Table of Contents

[Problem statement 5](#_Toc100596129)

[Objective 5](#_Toc100596130)

[Scope 5](#_Toc100596131)

[Target audience 5](#_Toc100596132)

[Existing system that are in use 5](#_Toc100596133)

[Project Duration 6](#_Toc100596134)

[Algorithm 7](#_Toc100596135)

[Flowchart for the system 7](#_Toc100596136)

[Updated flowchart design 9](#_Toc100596137)

[ERD 10](#_Toc100596138)

[EERD 10](#_Toc100596139)

[UML use case 11](#_Toc100596140)

[Technology to implement on the system 12](#_Toc100596141)

[Technology 12](#_Toc100596142)

[IDE to build system 12](#_Toc100596143)

[Interface design 13](#_Toc100596144)

[Updated interface design 15](#_Toc100596145)

[Storyboard 17](#_Toc100596146)

[Interface 18](#_Toc100596147)

[Main page 18](#_Toc100596148)

[Course page 18](#_Toc100596149)

[Session time selection 18](#_Toc100596150)

[Fill in form 19](#_Toc100596151)

[Confirmation popup 19](#_Toc100596152)

[Summary page 19](#_Toc100596153)

[SQL 20](#_Toc100596154)

[Table course 20](#_Toc100596155)

[Table session 21](#_Toc100596156)

[Feedback on the system 22](#_Toc100596157)

[Version 2.0 alpha 22](#_Toc100596158)

[Interface version 2.0 alpha 22](#_Toc100596159)

[Main page 22](#_Toc100596160)

[Register page 23](#_Toc100596161)

[Login page 23](#_Toc100596162)

[Logged in front page 23](#_Toc100596163)

[Course page 24](#_Toc100596164)

[Time session for subject 24](#_Toc100596165)

[Summary page 25](#_Toc100596166)

[Session history check 25](#_Toc100596167)

[Page code 26](#_Toc100596168)

[Heading.jsp 26](#_Toc100596169)

[Index.jsp 26](#_Toc100596170)

[Login.jsp 27](#_Toc100596171)

[Register.jsp 27](#_Toc100596172)

[Course.jsp 28](#_Toc100596173)

[Timesession.jsp 28](#_Toc100596174)

[Summary.jsp 29](#_Toc100596175)

[Sessionhistory.jsp 29](#_Toc100596176)

[ERD for version 2.0 alpha 30](#_Toc100596177)

[UML use case 31](#_Toc100596178)

[Initial design for v2.0a 31](#_Toc100596179)

[Real system use case 31](#_Toc100596180)

[SQL 32](#_Toc100596181)

[Class\_session table 32](#_Toc100596182)

[Users table 32](#_Toc100596183)

[Course table 33](#_Toc100596184)

[Session\_record table 34](#_Toc100596185)

[SQL Procedures 34](#_Toc100596186)

[Find\_user\_record 34](#_Toc100596187)

[This\_year\_session procedure 35](#_Toc100596188)

[Test plan 36](#_Toc100596189)

[Image evidence 36](#_Toc100596190)

[Image 101 36](#_Toc100596191)

[Image 102 36](#_Toc100596192)

[Library integrated in the system 37](#_Toc100596193)

[Update version 3.0 alpha version 38](#_Toc100596194)

[UML Use Case version 3.0 38](#_Toc100596195)

[ERD version 3.0 39](#_Toc100596196)

[Flowchart 40](#_Toc100596197)

[User flowchart 40](#_Toc100596198)

[Register course 41](#_Toc100596199)

[Check session history flowchart 42](#_Toc100596200)

[Profile page 43](#_Toc100596201)

[Admin page 44](#_Toc100596202)

[SQL version 3.0 45](#_Toc100596203)

[Class\_session table 45](#_Toc100596204)

[Users table 45](#_Toc100596205)

[Course table 46](#_Toc100596206)

[Session\_record table 47](#_Toc100596207)

[SQL Procedures 47](#_Toc100596208)

[Find\_user\_record 47](#_Toc100596209)

[This\_year\_session procedure 48](#_Toc100596210)

[Insert\_class\_session 49](#_Toc100596211)

[Check\_else\_insert\_course 49](#_Toc100596212)

[Webpage interface version 3.0 50](#_Toc100596213)

[Main menu 50](#_Toc100596214)

[Login 50](#_Toc100596215)

[Register 51](#_Toc100596216)

[Main page for user 51](#_Toc100596217)

[Main page for admin 51](#_Toc100596218)

[Admin page import spreadsheet 52](#_Toc100596219)

[Import successful page after spreadsheet is uploaded 52](#_Toc100596220)

[Wrong file content 52](#_Toc100596221)

[Course selection page 53](#_Toc100596222)

[Select time session page 53](#_Toc100596223)

[Summary page 53](#_Toc100596224)

[Email page 54](#_Toc100596225)

[Session history page 54](#_Toc100596226)

[Edit profile page 55](#_Toc100596227)

[Change profile 55](#_Toc100596228)

[Source code 3.0 56](#_Toc100596229)

[Heading.jsp 56](#_Toc100596230)

[Index.jsp 56](#_Toc100596231)

[Login.jsp 57](#_Toc100596232)

[Register.jsp 57](#_Toc100596233)

[Import.jsp 58](#_Toc100596234)

[Form.jsp 58](#_Toc100596235)

[Editprofile.jsp 59](#_Toc100596236)

[Timesession.jsp 59](#_Toc100596237)

[Summary.jsp 60](#_Toc100596238)

[Sessionhistory.jsp 60](#_Toc100596239)

[Course.jsp 61](#_Toc100596240)

[Library integrated in the system 3.0 62](#_Toc100596241)

[Justify decisions made how the system will result in an effective solution. 63](#_Toc100596242)

[Future planning for the system 65](#_Toc100596243)

# Problem statement

The training department produce a spreadsheet which lists the courses they run each month.

# Objective

A spreadsheet that contains data will be include into the database.

Allow users to select the training sessions they wish to attend on a form based on a spreadsheet.

The responses the user makes on the form will then create a summary document listing selections made which is emailed to the user.

# Scope

The user for this system will be employee of the company, supervisor of the company training department and the superior management.

# Target audience

This system will be targeted to the employee of training department of the company.

# Existing system that are in use

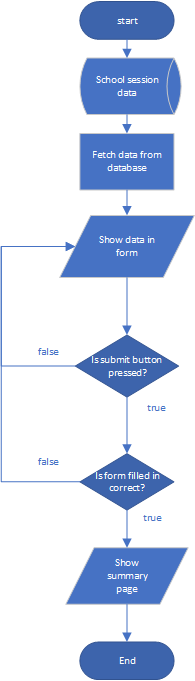
1. Spreadsheet system to provide monthly courses to employee to attend.
2. Employee required to fill the form and submit to the associated supervisor or the management team.
3. The system will help to digitalise the whole process.

# Project Duration

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Phase | No | Activity | Week 1 | Week 2 | Week 3 |
| Phase 1 | 1 | Gathering requirements and information of the training department |  |  |  |
| 2 | Identify the problem |  |  |  |
| 3 | Estimate project duration |  |  |  |
| Phase 2 | 1 | Prepare algorithm for project development |  |  |  |
| 2 | Provide interface design |  |  |  |
| 3 | Update and changes |  |  |  |
| Phase 3 | 1 | Implement solution |  |  |  |
|  |  |  |
|  |  |  |
| Phase 4 | 1 | Test plan |  |  |  |
| 2 | Optimise and update |  |  |  |
| Phase 5 | 1 | Writing report |  |  |  |
| Phase 6 | 1 | Show products |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# Algorithm

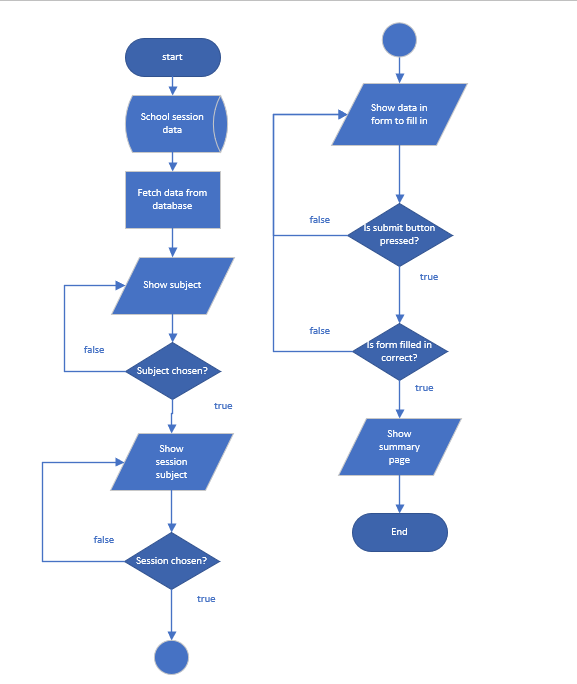
## Flowchart for the system



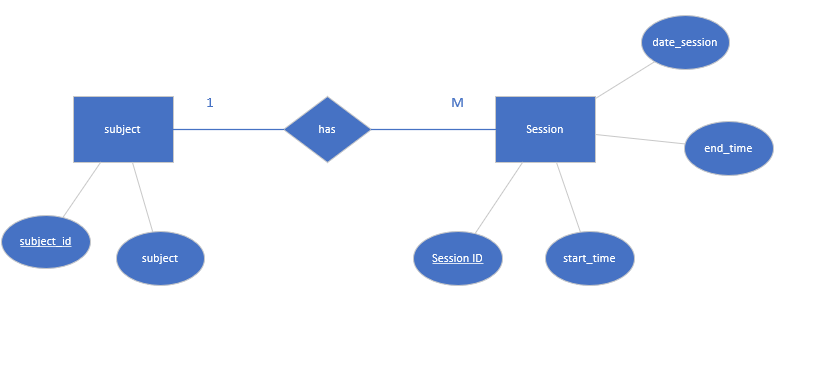
I show this flowchart to my friend, I received feedback and made an improvement to my flowchart.

Flowchart above contain some problem as the subject time is not consistent every time. This might be harder for development. It is better to show the subject first, then the session time for it. At last page is a form before summary, the date and time for the subject in the month will be shown.

## Updated flowchart design

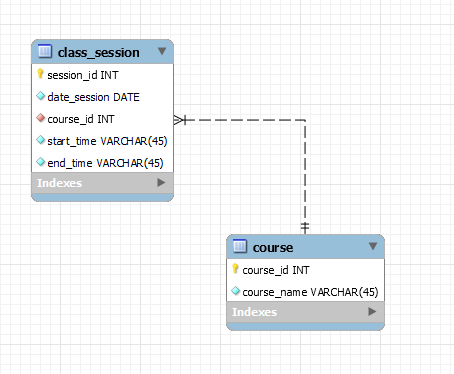


# ERD

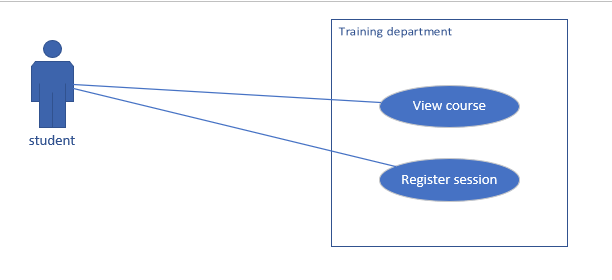


1 subject can have many sessions. 1 session can only have 1 subject.

## EERD



# UML use case



# Technology to implement on the system

## Technology

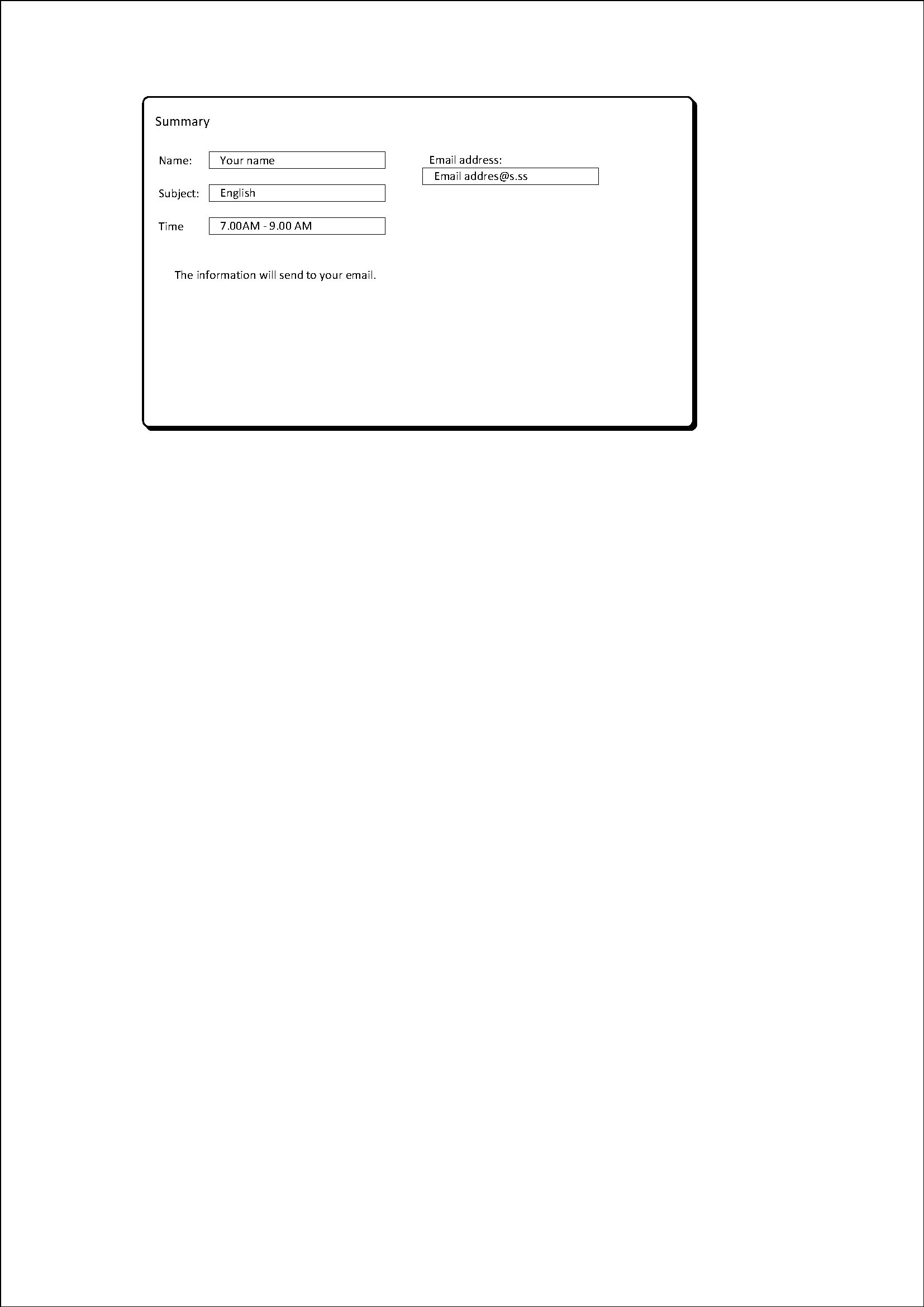
* Java Programming language
* JSTL
* Servlet
* Javascript
* HTML/JSP
* CSS

## IDE to build system

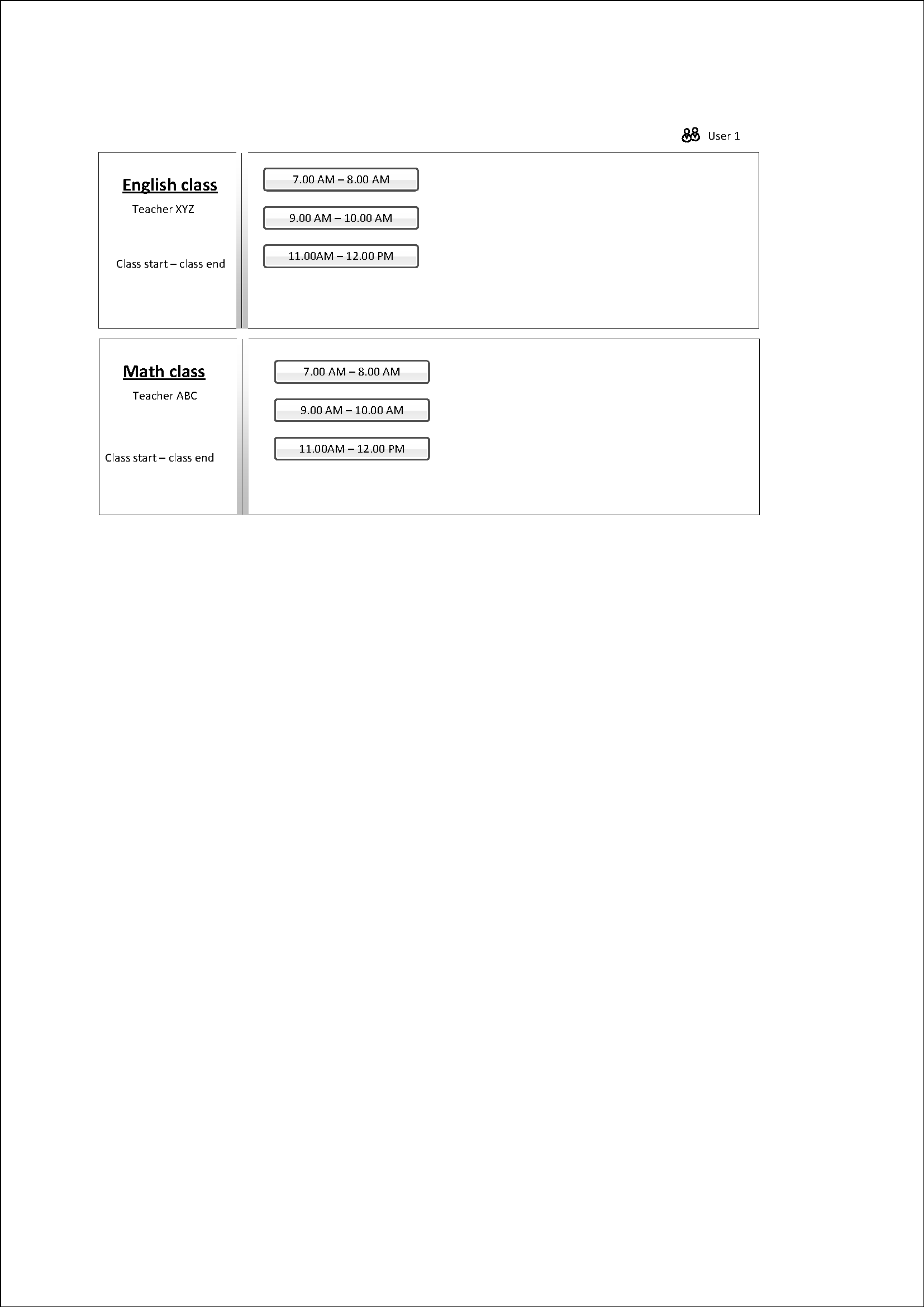
* Ecplise IDE
* MySQL Workbench
* Visual Studio Code

