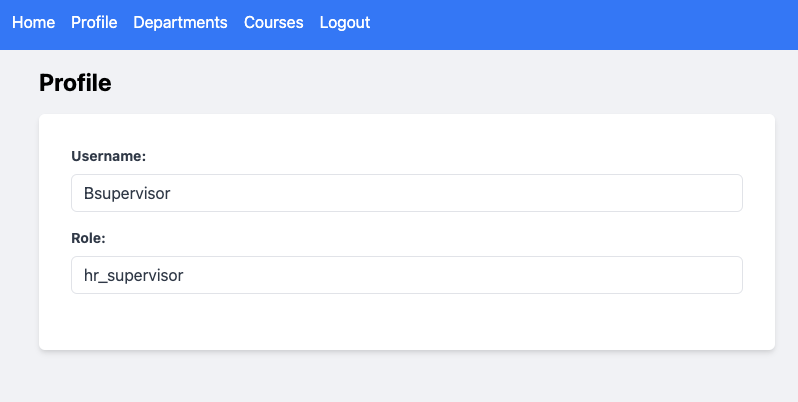
1. Login page - ‘HR officer’ role



1. ‘HR officer’ when click ‘Reports’ from the navigation, a dropdown menu will show with two options: ‘Department Report’ & ‘User Report’



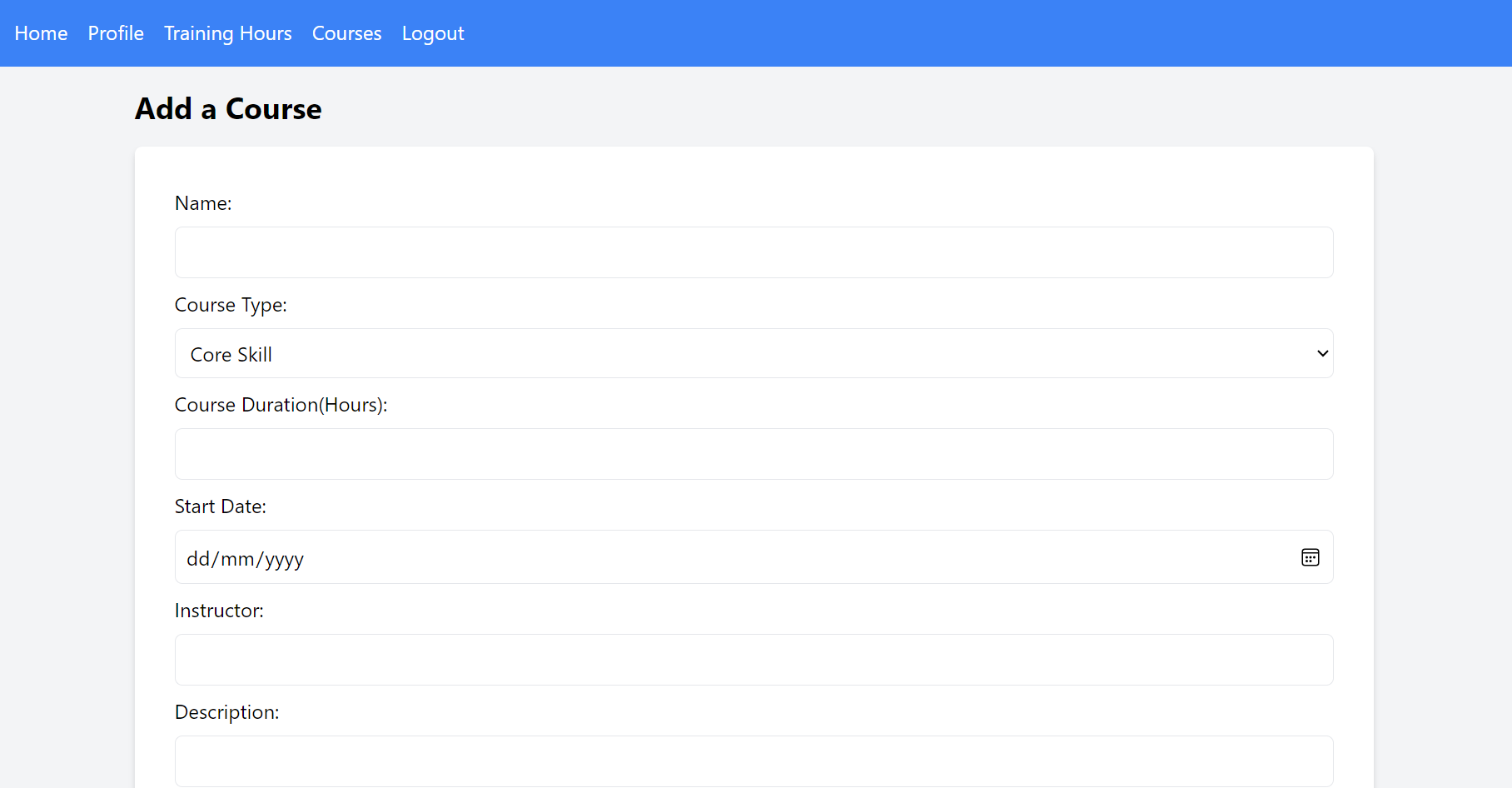
1. Login page - ‘HR supervisor’ role



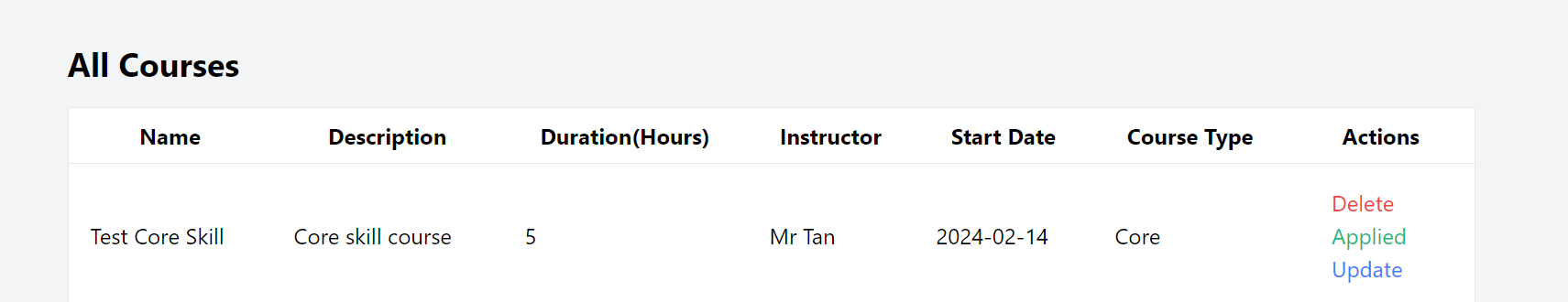
# 

Create Courses

1. Navigate to Courses from the header, the ‘Add a course’ form is shown



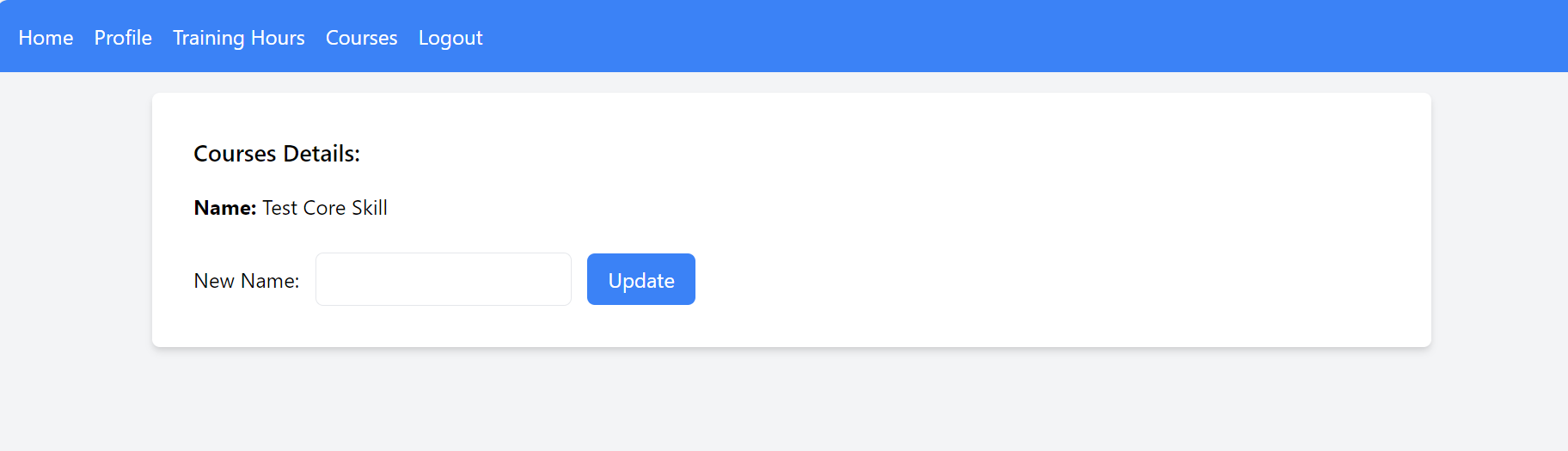
1. Populate all mandatory fields and click submit, the course will be created at the bottom of the page



Update Courses

1. Navigate to Courses from the header, scroll down to the bottom of the page. Select the courses that you choose to update and click on the Update button under the action column.

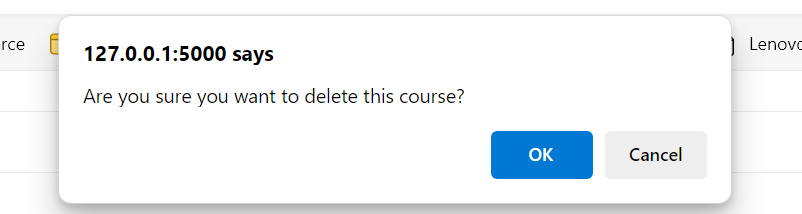
The update page is shown where you can update the name of the selected course



Delete Course

1. Navigate to Courses from the header, scroll down to the bottom of the page. Select the courses that you choose to update and click on the Delete button under the action column.

The message is prompted to the user ‘Are you sure you want to delete this course?’