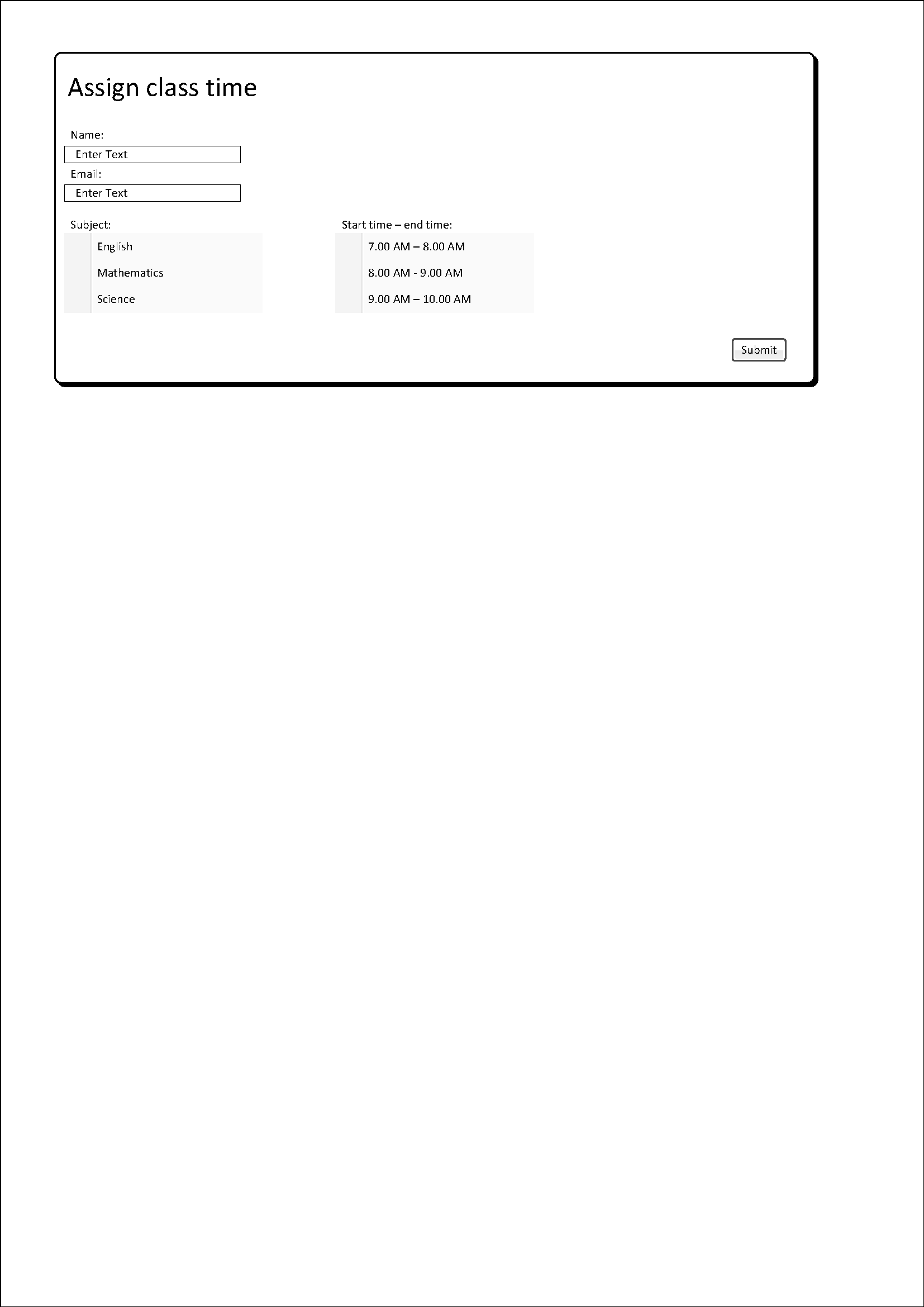
# Interface design

Summary page

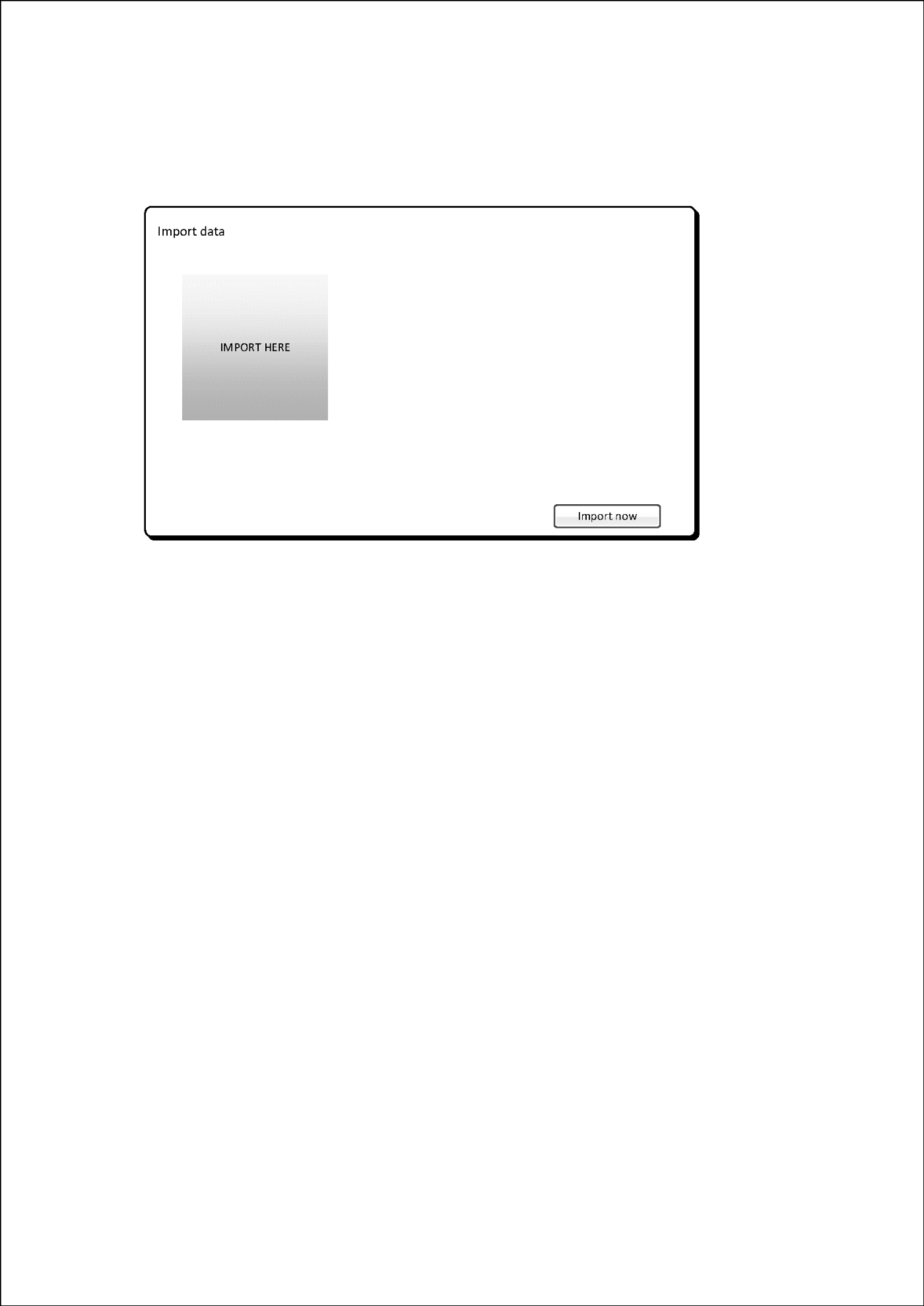


Selection class design 1- login/signup required

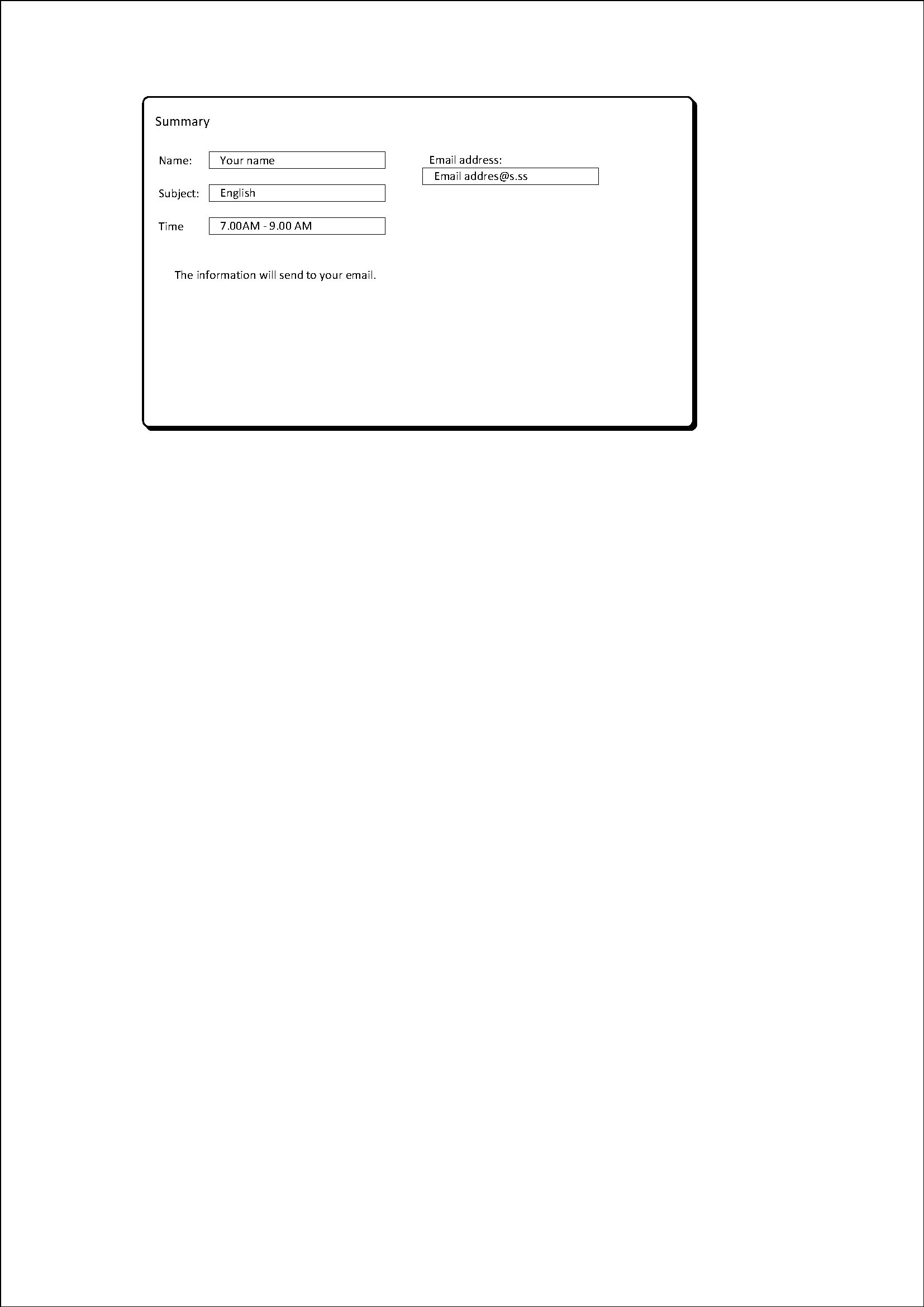


Selection class design 2- login/signup not required (form based)