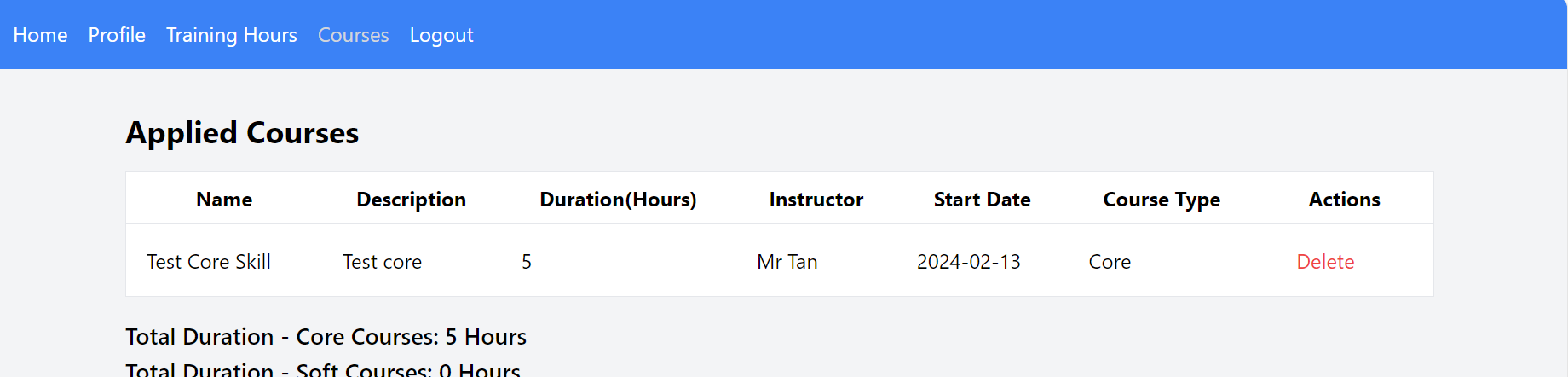


1. Click OK, the course is delete and removed from the list at the bottom of the page.

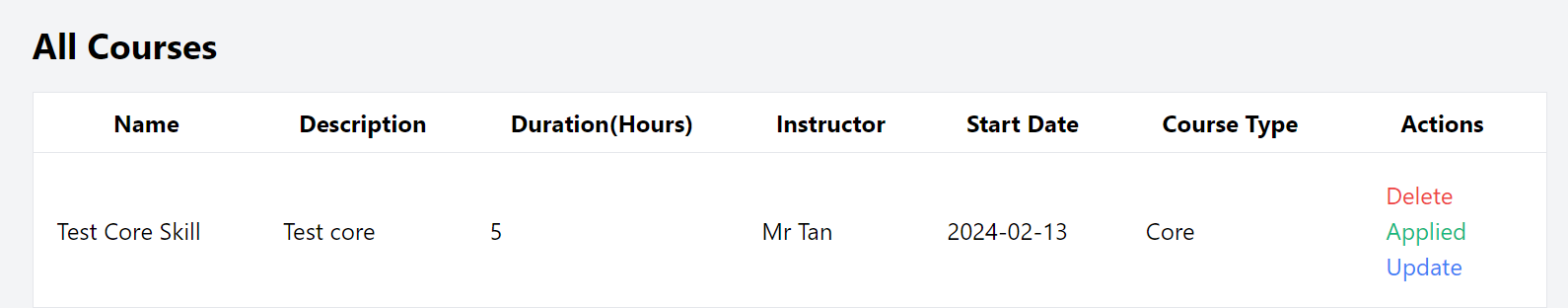
Apply Courses

1. Navigate to Courses from the header, scroll down to the bottom of the page. Select the courses that you choose to Apply and click on the Apply button under the action column.

User will be redirected to the ‘Applied Courses’ page where they can view their applied courses and the total duration of each ‘Core’ or ‘Soft’ skills courses which they had applied



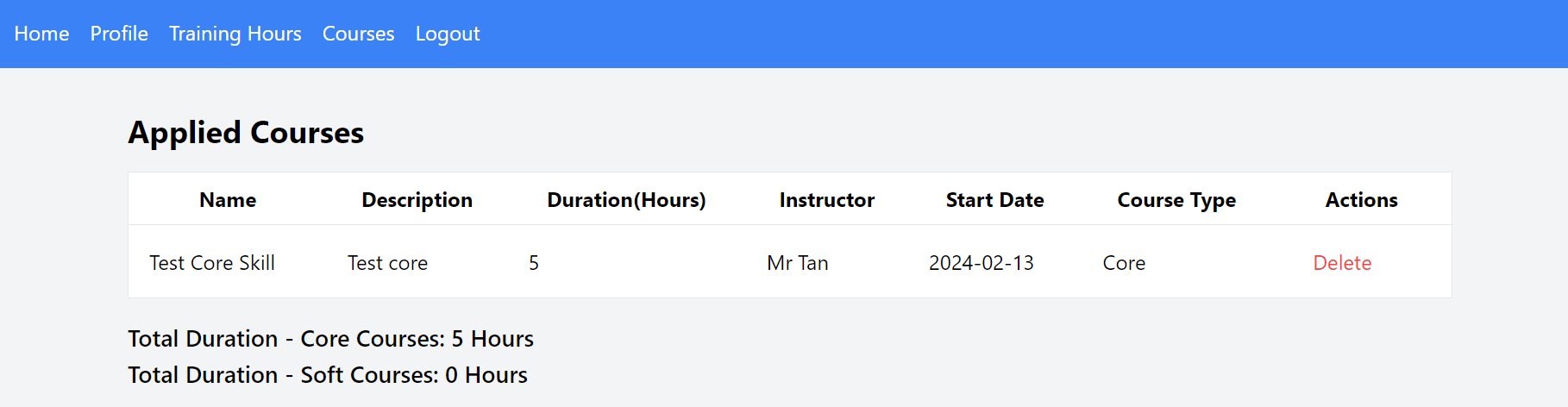
1. Navigate back to the Courses list and the applied course button will be disabled and shown as ‘Applied’



Delete Applied course

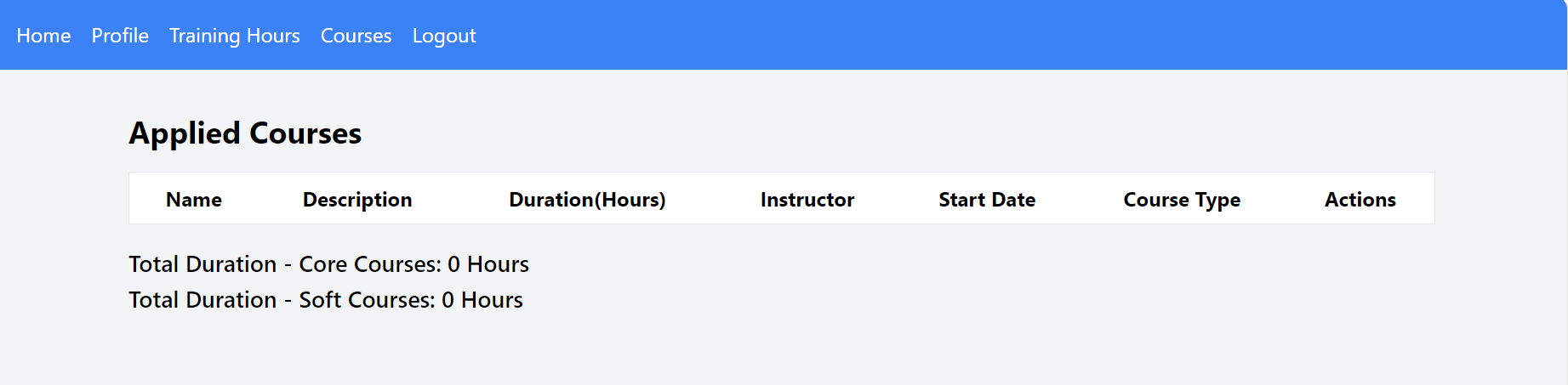
1. Navigate to Training Hours from the header

The same ‘Applied course’ page is shown with the user’s previously applied course