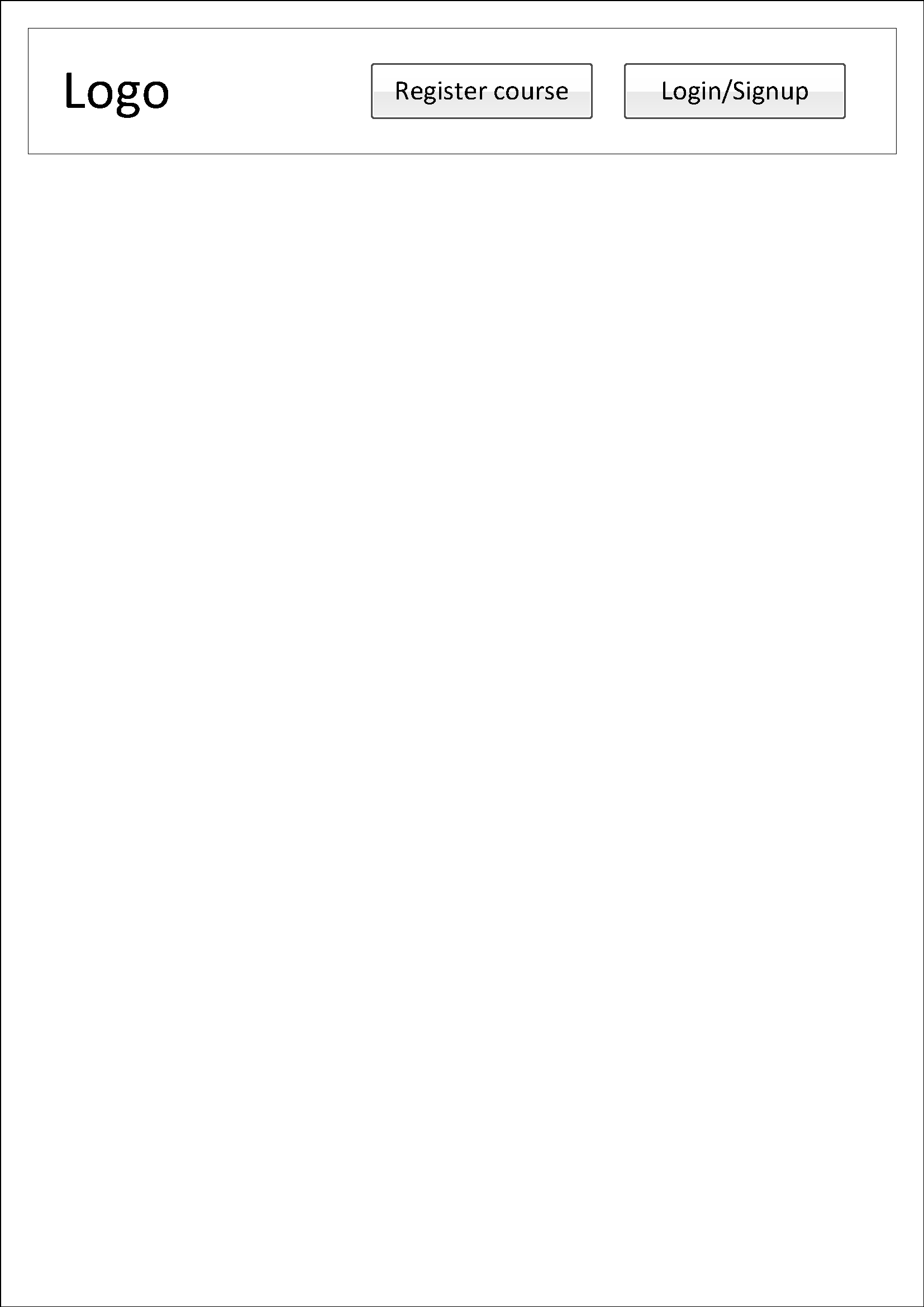
Import page – optional



Summary page - design 2



Heading with login/signup - optional



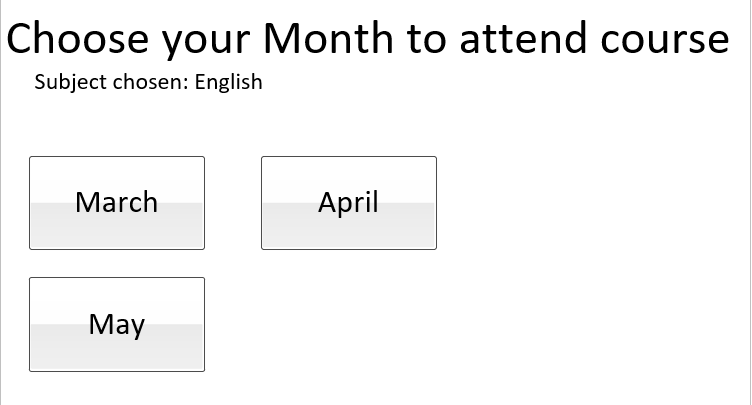
Heading 

## Updated interface design

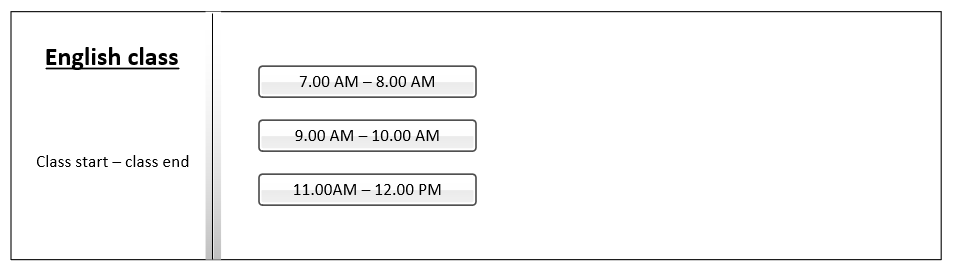
Main page



Choose the month to attend the course after selecting the subject



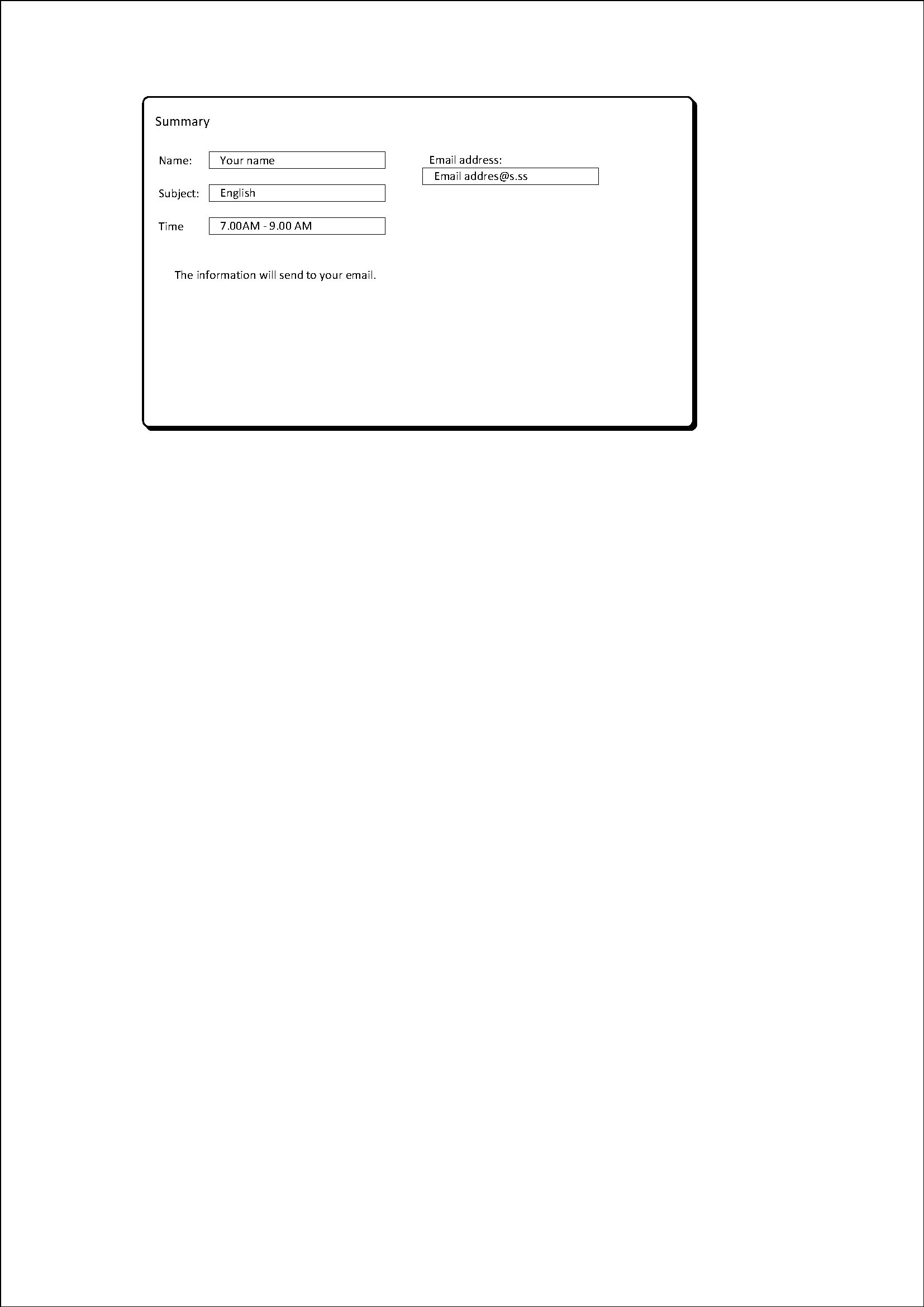
Page after subject selection



Form page after time selection

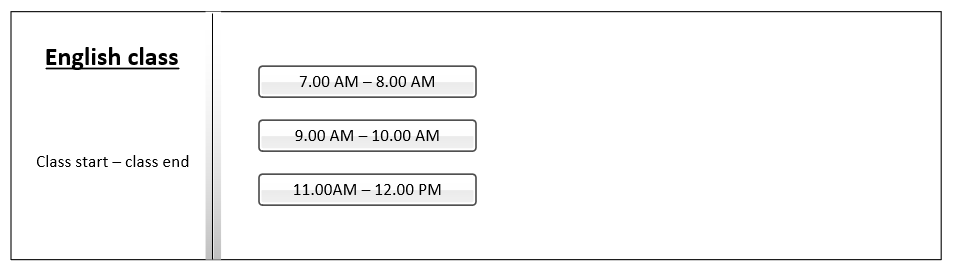


Summary

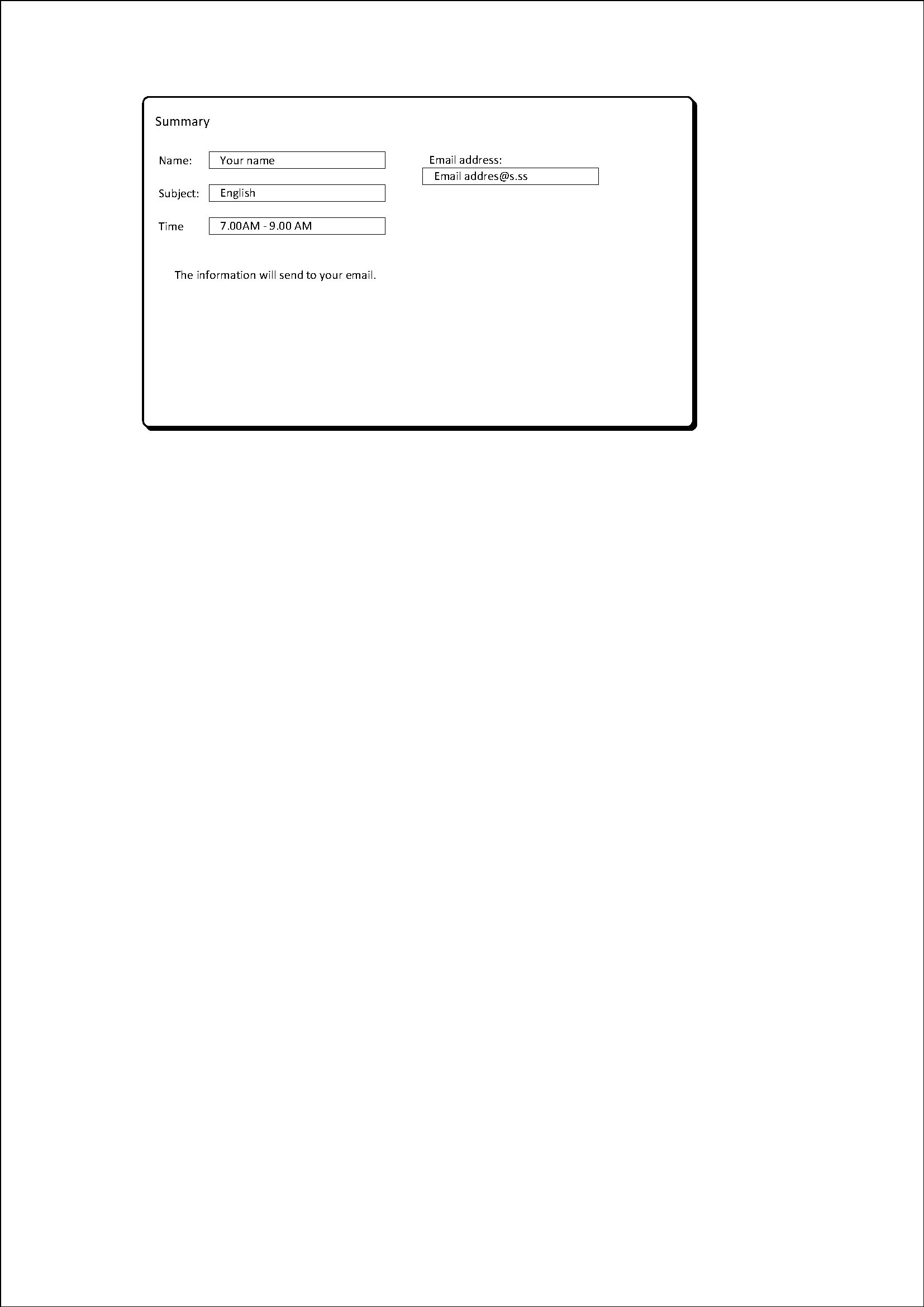


# Storyboard



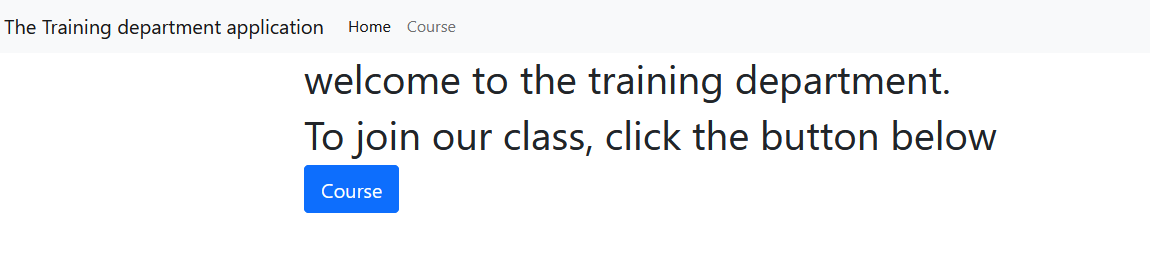




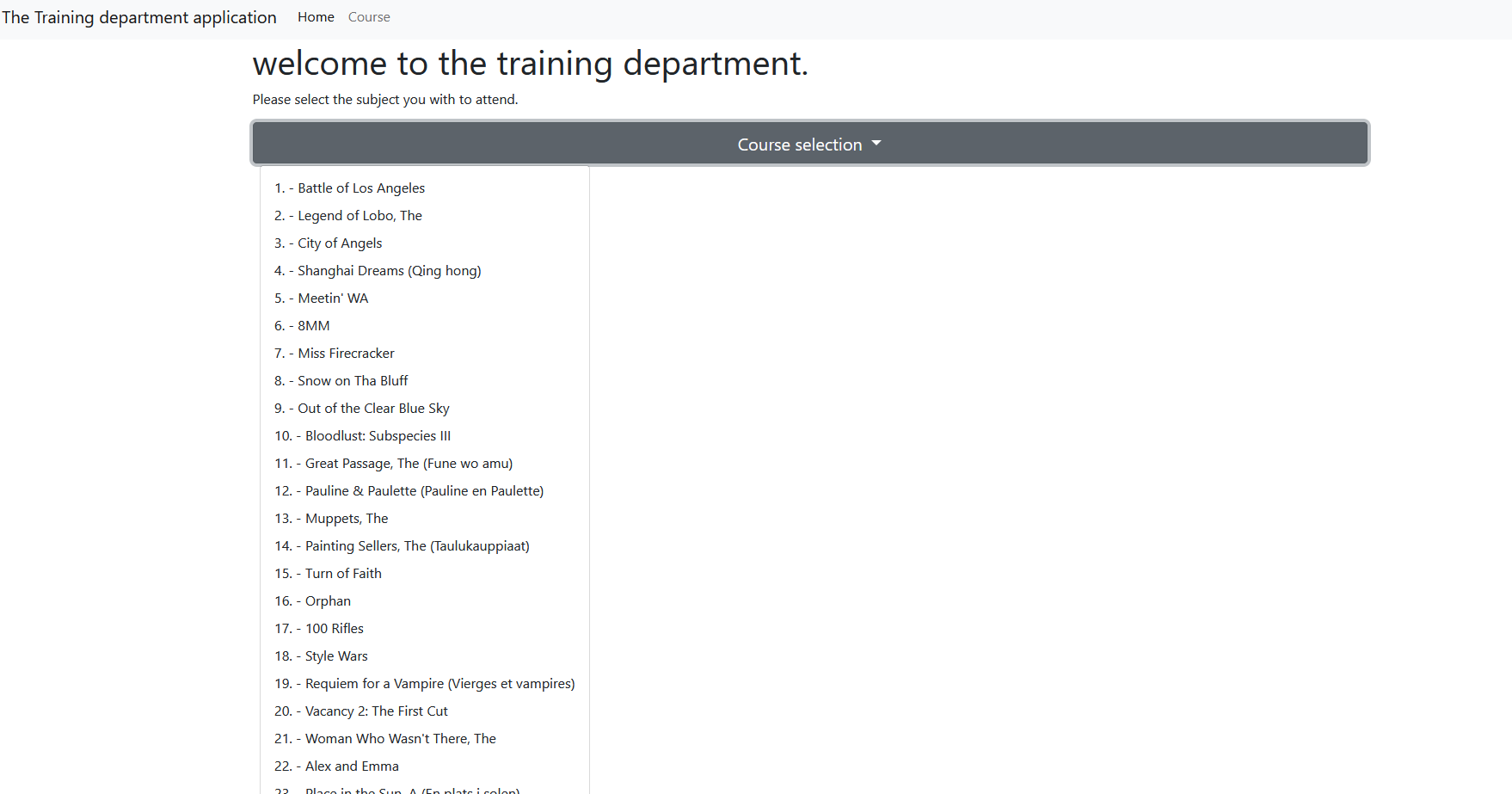


# Interface

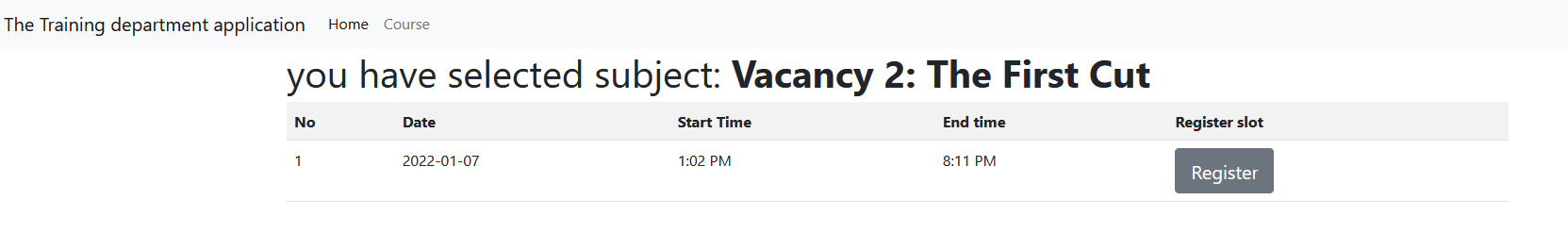
## Main page



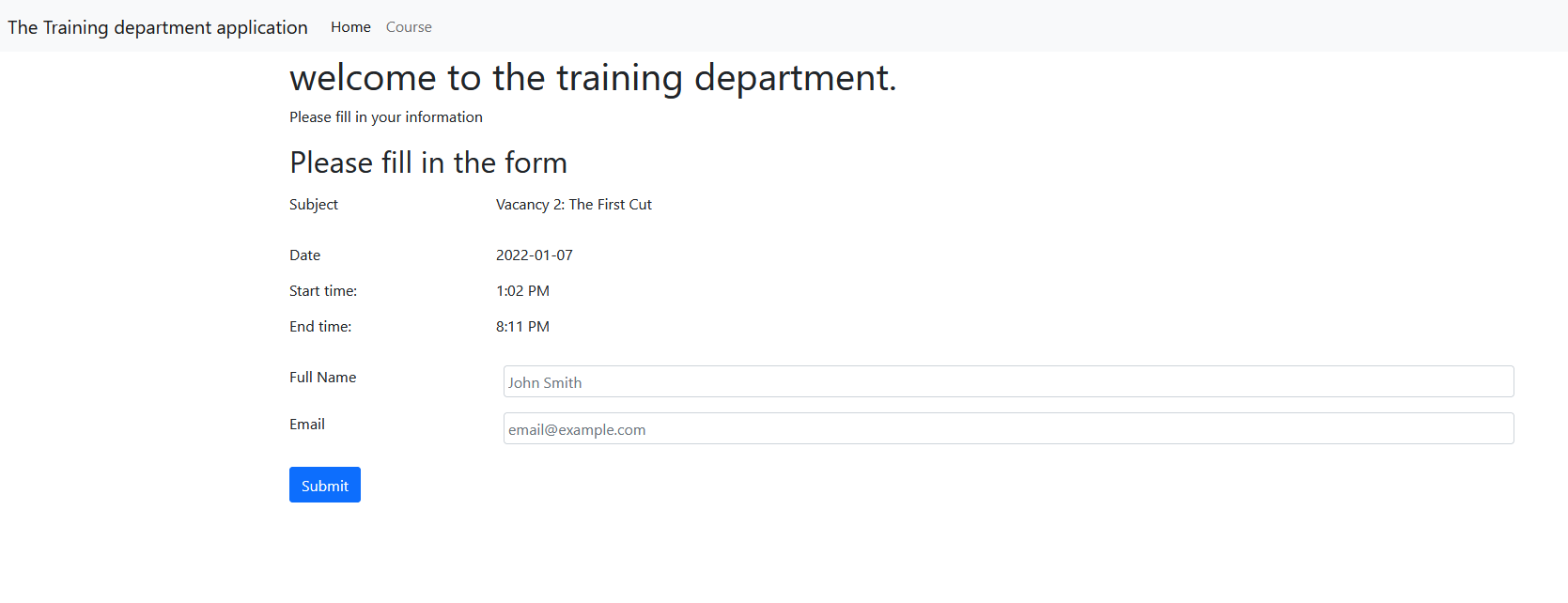
## Course page



## Session time selection



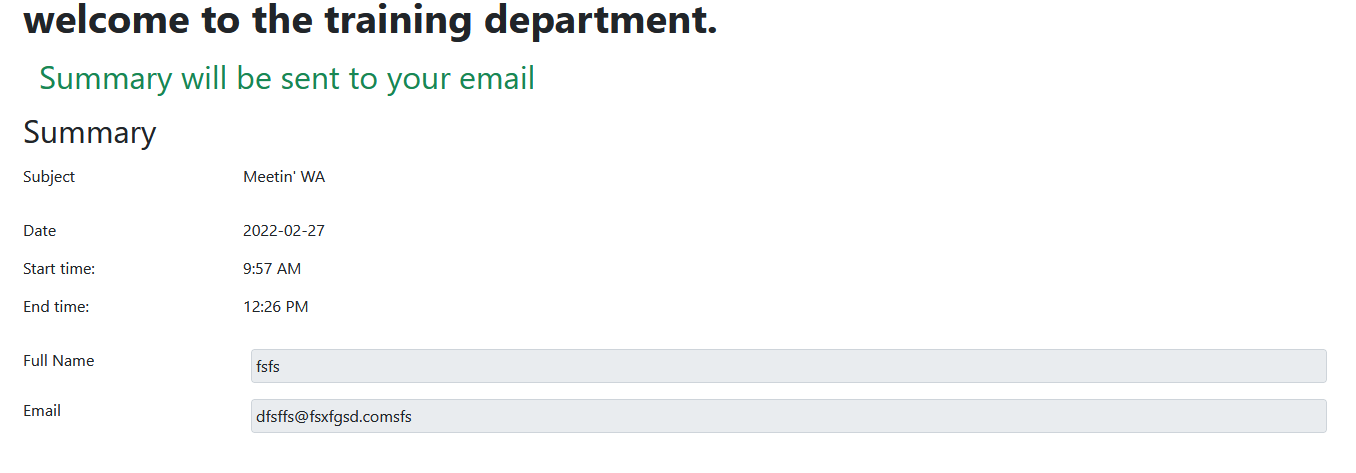
## Fill in form



## Confirmation popup



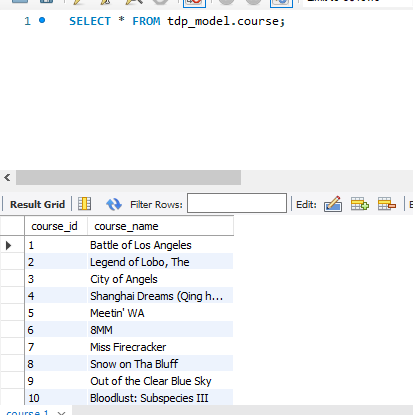
## Summary page



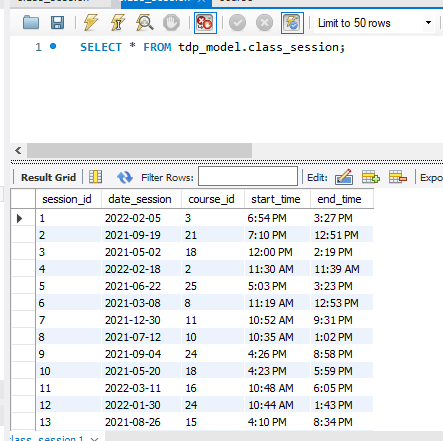
# SQL

Database name: tdp\_model

## Table course



## Table session



# Feedback on the system

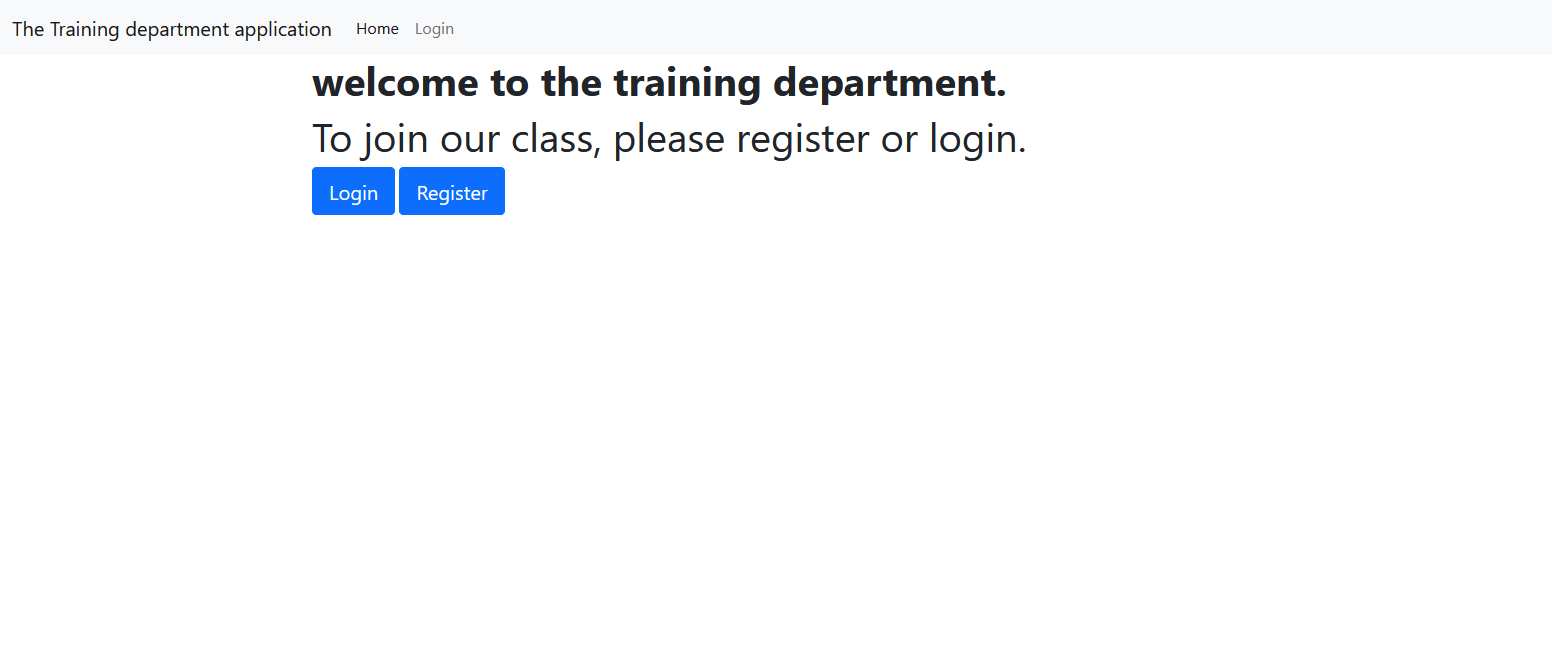
|  |  |
| --- | --- |
| System version | V1.0 Alpha |
| Feedback commenter: | Mentor Myint Haymar Lwin in online classroom meeting |
| Comment: | If time allows, please complete the system. Add login/signup, admin system, history check, create session. |

# Version 2.0 alpha

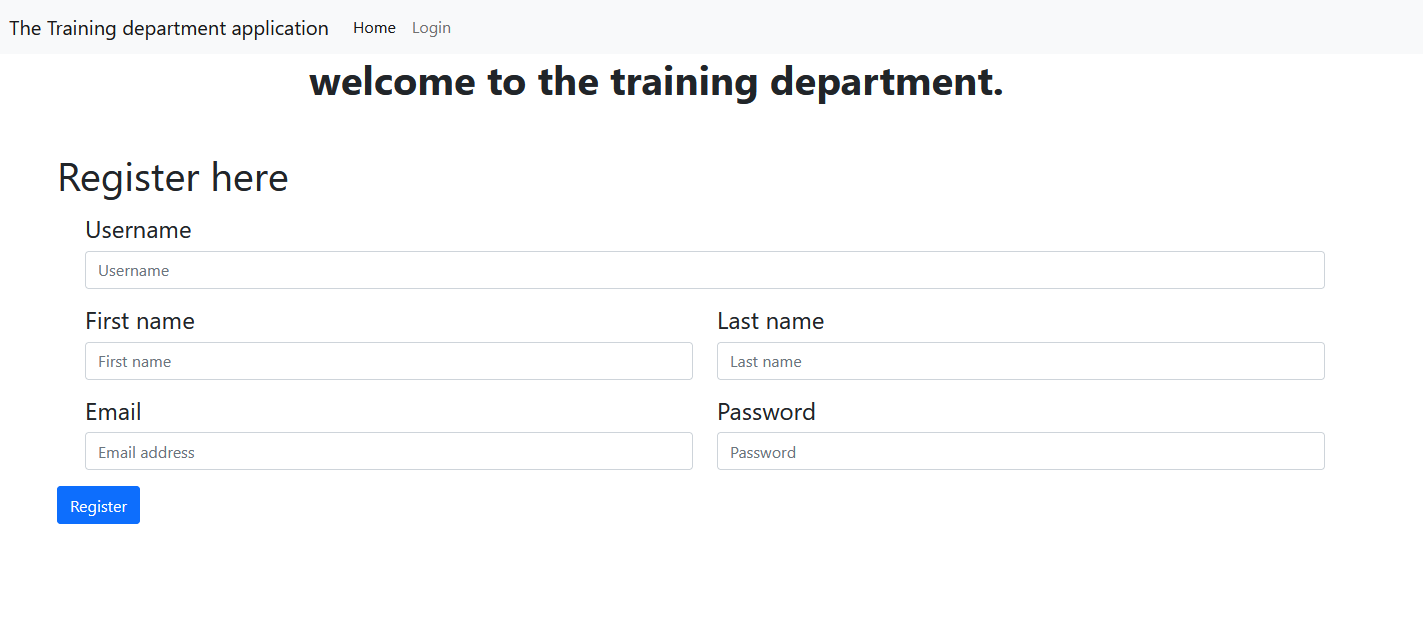
|  |  |
| --- | --- |
| System version | V2.0 alpha |
| Functions included | Login/signup, create session, history check.  Added teacher teaching session. |
| Functions not included | Admin system, history check, create session, add subject function. Account edit, |
| Reasons not include | No time to work on |