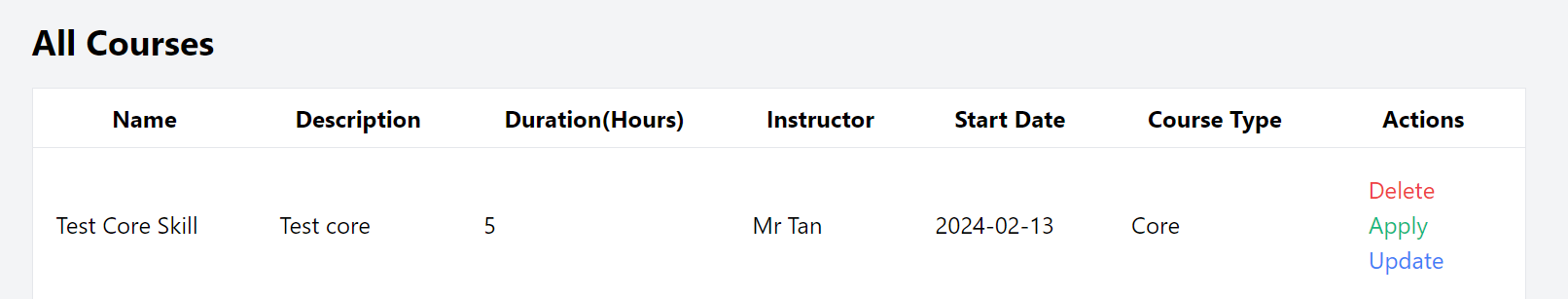


1. Select the course to delete and click on the delete button under the action column

The applied course is removed and the total hours is recalculated

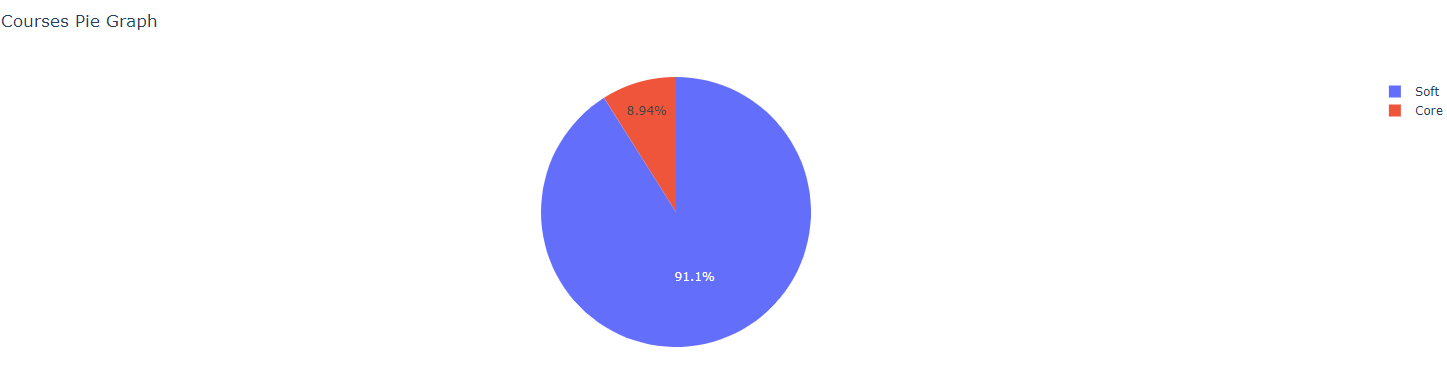
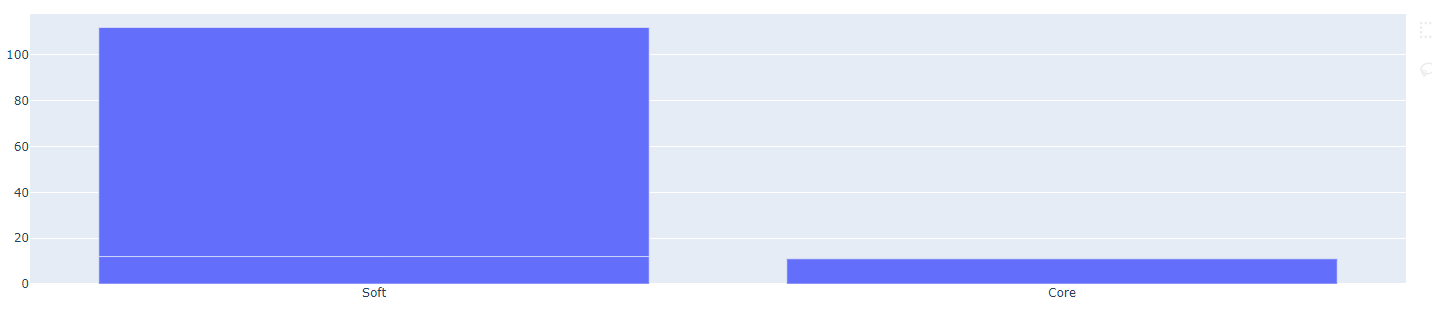


1. Navigate back to the Courses page from the header and the apply button is reenabled on the delete applied course



Graphs

1 Once data has been added, the graphs will visually represent it in the dashboard (homepage)



# 

# **2.** **MVP Requirements and Risk Log**

*Insert the content or screenshot of your team’s* ***RequirementsAndRiskLog.md*** *document here.*

***Requirements and Risk Log***

*In this markdown document, list all of the requirements that you have identified for your product so far.*

***Requirements***

*List several user stories (no more than 8) that describe the product requirements. For each user story that you include:*

**User stories:**

**Staff**

* As a Staff, I want to track my training hours in core technical and soft skills, so I can ensure I meet the yearly requirements
* As a staff, i would like my attendance and details to be recorded
* As a staff , i would like to view my own training hours, how much they are lacking for the year for each category of skills.

**HR Officer**

* As a HR Officer, I need to adjust training hours and percentages for technical and soft skills for staff, so I can tailor the training needs to department requirements.
* As a HR officer, i would like to generate a current report for a particular department to show the number of staff who have completed their obligations as well as staff who are still falling behind in the training hour

**HR Supervisor**

* As HR Supervisor, I want to manage HR officers and add new courses, so that the training offerings remain relevant and up-to-date.

**Requirements:**

* Secure login for STaff, HR Officers and HR supervisors
* Dashboard for users to view training hours and requirements
* Functionality for HR officers to adjust training settings.
* Database to store training hours, courses and user information
* Reporting tools for HR officers and supervisors
* *Paste in the Story Description and Story Summary from Jira. Attempt to keep story summaries in the form taught in the lectures, i.e. the form "As a [persona], I [want to], [so that]". Feel free to update your Jira issues into this form if they are not already.*
* *Give details of whether this story has been assigned to someone yet, and whether it is completed yet.*
* *Include a screenshots of any relevant attachments.*

*User registration*

[*https://ce201.atlassian.net/browse/CE2-32*](https://ce201.atlassian.net/browse/CE2-32)

*User login*

[*https://ce201.atlassian.net/browse/CE2-31*](https://ce201.atlassian.net/browse/CE2-31)

*Create user*

[*https://ce201.atlassian.net/browse/CE2-26*](https://ce201.atlassian.net/browse/CE2-26)

*Update user*

[*https://ce201.atlassian.net/browse/CE2-25*](https://ce201.atlassian.net/browse/CE2-25)