# Interface version 2.0 alpha

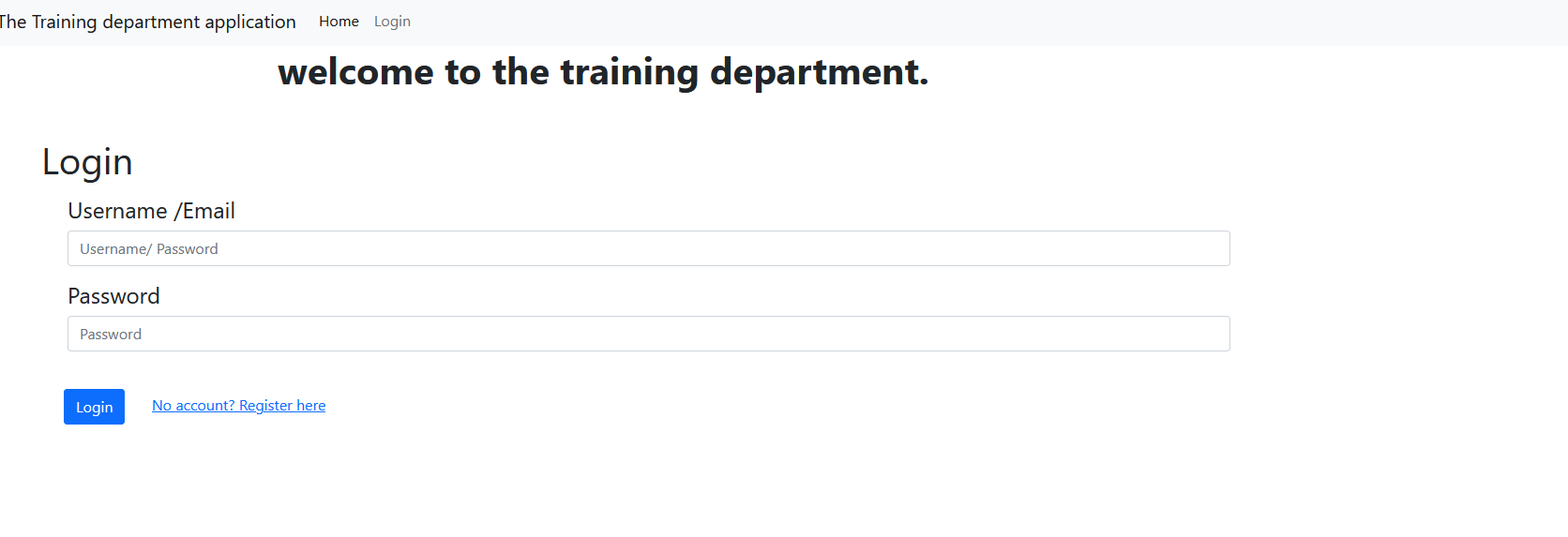
## Main page



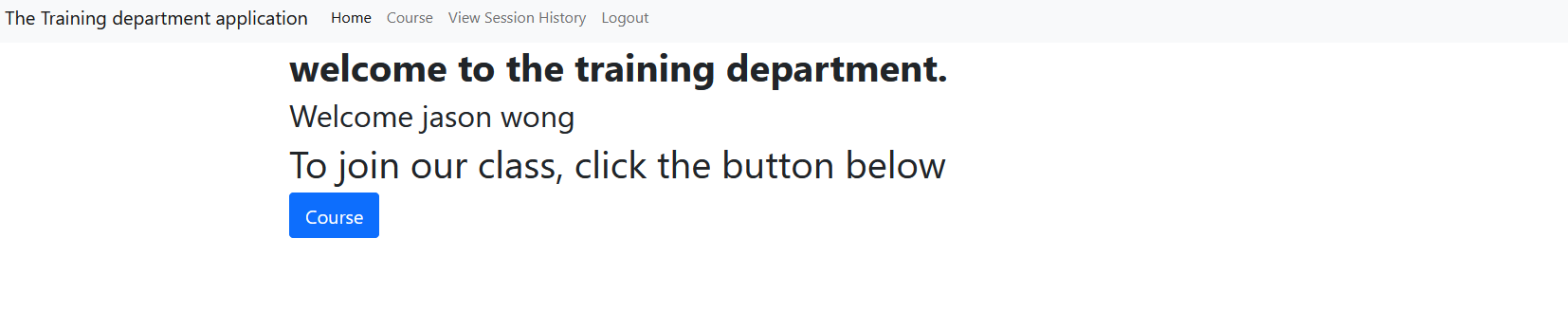
## Register page



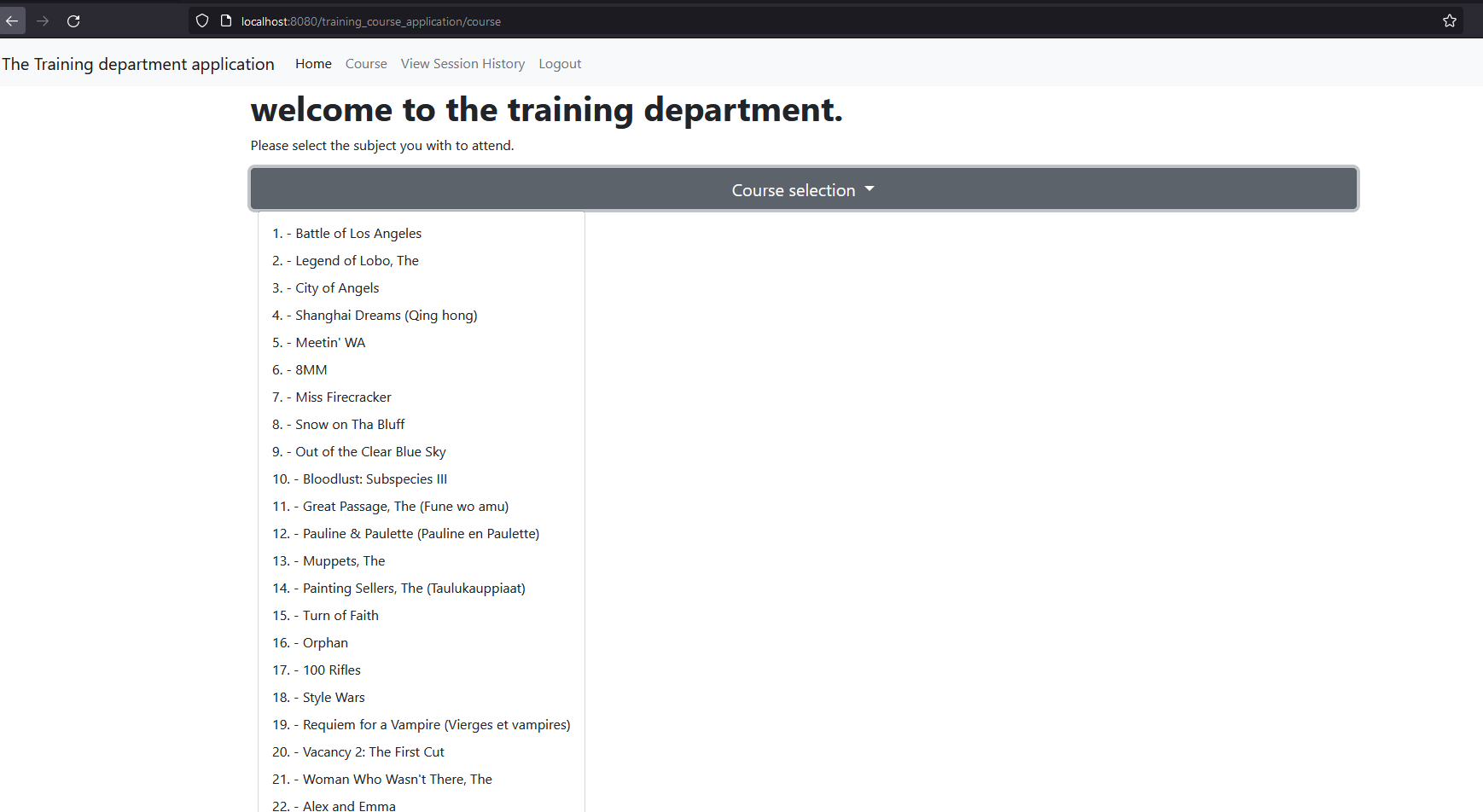
## Login page



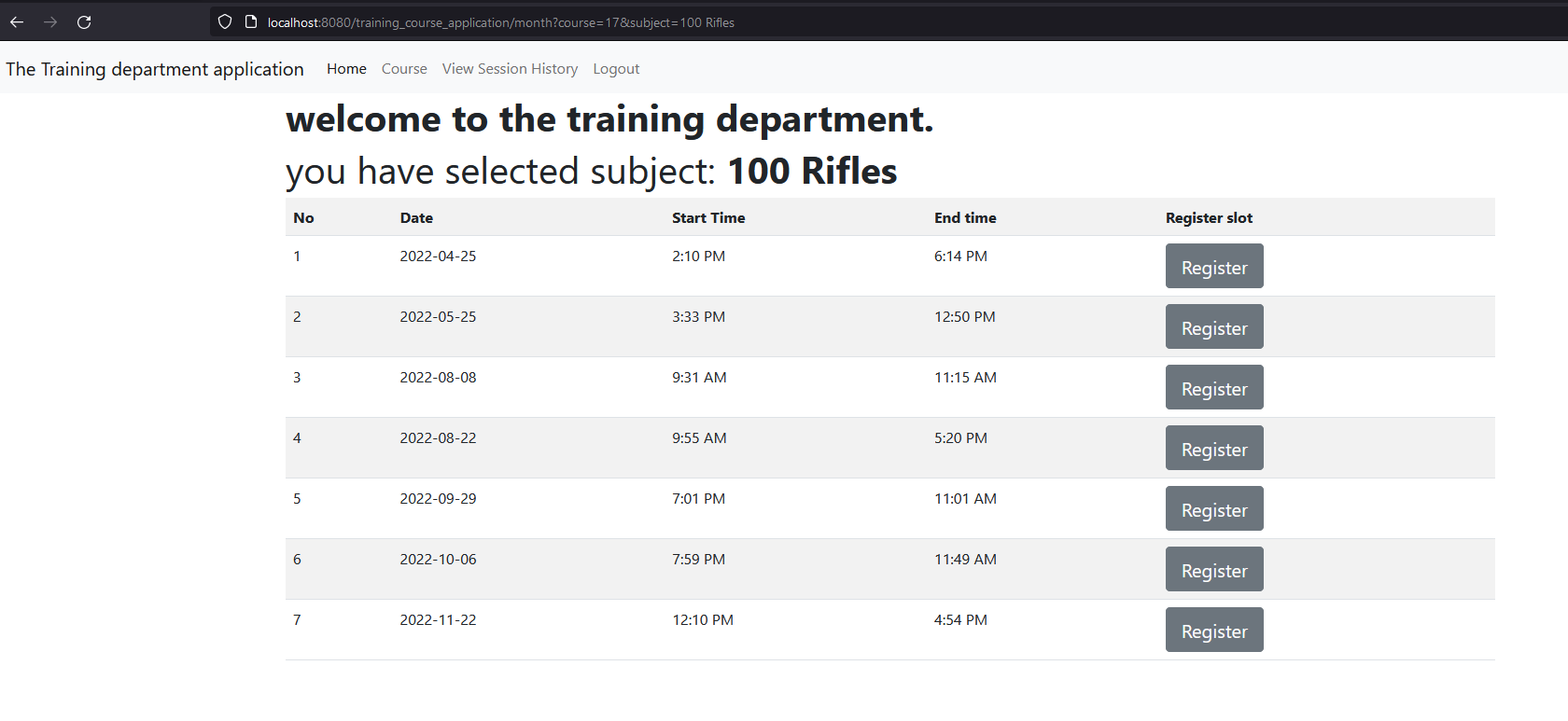
## Logged in front page



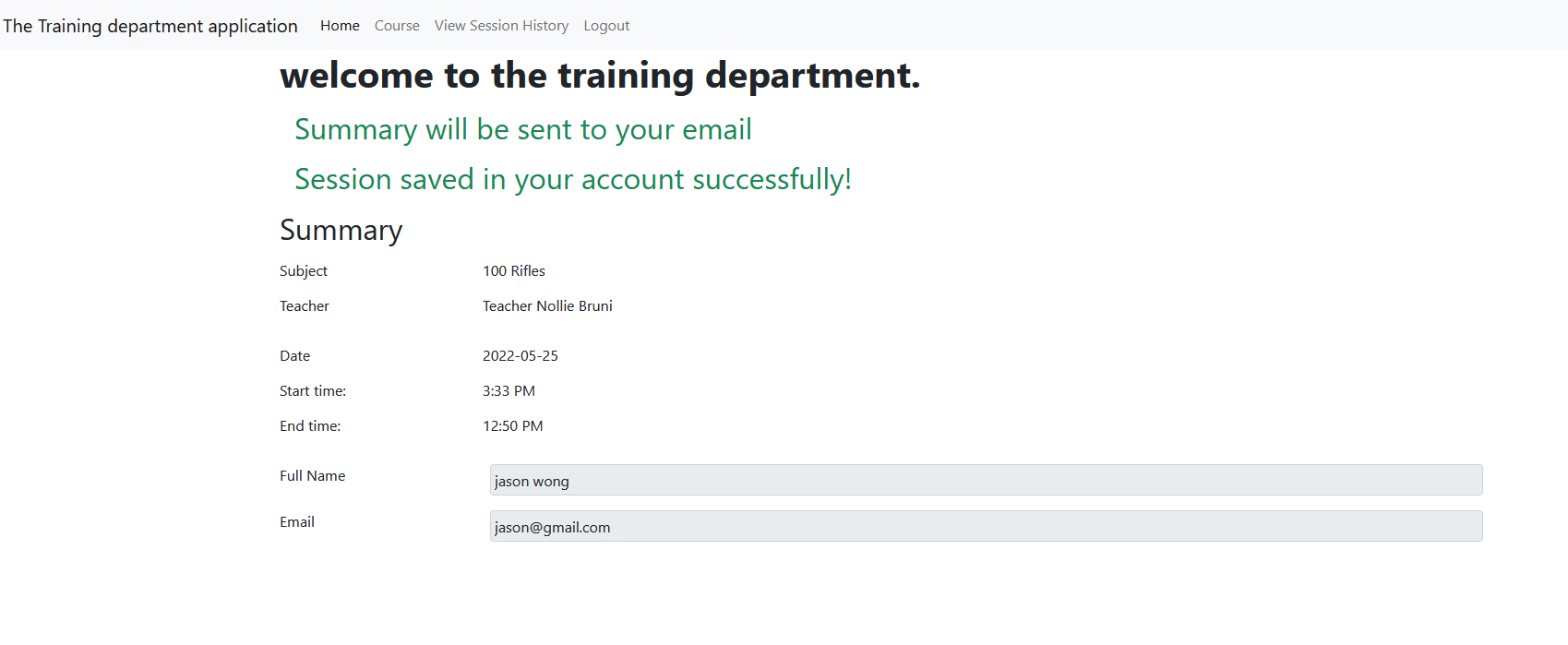
## Course page



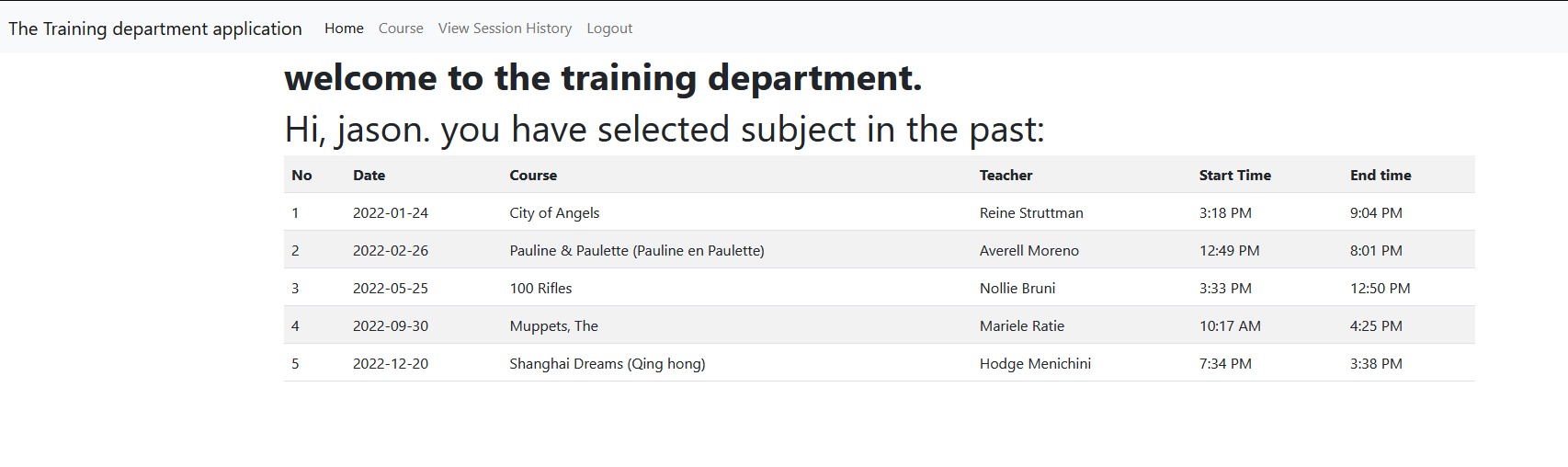
## Time session for subject



## Summary page

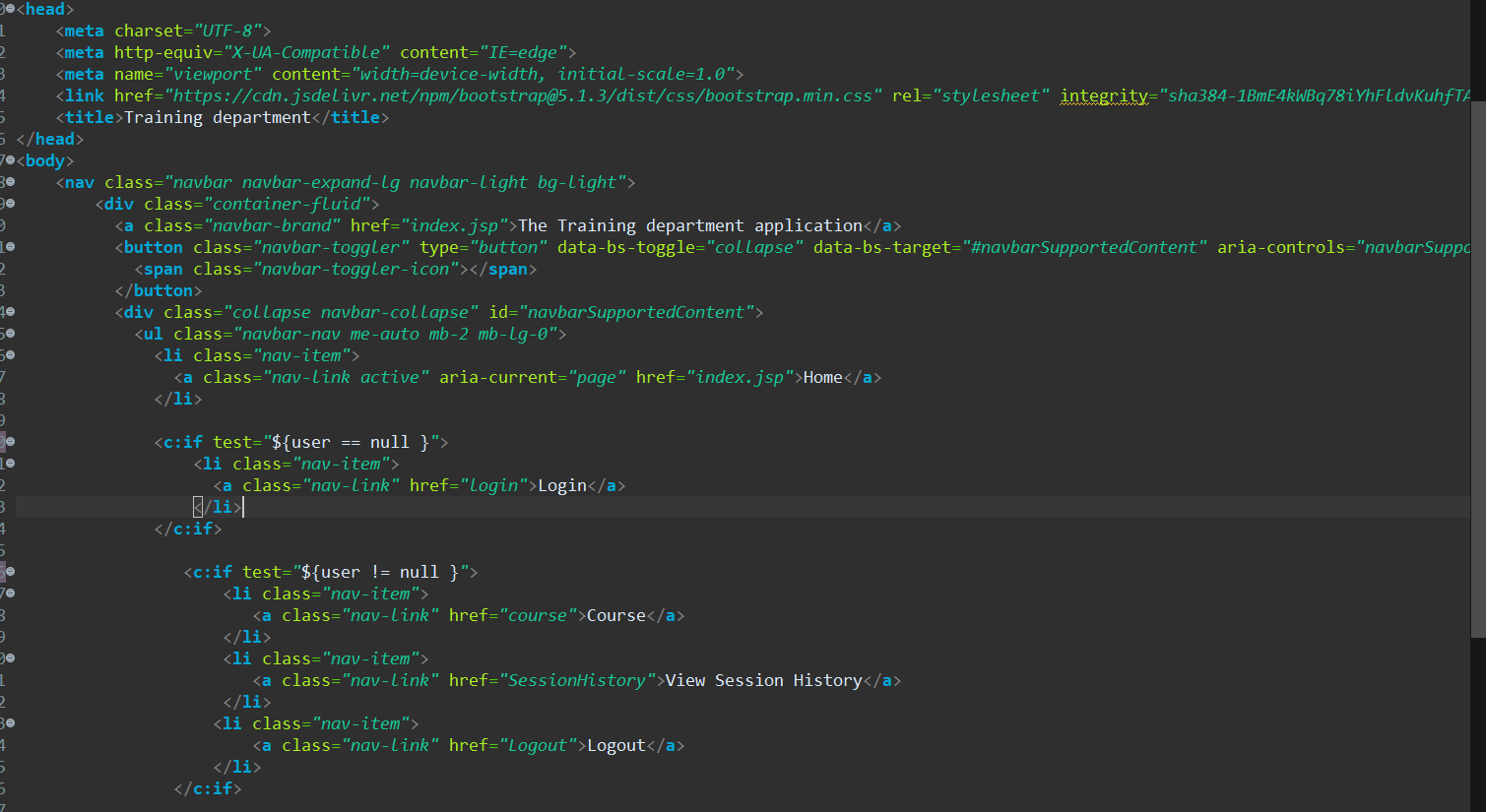


## Session history check

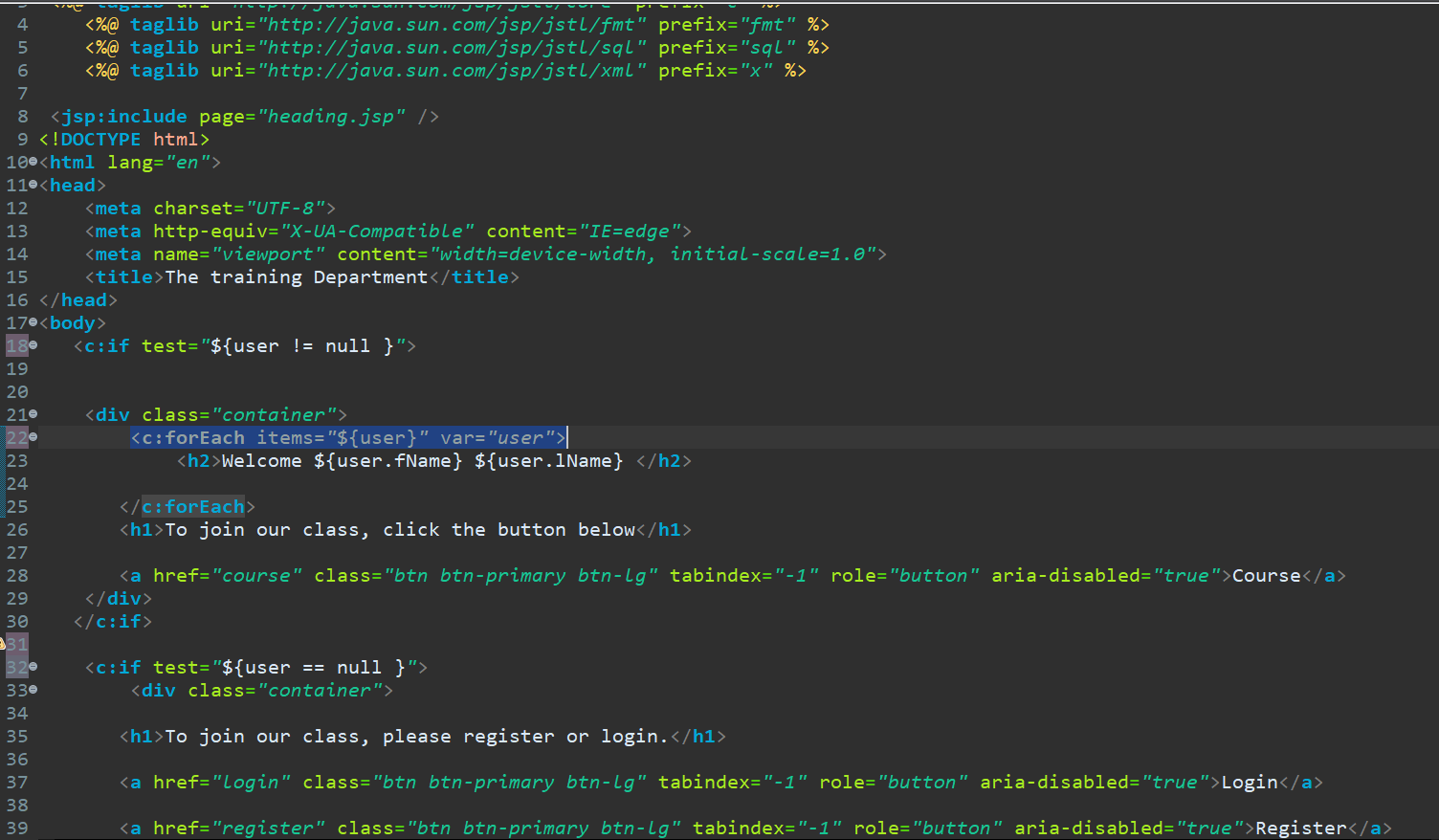


# Page code

## Heading.jsp

­

## Index.jsp



## Login.jsp



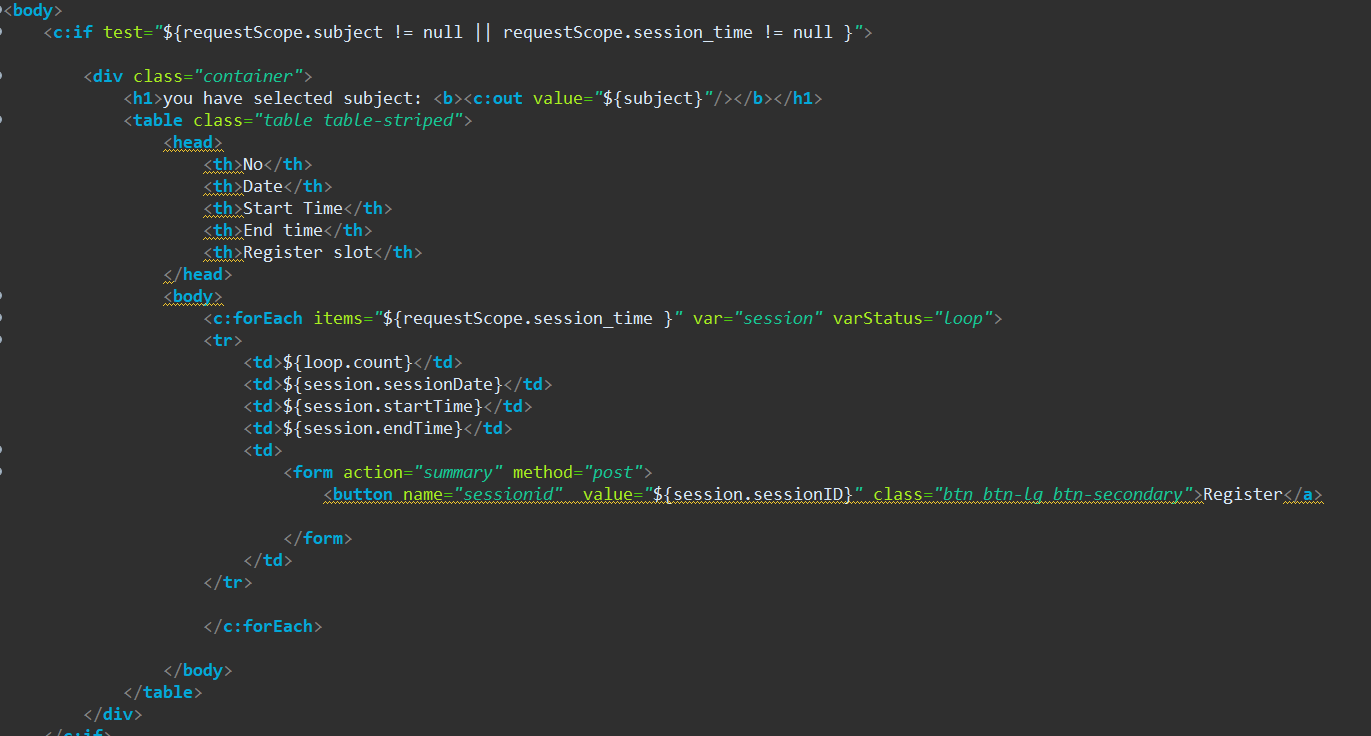
## Register.jsp



## Course.jsp



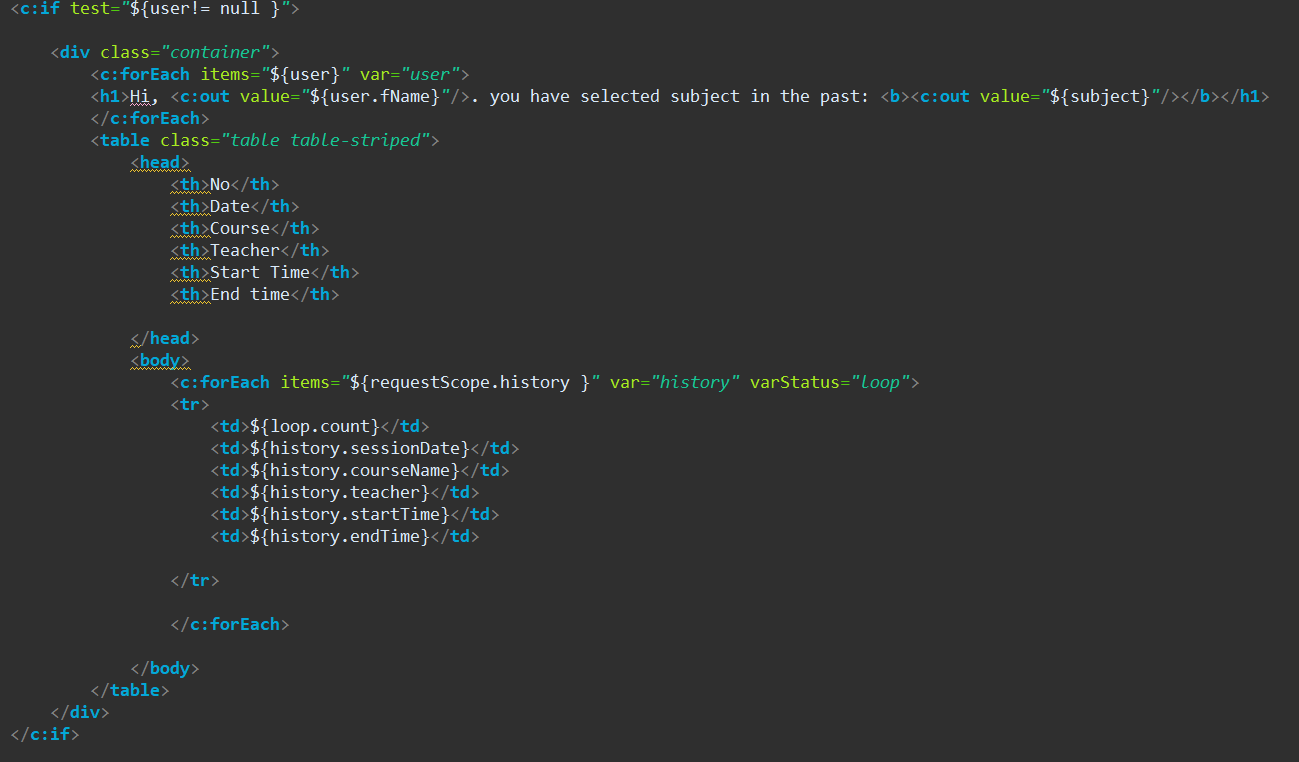
## Timesession.jsp



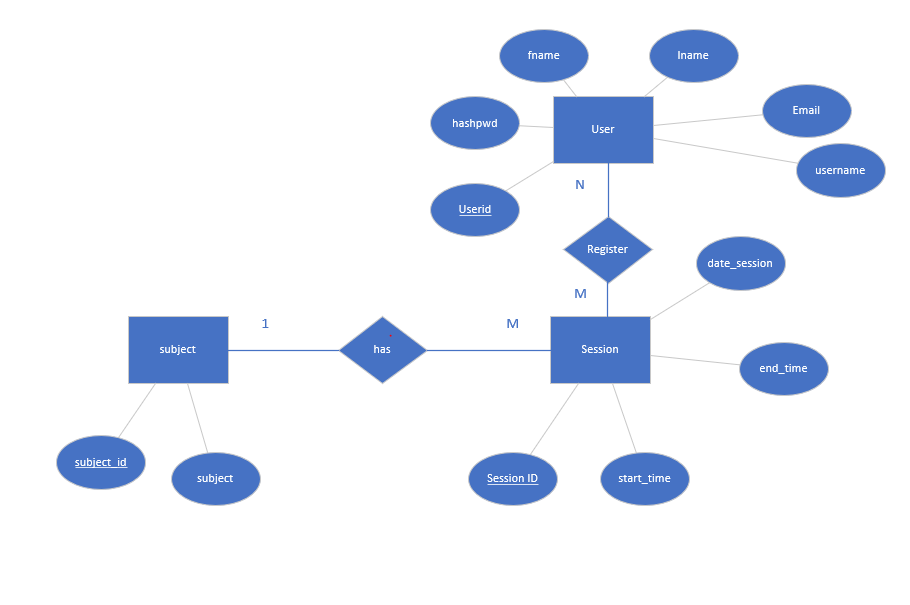
## Summary.jsp



## Sessionhistory.jsp



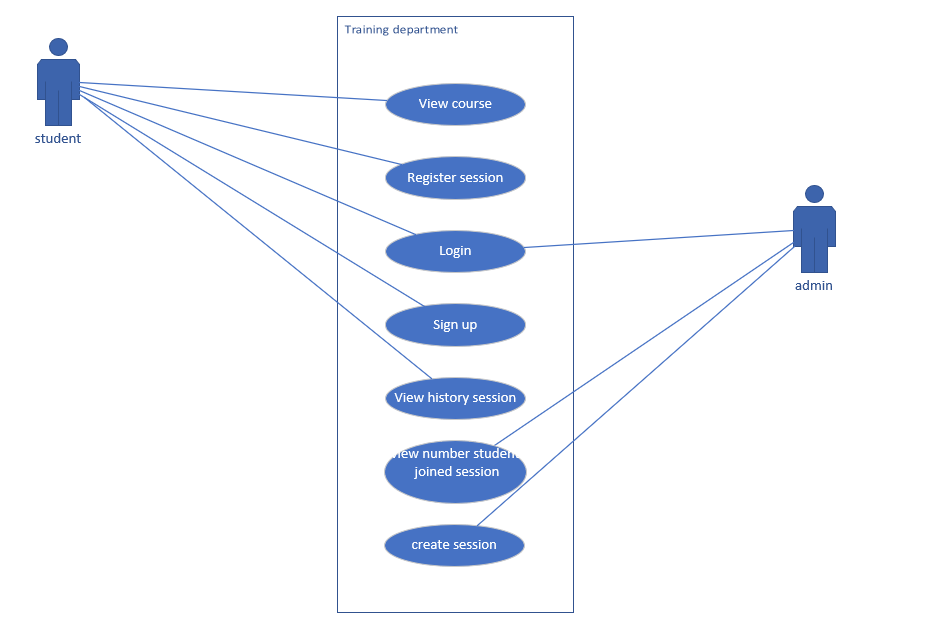
# ERD for version 2.0 alpha



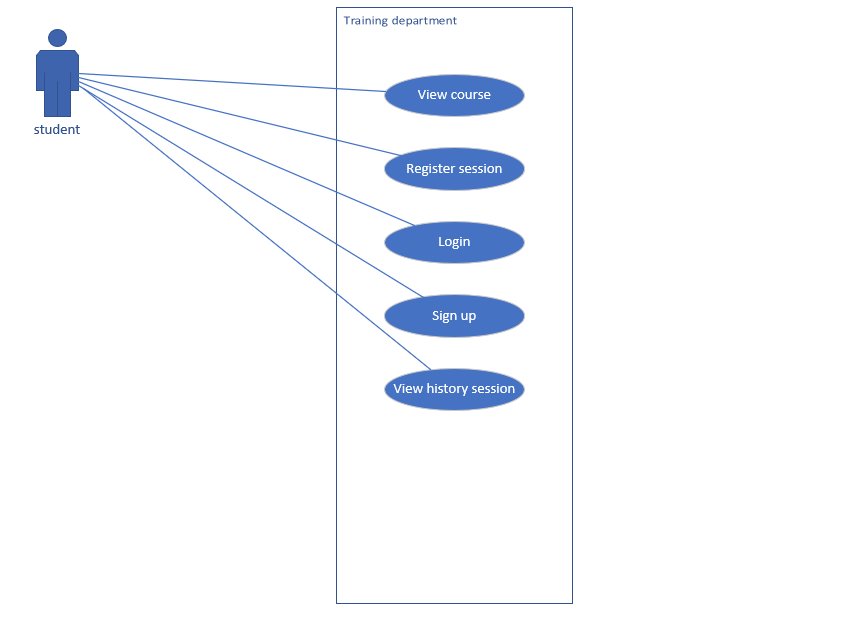
1 subject can have many sessions. One session has 1 subject only. User can register many sessions, many sessions can be registered by many users.

# UML use case

## Initial design for v2.0a



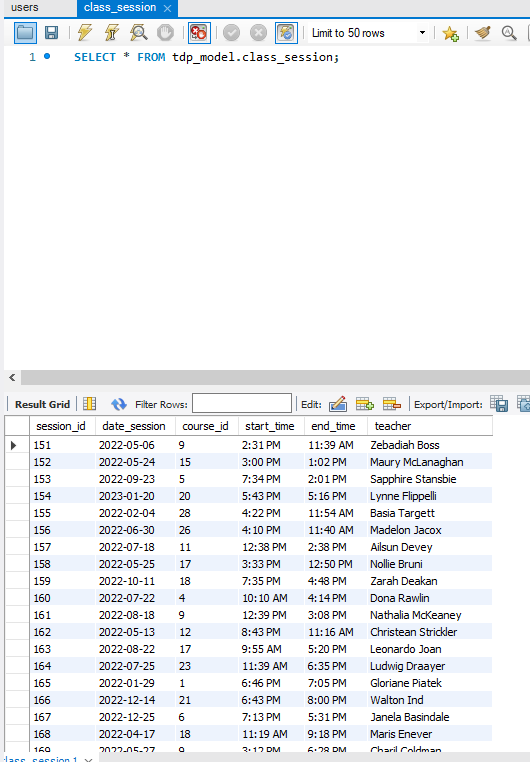
## Real system use case



# SQL

Database name: tdp\_model

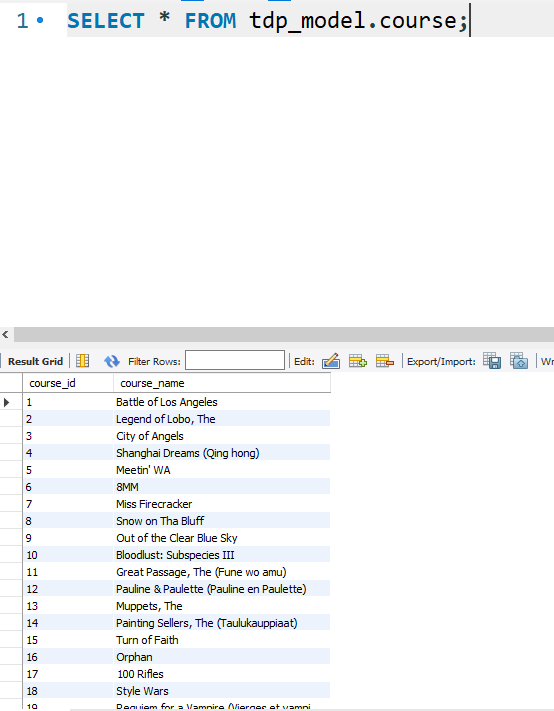
## Class\_session table



## Users table



## Course table

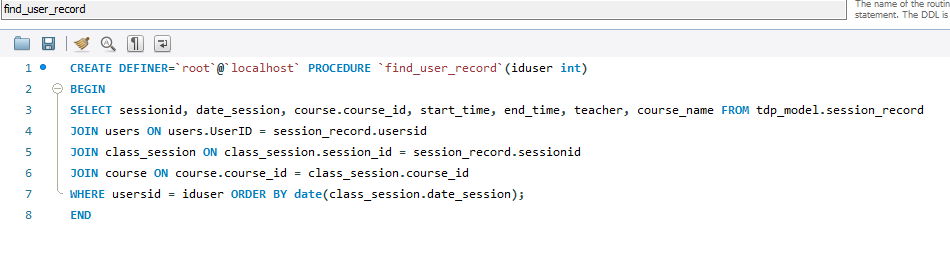


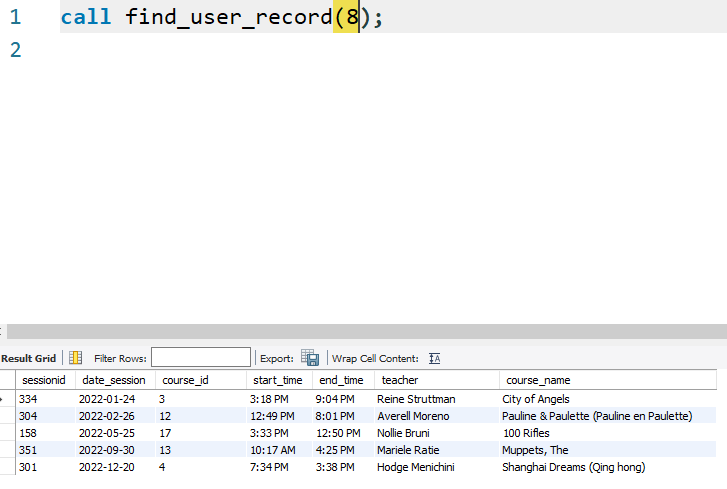
## Session\_record table



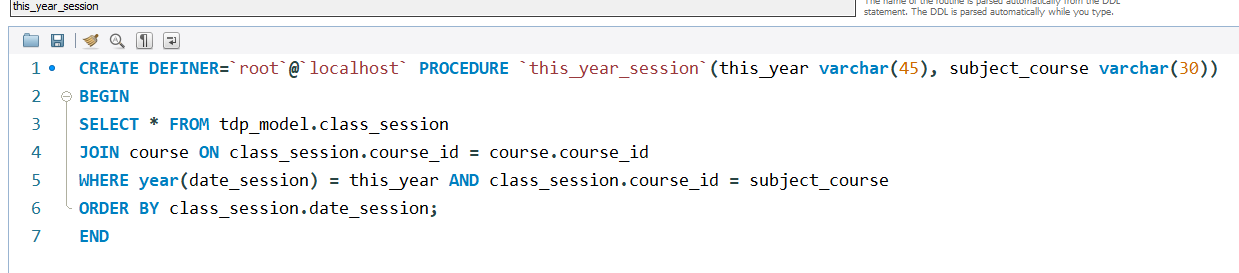
## SQL Procedures

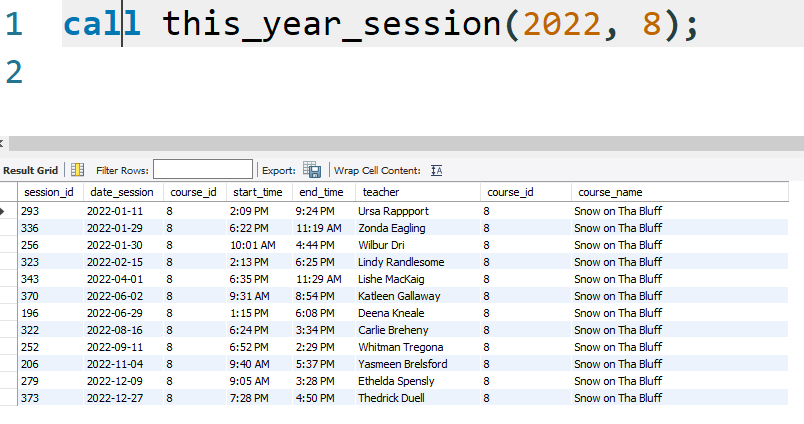
### Find\_user\_record





### This\_year\_session procedure



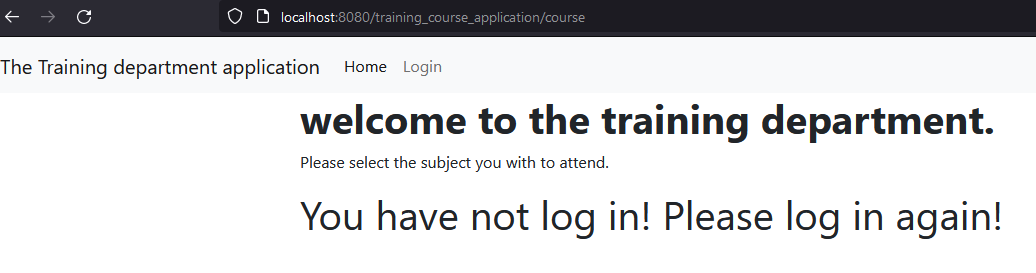


# Test plan

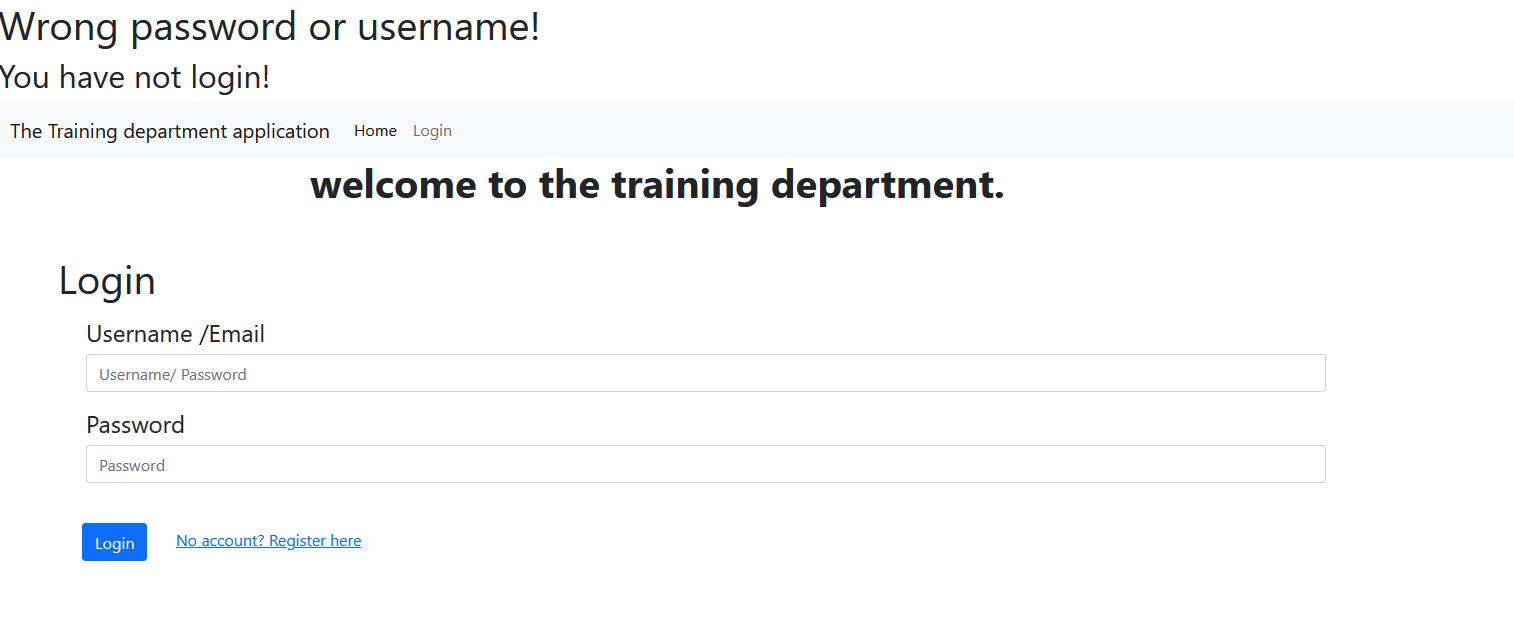
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Test area | Test expectation | Test result | Status |
| 1 | Interface testing | When user skip login and went to course such as,  System will lock user from accessing it | Refer to Image 101 | success |
| 2 | Acceptance testing | If the user fill in the wrong information when login, system will reject | Refer to Image 102 | success |
|  |  |  |  |  |