*Delete user*

[*https://ce201.atlassian.net/browse/CE2-27*](https://ce201.atlassian.net/browse/CE2-27)

*Create courses*

[*https://ce201.atlassian.net/browse/CE2-16*](https://ce201.atlassian.net/browse/CE2-16)

*Update courses*

[*https://ce201.atlassian.net/browse/CE2-18*](https://ce201.atlassian.net/browse/CE2-18)

*Delete courses*

[*https://ce201.atlassian.net/browse/CE2-19*](https://ce201.atlassian.net/browse/CE2-19)

*Apply courses*

[*https://ce201.atlassian.net/browse/CE2-37*](https://ce201.atlassian.net/browse/CE2-37)

*Delete Applied courses*

[*https://ce201.atlassian.net/browse/CE2-11*](https://ce201.atlassian.net/browse/CE2-11)

*Read all applied courses*

[*https://ce201.atlassian.net/browse/CE2-9*](https://ce201.atlassian.net/browse/CE2-9)

*Graphs*

[*https://ce201.atlassian.net/browse/CE2-34*](https://ce201.atlassian.net/browse/CE2-34)

*In addition to user stories (which is the main thing we are trying to teach for agile), try to include one or two other requirement modelling techniques, e.g. as listed in the lecture.*

*Try to group the requirements under two sub-headings: functional requirements, and non-functional requirements.*

*When marking this section we will be looking to see your team has understood the correct way to represent User-Stories and one or more other requirement modelling methods, and the requirements listed are giving as much relevant information for the development team as possible.*

***Risk Log***

*In this section list the risks you have identified for your project. For each Risk identified:*

***Technical Challenges and Learning Curve***

* ***Description:*** *Encountering technical issues or team members facing a steep learning curve with scripting*
* ***Impact:*** *Medium*
* ***Likelihood:*** *Low*
* ***Mitigation:*** *Peer to peer teaching and guidance, encourage pair programming. Assign tasks based on individual strengths and provide access to online resources.*

***Scope Creep***

* ***Description:*** *Changes in project scope leading to delays and resource overruns.*
* ***Impact:*** *High*
* ***Likelihood:*** *Low*
* ***Mitigation:*** *Clearly define project scope and requirements. Use Agile methodologies to manage changes effectively. Regularly review project progress against original goals and adjust timelines and resources as needed.*

***Tight Deadlines***

* ***Description:*** *Unrealistic timelines leading to rushed work and potential quality issues.*
* ***Impact:*** *High*
* ***Likelihood:*** *High*
* ***Mitigation:*** *Develop a realistic project timeline with input from all team members. Include buffer time for unforeseen delays. Regularly review and adjust deadlines as necessary.*

***Team Collaboration Issues***

* ***Description:*** *Ineffective communication or collaboration within the team.*
* ***Impact:*** *Medium*
* ***Likelihood:*** *High*
* ***Mitigation:*** *Establish clear communication channels and regular meetings. Use collaboration tools effectively. Foster a team culture that encourages open communication and support.*

*When marking this section we will be looking to see several realistic risks have been noted, and are actively being tracked and mitigated against.*

3. MVP Project Management Log

***Team Introduction and Communication Protocol***

When we started, although we were strangers, we made it a point to introduce ourselves and set up a simple rule to follow. We decided to have short Zoom meetings outside of class to quickly catch up, keep each other updated, and offer help with any problems. We also established a texting system to ask questions or seek assistance, even if we couldn't talk on the phone.

***Getting Familiar with Scrum***

We realized that becoming more familiar and experienced with how scrum works would benefit our project. This would ensure that our meetings are efficient and no time is wasted, allowing us to stay on track with our team goals.

***Addressing Communication Challenges***

Like any new project involving unfamiliar faces, we encountered some communication issues initially. However, we promptly addressed them by making sure everyone understood their tasks before wrapping up our meetings, improving overall team communication and collaboration.

Sprint Burndown Charts

*In this section include a screenshot of a sprint-burndown chart for each of your completed sprints.*

*For each sprint, include links to the standup meetings (recorded on Jira, hopefully) associated with each sprint.*

## Burndown-Charts Discussion

*Add a discussion of what the burndown charts indicate, and how your team might improve their velocity / estimation ability in future sprints.*

## 

## 

## 

## 

## 

## Product Backlog

## Other Areas

*If your team has anything else they want to demonstrate, e.g. advanced project management tools, reporting graphs with suitable explanation and interpretation, advanced use of Jira (e.g. issue links, epics, versions), or continuous integration tools, then include a summary here.*

4. MVP Team Effort Log

*Insert the content or screenshot of your team’s* ***TeamEffortLog.md*** *document here.*

***Timothy***

Sprint 1

#### 1. Project Setup:

* Getting the project set up took a good chunk of time, but it was super important because it lays the groundwork for everything else. We had to pick out the best tools and frameworks that fit our needs, like Python with Flask, SQLite3, Jinja, Tailwind and Github.

#### 2. CRUD:

* Added a fully functioning CRUD (Create, Read, Update, Delete) sample that could serve as a standard reference for everyone involved. This is crucial in ensuring uniformity across the codebase.