## Image evidence

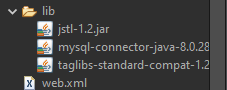
### Image 101



### Image 102



# Library integrated in the system

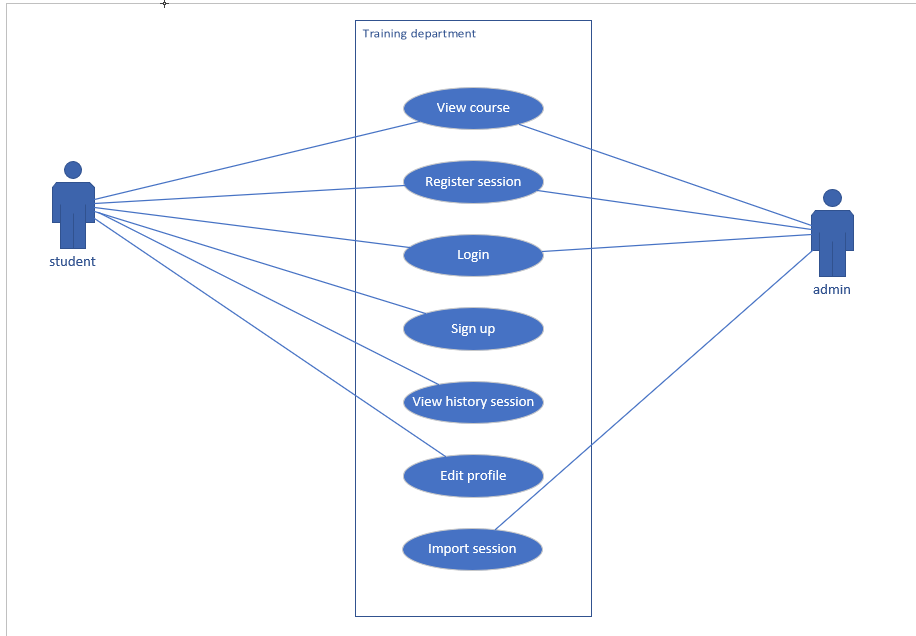


Jstl 1.2.jar

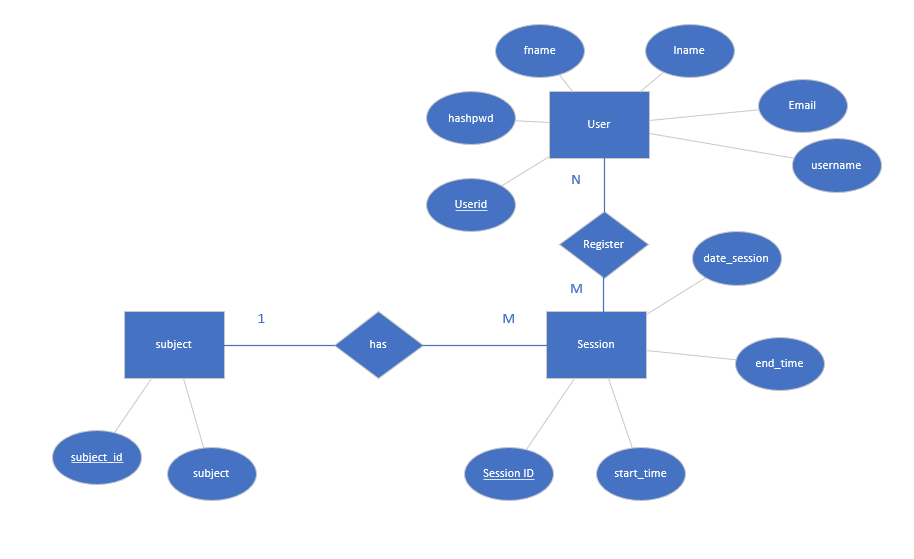
Mysql. JDBC- 8.0.28

# Update version 3.0 alpha version

# UML Use Case version 3.0

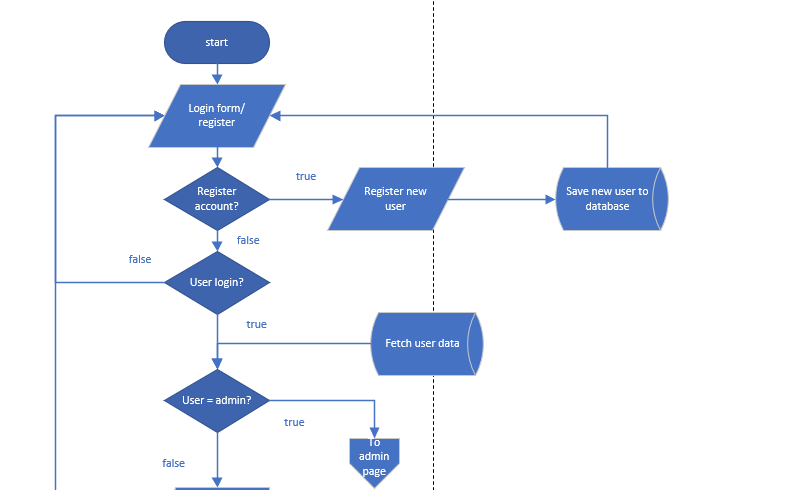


# ERD version 3.0



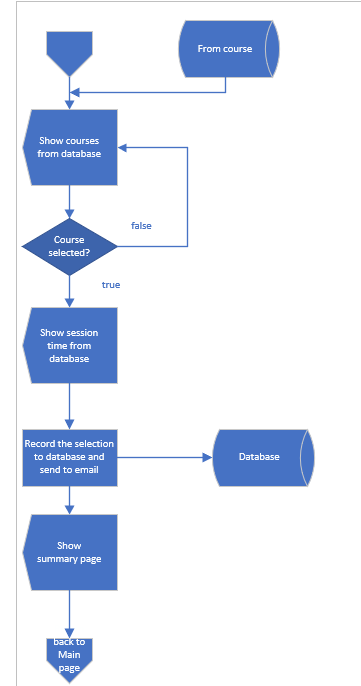
# Flowchart

## User flowchart

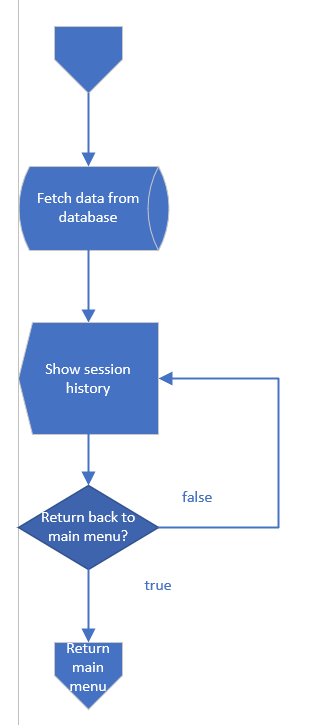




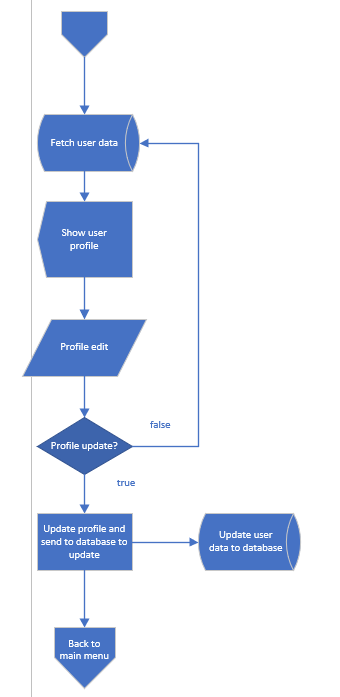
## Register course



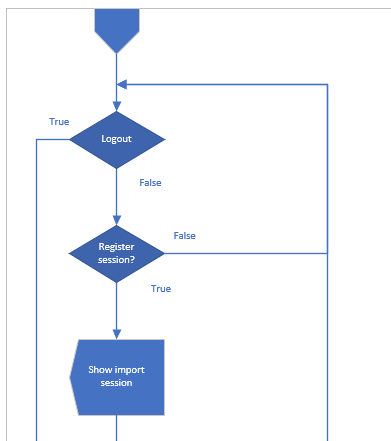
## Check session history flowchart

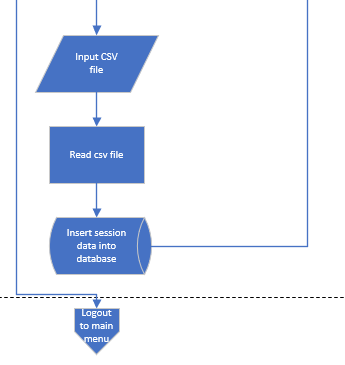


## Profile page



## Admin page

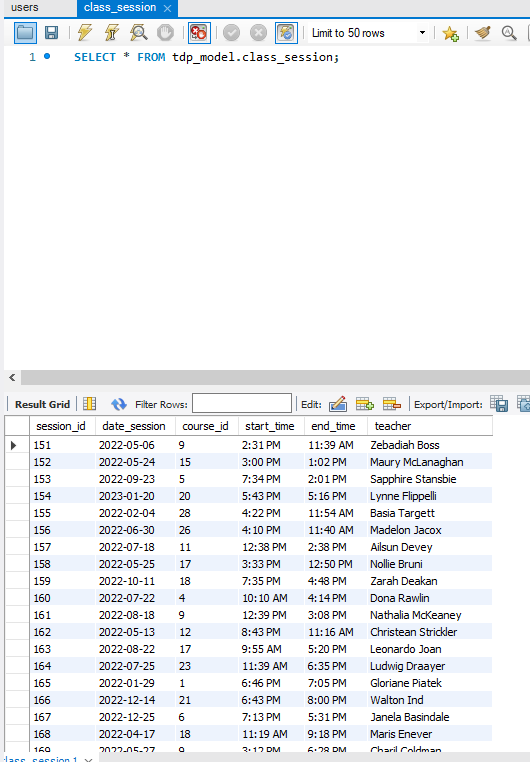




# SQL version 3.0

Database name: tdp\_model

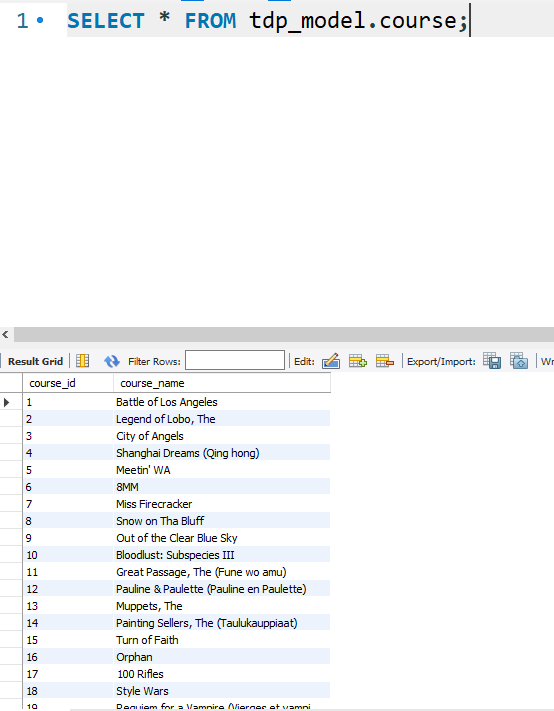
## Class\_session table



## Users table



## Course table

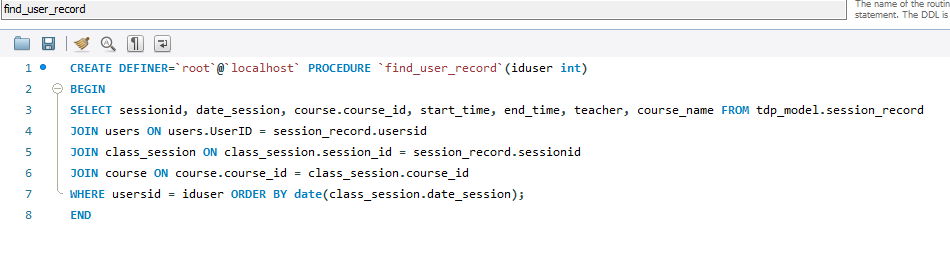