#### 3. Graphs:

* Integrated graphs to visually represent data for users interacting with the web application, leveraging its ability to retrieve data from the database and display it in a more understandable format.
* The entire setup and completion were wrapped up neatly within Sprint 1, taking about 6 hours in total, which included both research and implementation phases.

***Bernard***

Sprint 1

1. Register:

Users need to input their username, password and select their roles (staff, HR officer, HR supervisor)

After registering, it will store the user's information (ID, name, role, password (hash)) in the User table and go to the login page

User table

ID(primary key)

Name

Role

password\_hash

2. Login:

Users need to input their username and password and click the login button to log in.

If input the wrong username or password, an error message “Wrong username or password” will show on the login page.

3. Profile:

After login, users can click ‘Profile’ on the navigation bar and see their username and role.

4. Roles:

For different roles, after login will show different selections in the navigation bar based on their role.  
Staff role: Home, Profile, Training Hours, Courses and Logout

HR officer: Home, Profile, Departments, Reports(Department report and User report) and Logout

HR supervisor: Home, Profile, Departments, Courses and Logout

***Hangman***

Sprint 1

CRUD Skills

#### 1. Course Application:

Users can apply for courses by selecting from a list of available courses. The application process involves capturing user details and the desired course.

#### 2. Application Retrieval:

The system enables users to view a list of courses they have applied for. This feature provides transparency and allows users to keep track of their course applications.

#### 3. Application Withdrawal:

To accommodate changing preferences, users can withdraw their applications for specific courses.

Training hours calculation

Users are able to track their total training hours on both core and soft skills through the applied course page

UserCourses Table

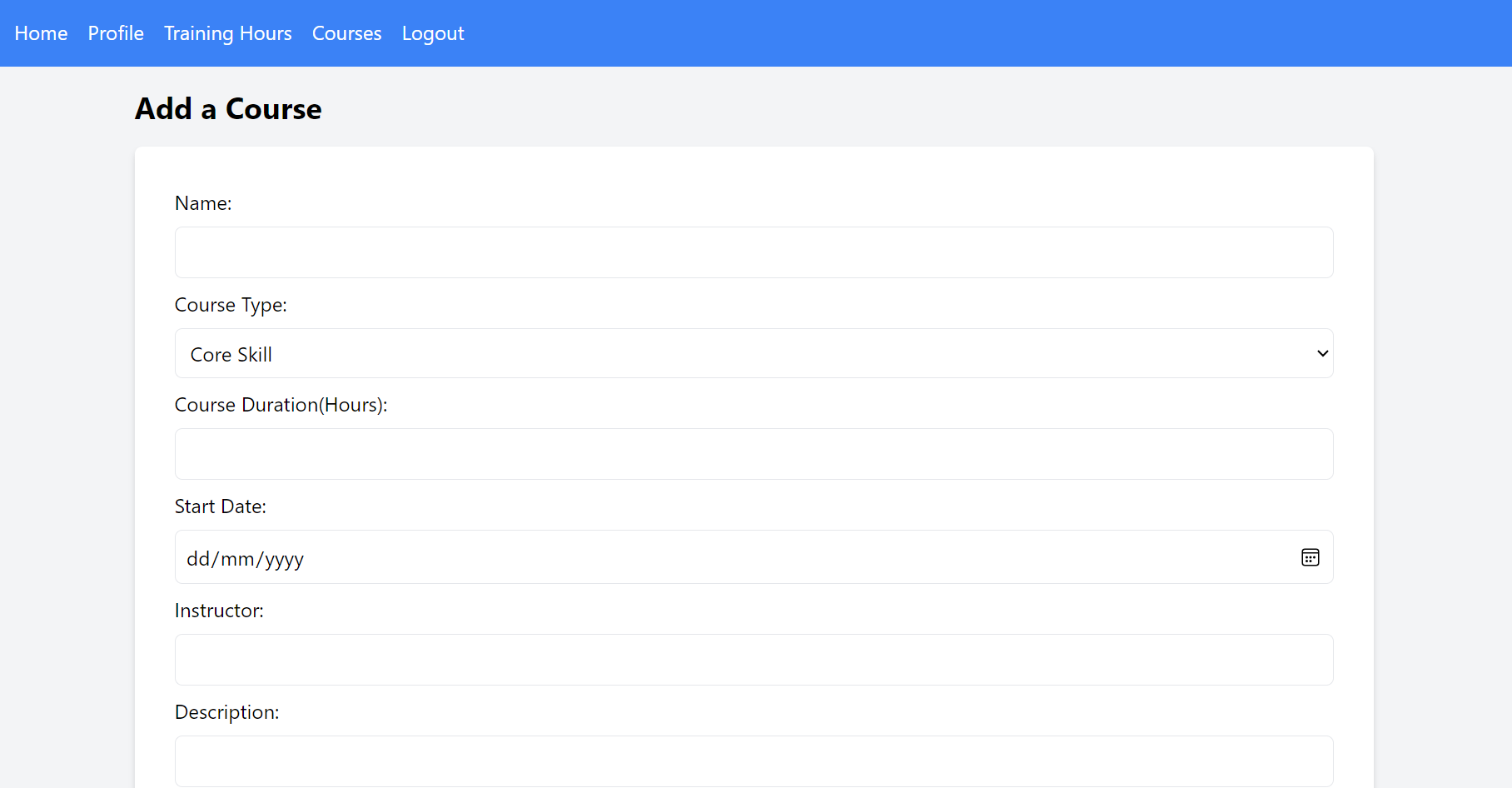
* Id (Primary key)
* User\_id
* Name
* Duration
* Course\_id
* Course\_type
* Foreign key (user\_id) reference User table (id)
* Foreign key (course\_id) reference Courses (id)

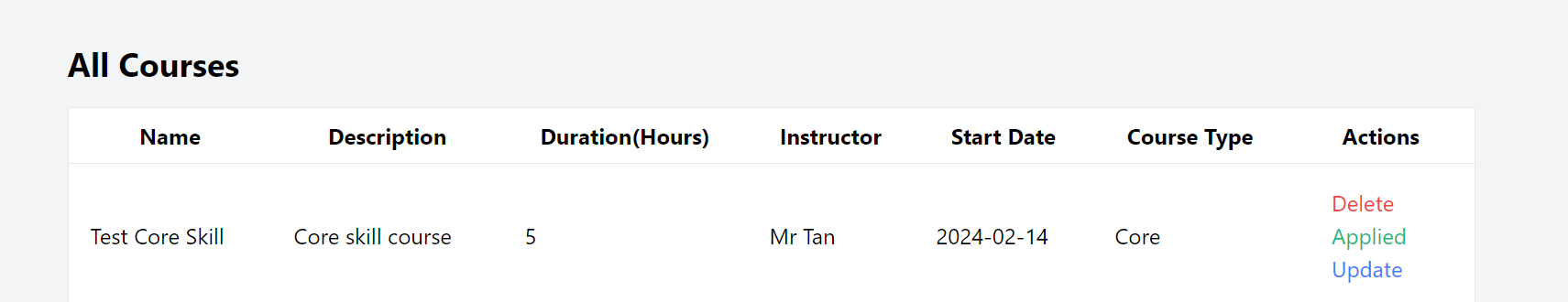
***Ying Li***

Sprint 1

CRUD Courses

The user is able to create the course

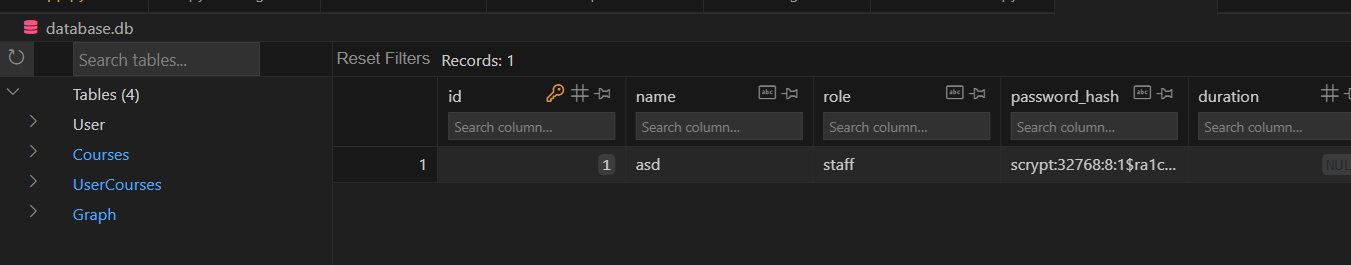




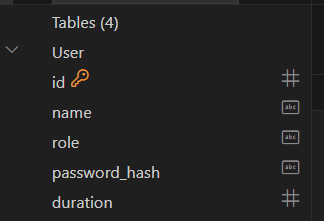
Delete or update the course

### **Database Schema:**

The database schema is structured to support user-course interactions efficiently. Key entities include:



The tables are User , Courses, UserCourses and Graph

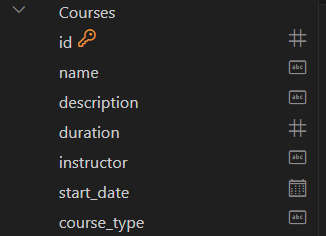


The user table stores

id as the primary key

Attributes

name , role , password (hashed) , duration

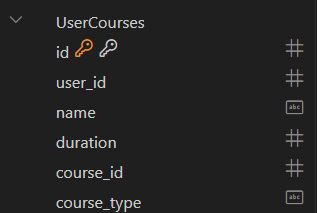


The Courses table stores

id as the primary key

Attributes

name , description , duration , instructor , start\_date, course\_type

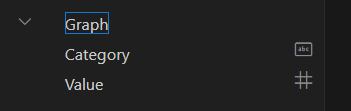


The UserCourses table stores

id as the primary key

Attributes

user\_id , name , duration , course\_id , course\_type



The Graph table

Attributes

Category

Value

***Document Instructions:***

*This document should contain the full content or screenshot of your team’s markdown documentation. Edit this document, replacing all italicised text.*

*All markdown documents should be developed collaboratively by the team, using the GitHub.*

*When following markdown templates, ensure that all section headings are included fully.*

*For the MVP write-up in the four markdown documents, base your information up till the MVP Demonstration on lesson 5.*

*A minimum word count of 1,000 words is expected.*