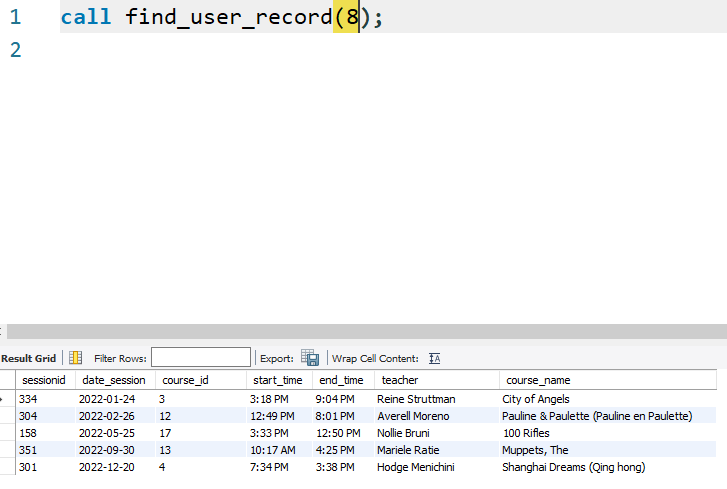
## Session\_record table



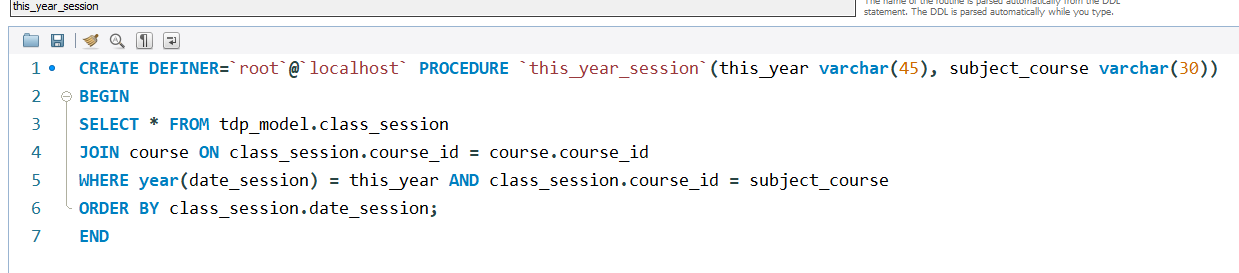
## SQL Procedures

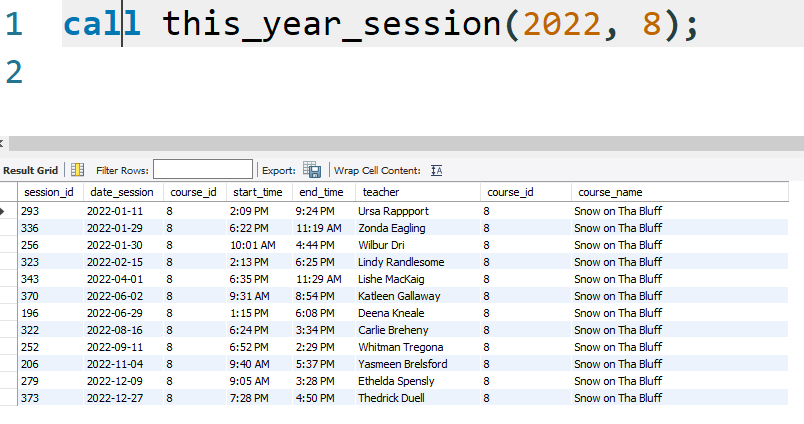
### Find\_user\_record



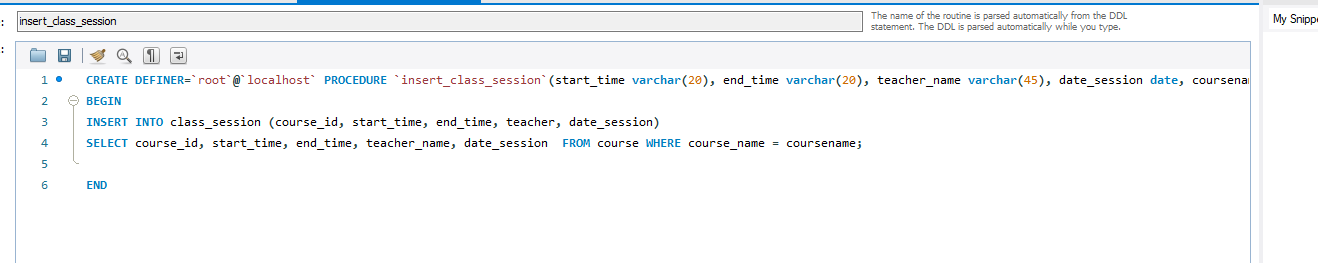


### This\_year\_session procedure

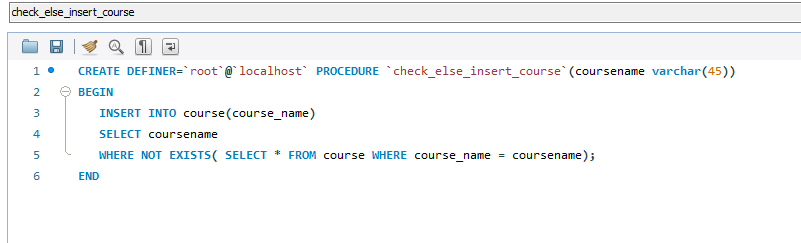




### Insert\_class\_session

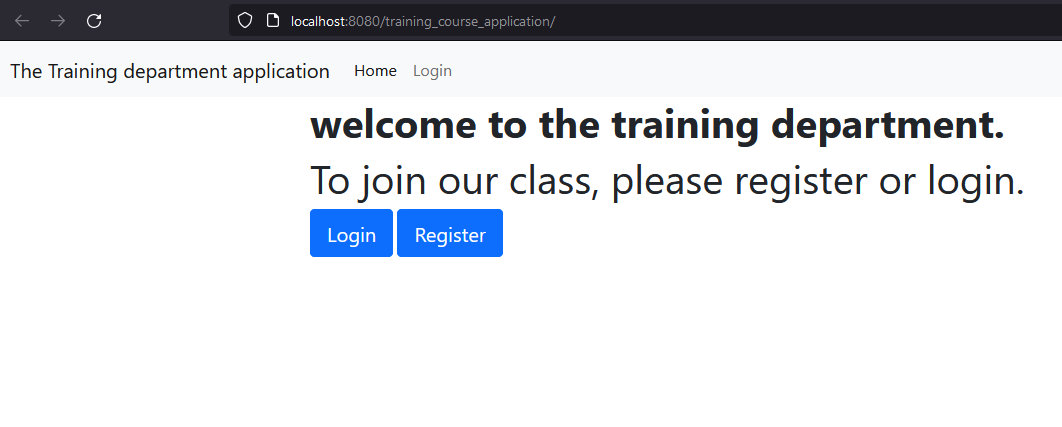


### Check\_else\_insert\_course

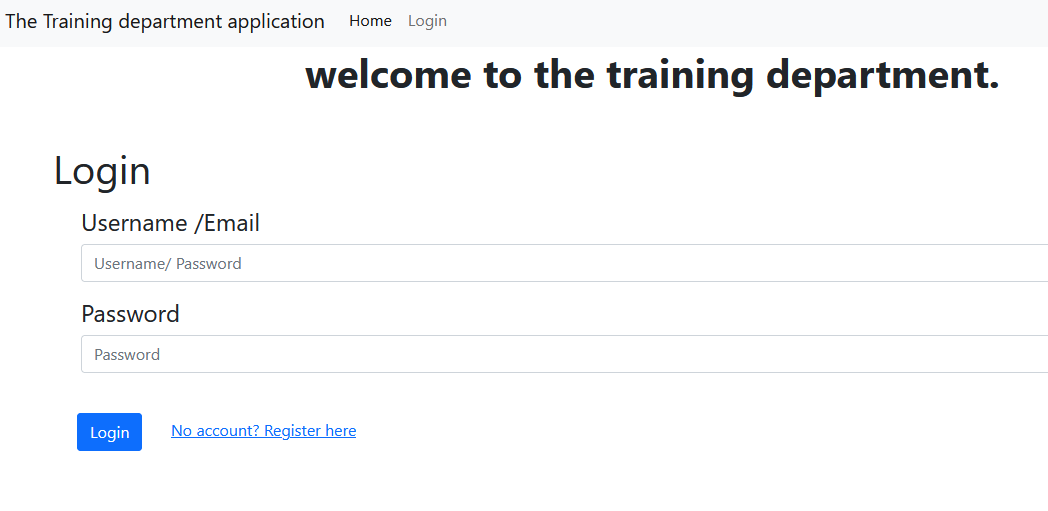


# Webpage interface version 3.0

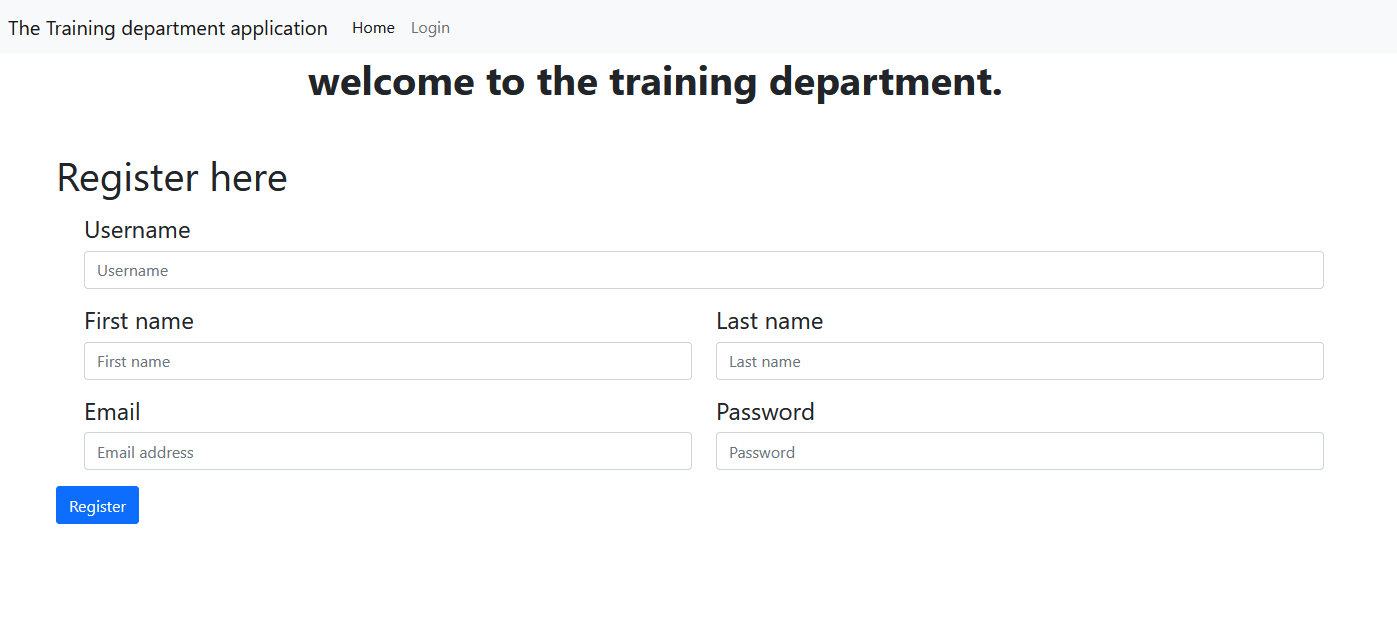
## Main menu



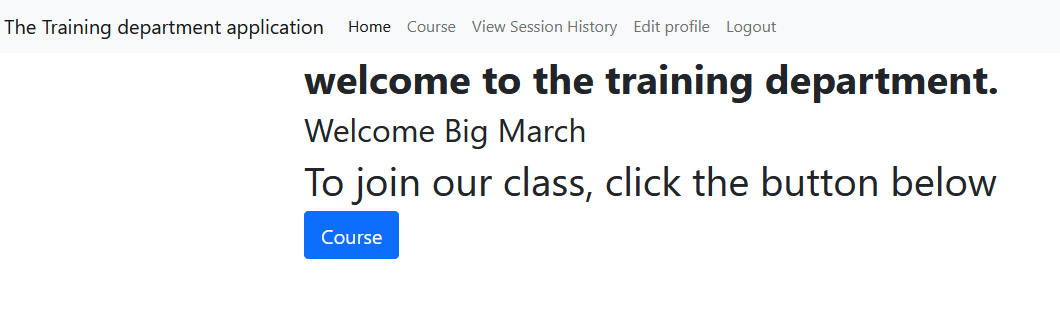
## Login



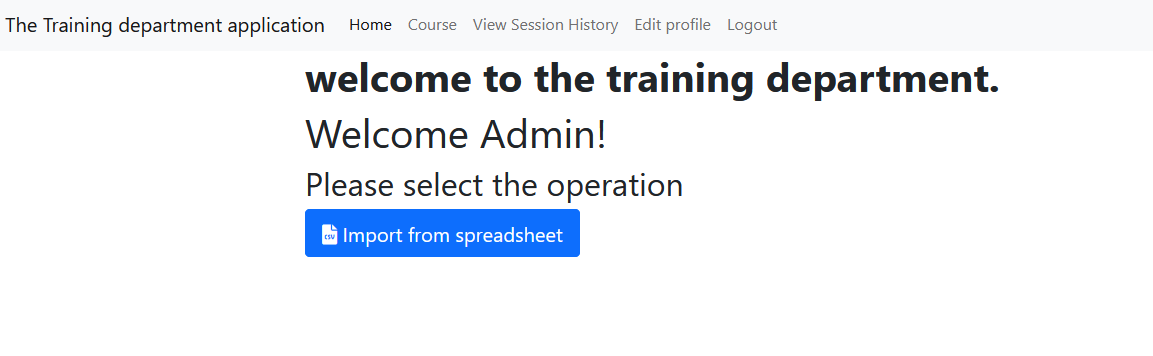
## Register



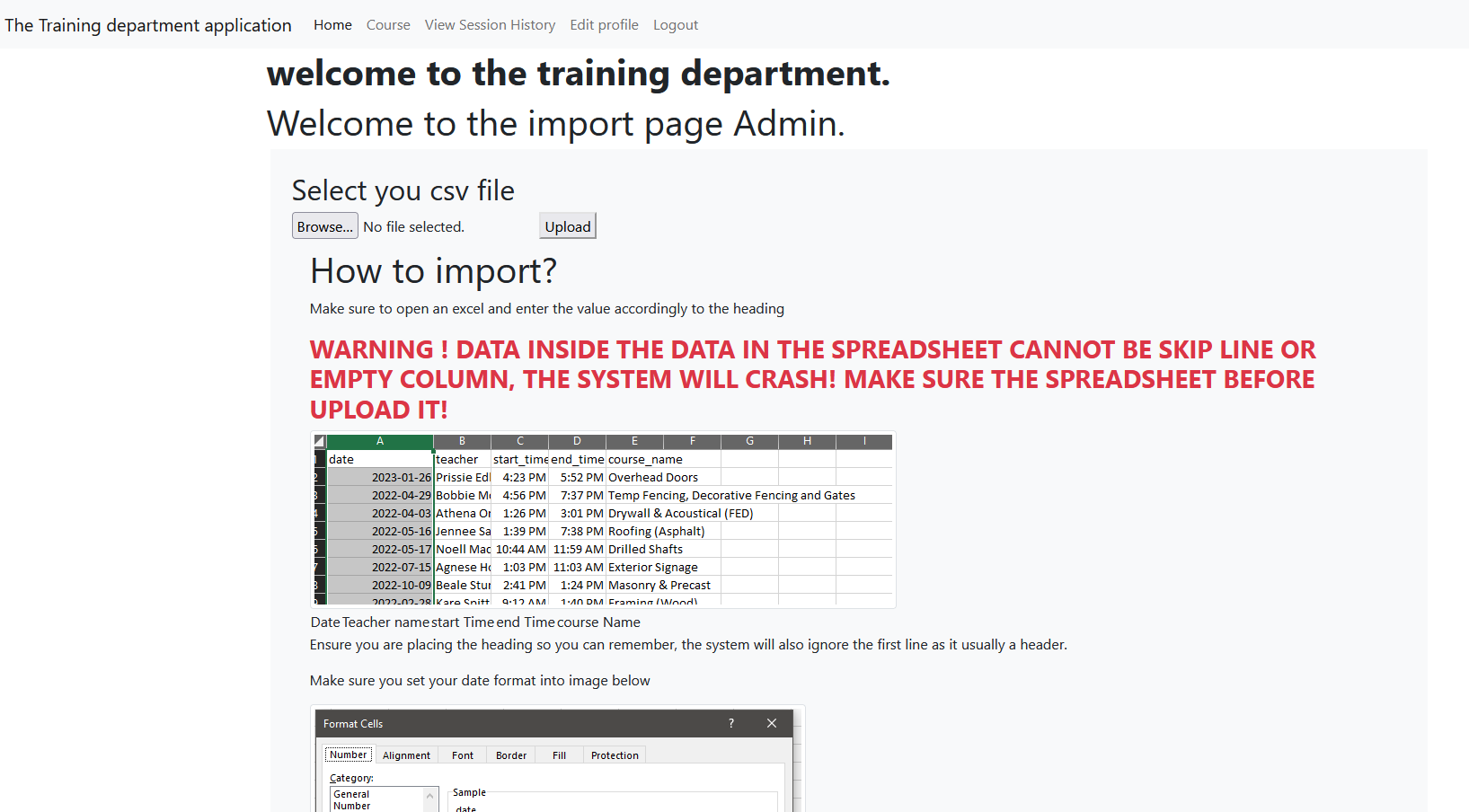
## Main page for user



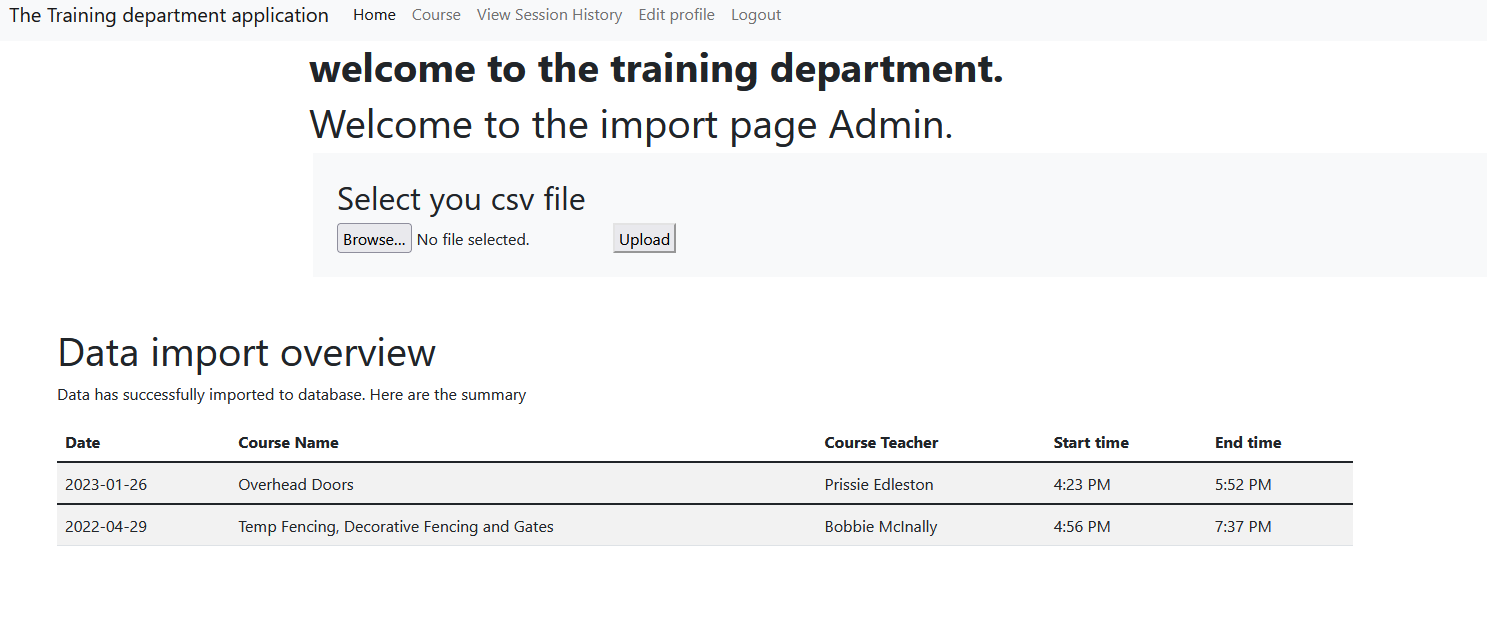
## Main page for admin



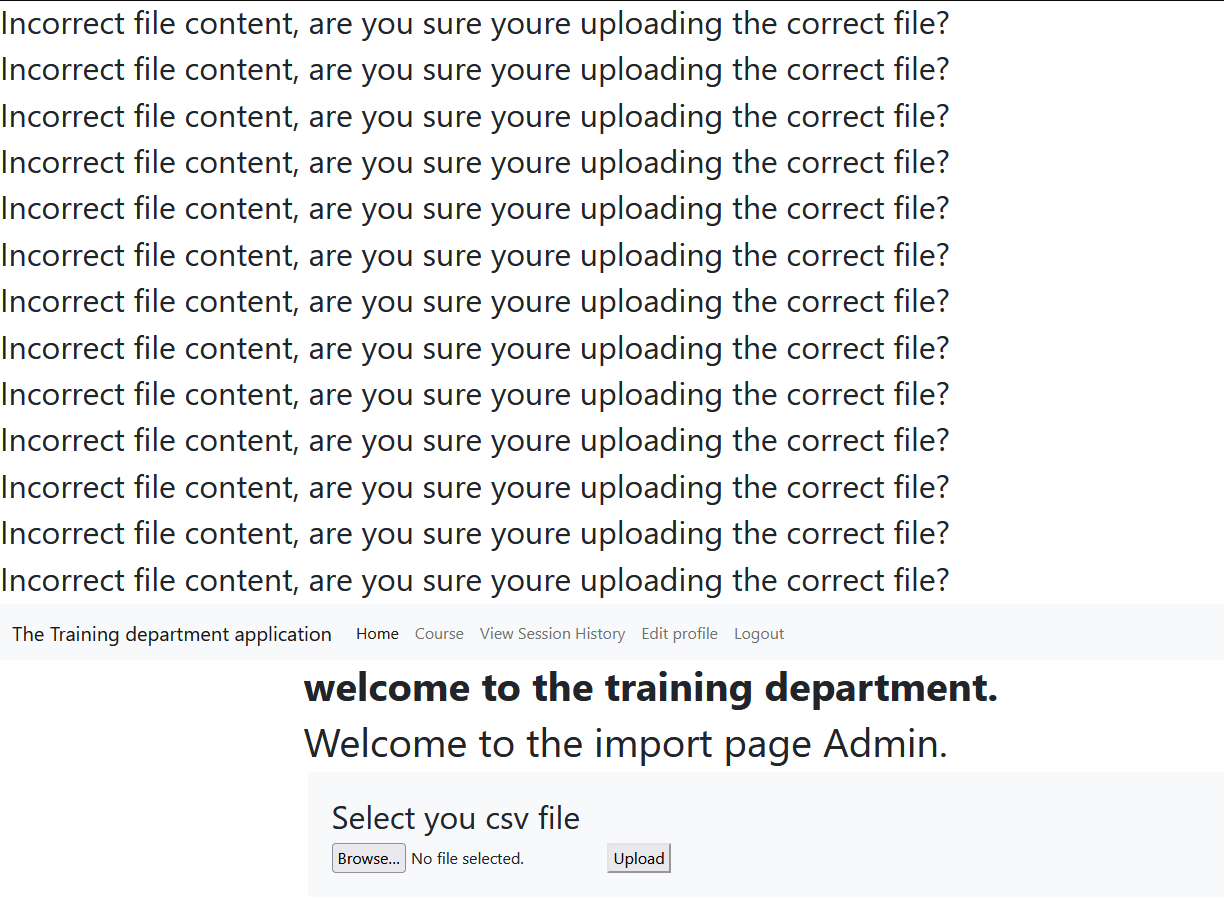
## Admin page import spreadsheet



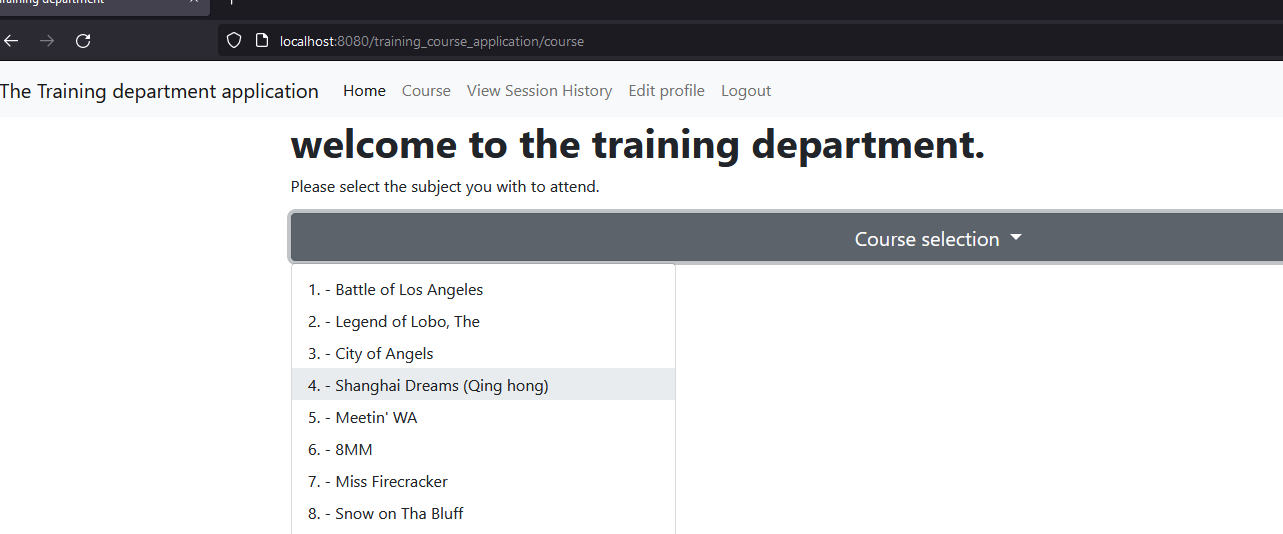
## Import successful page after spreadsheet is uploaded



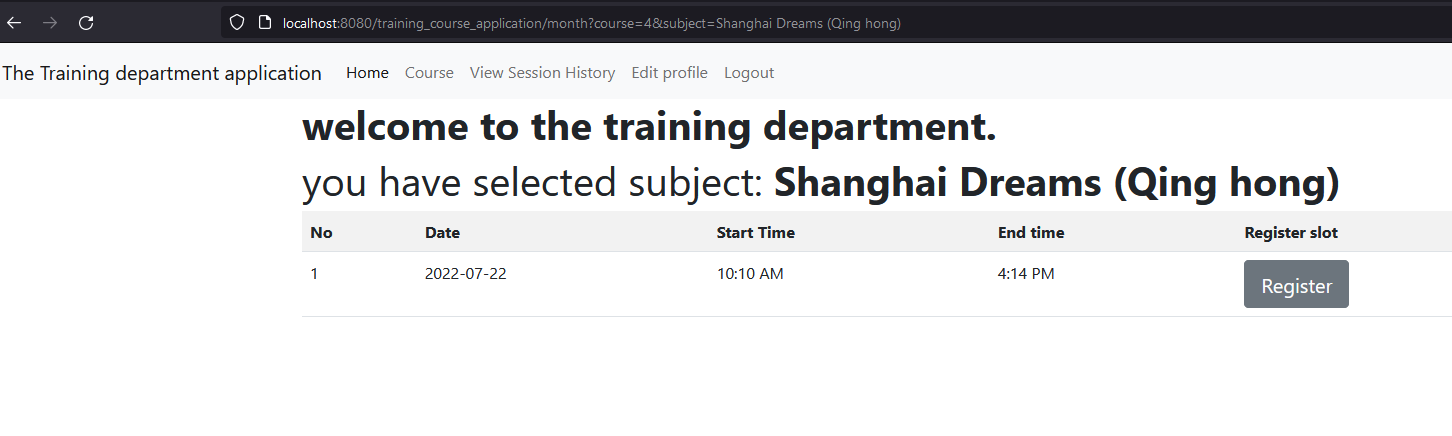
## Wrong file content



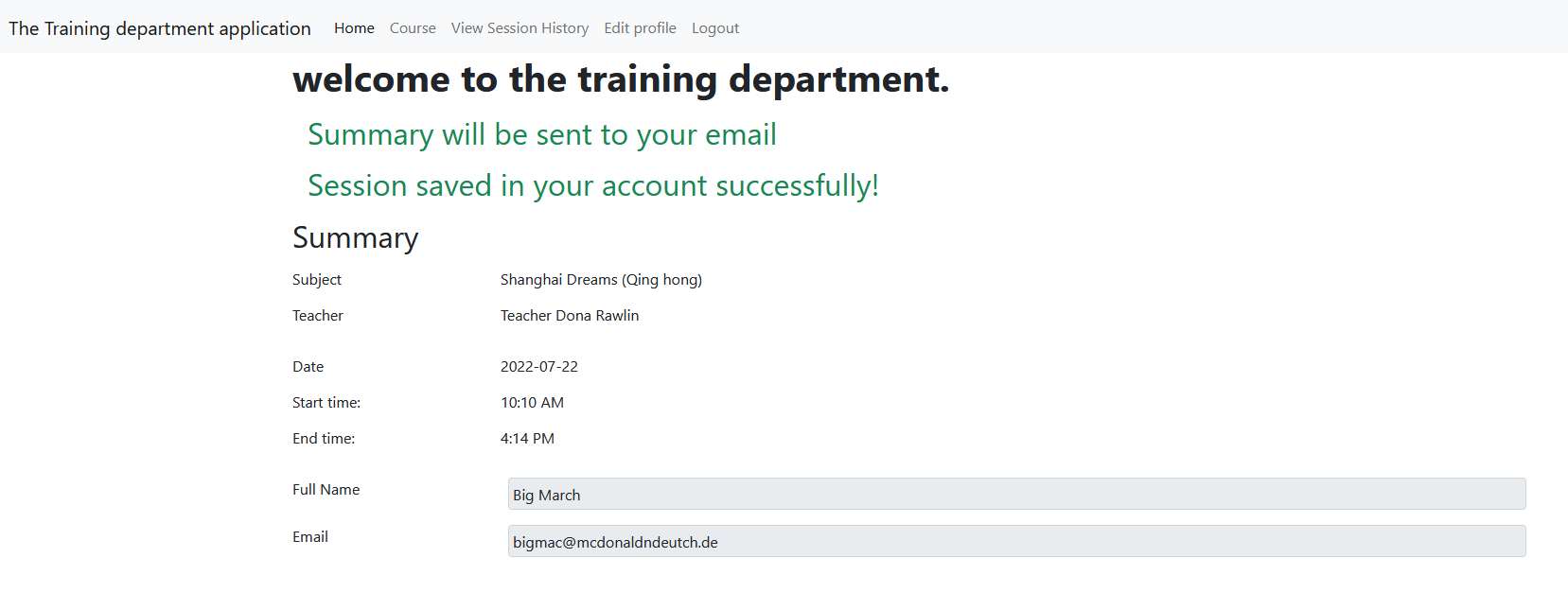
## Course selection page



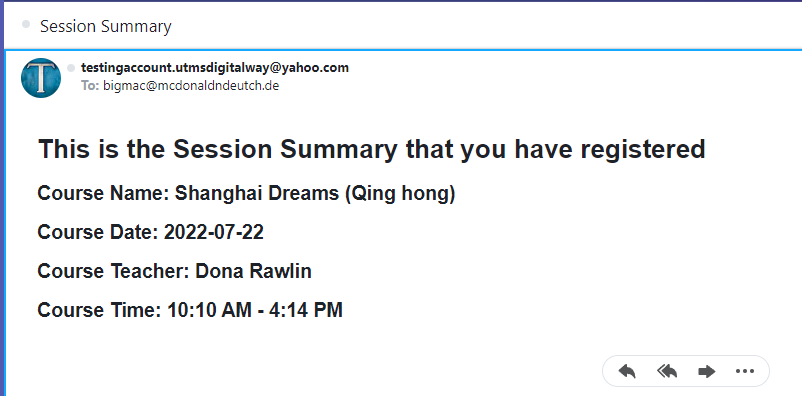
## Select time session page



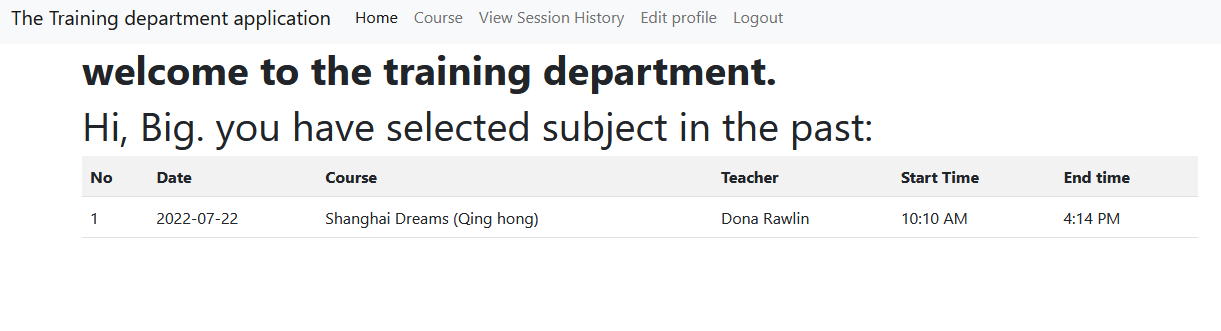
## Summary page



## Email page



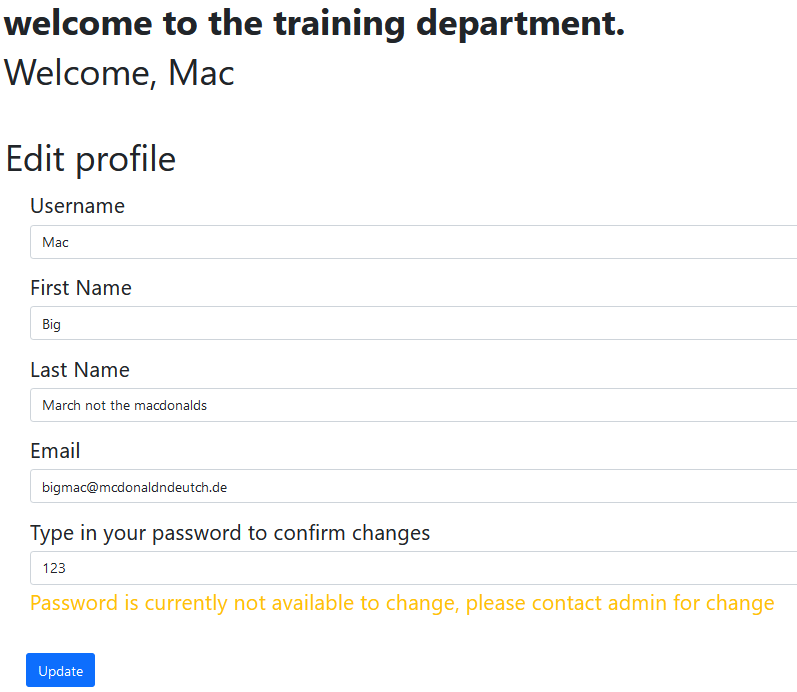
## Session history page



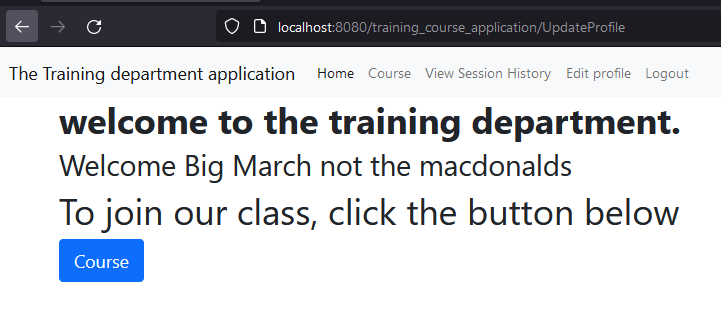
## Edit profile page



## Change profile

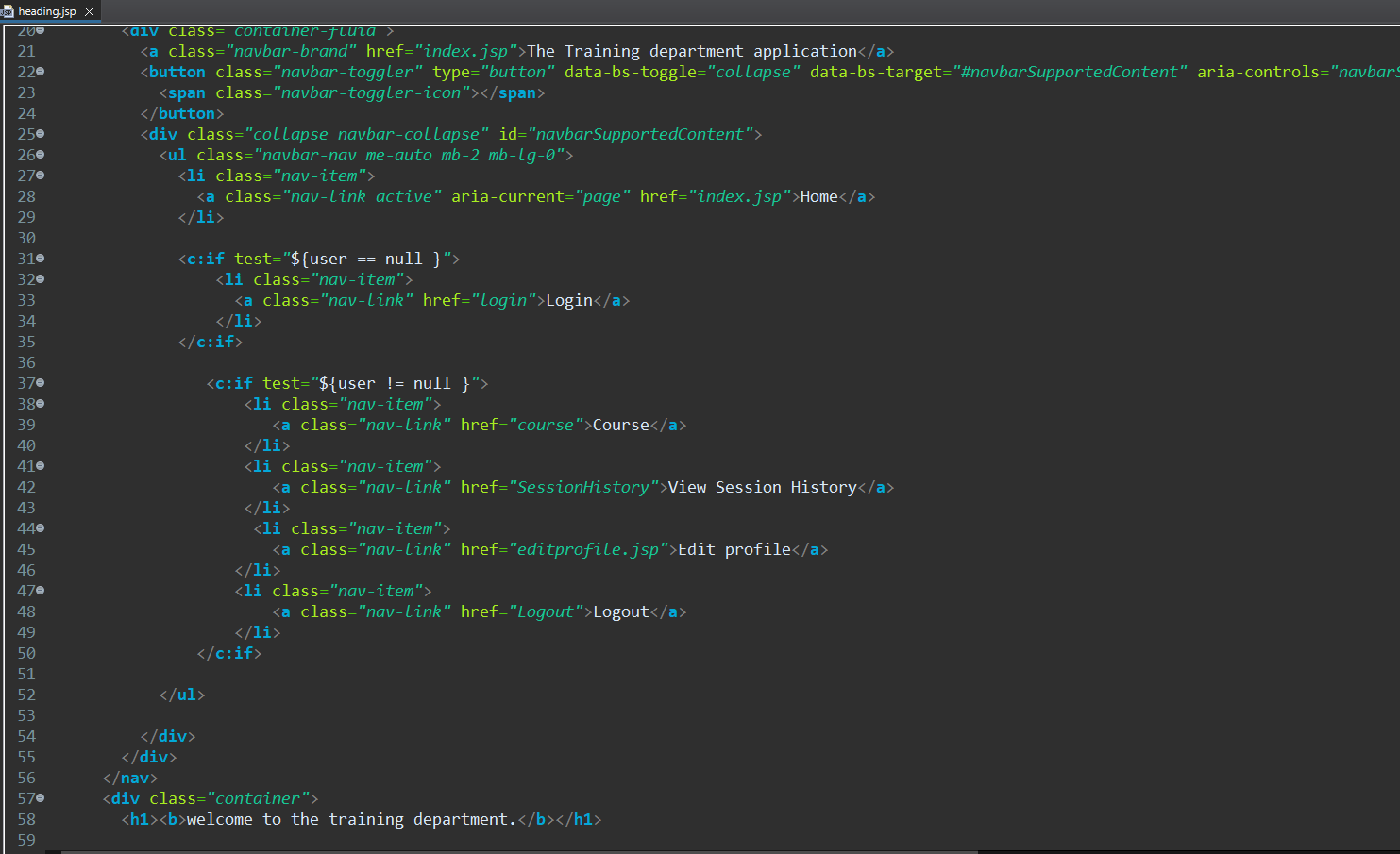


Profile after successfully update

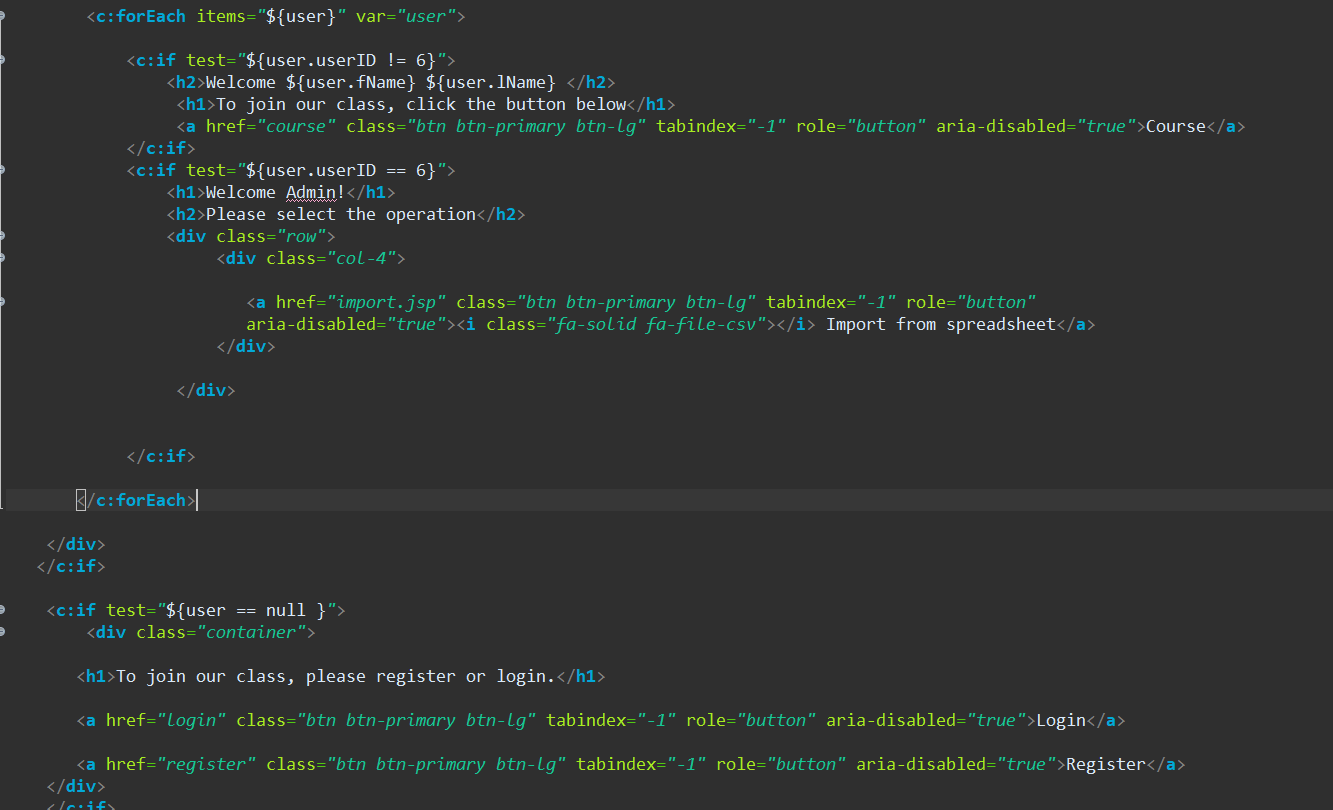


# Source code 3.0

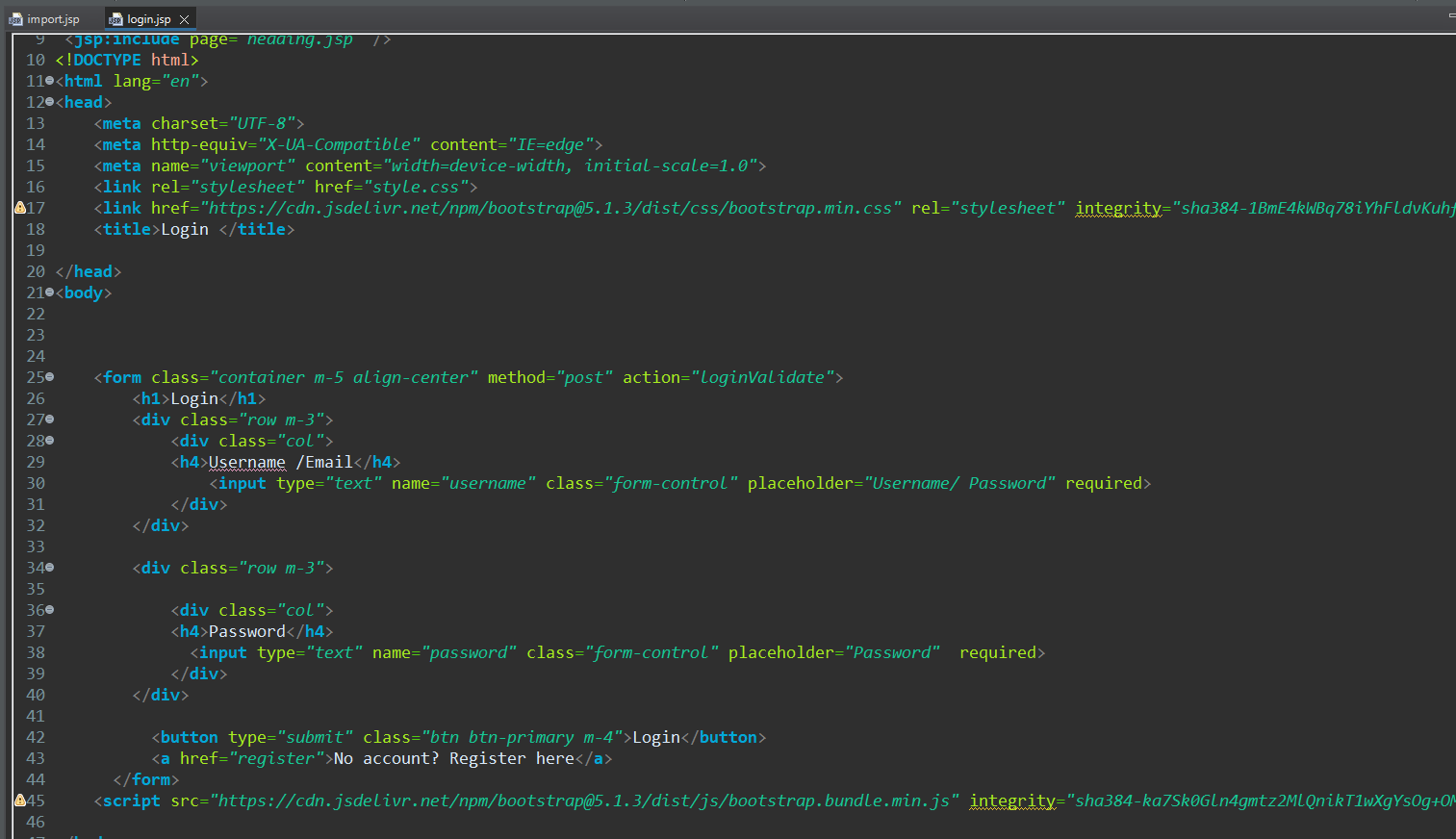
## Heading.jsp



## Index.jsp



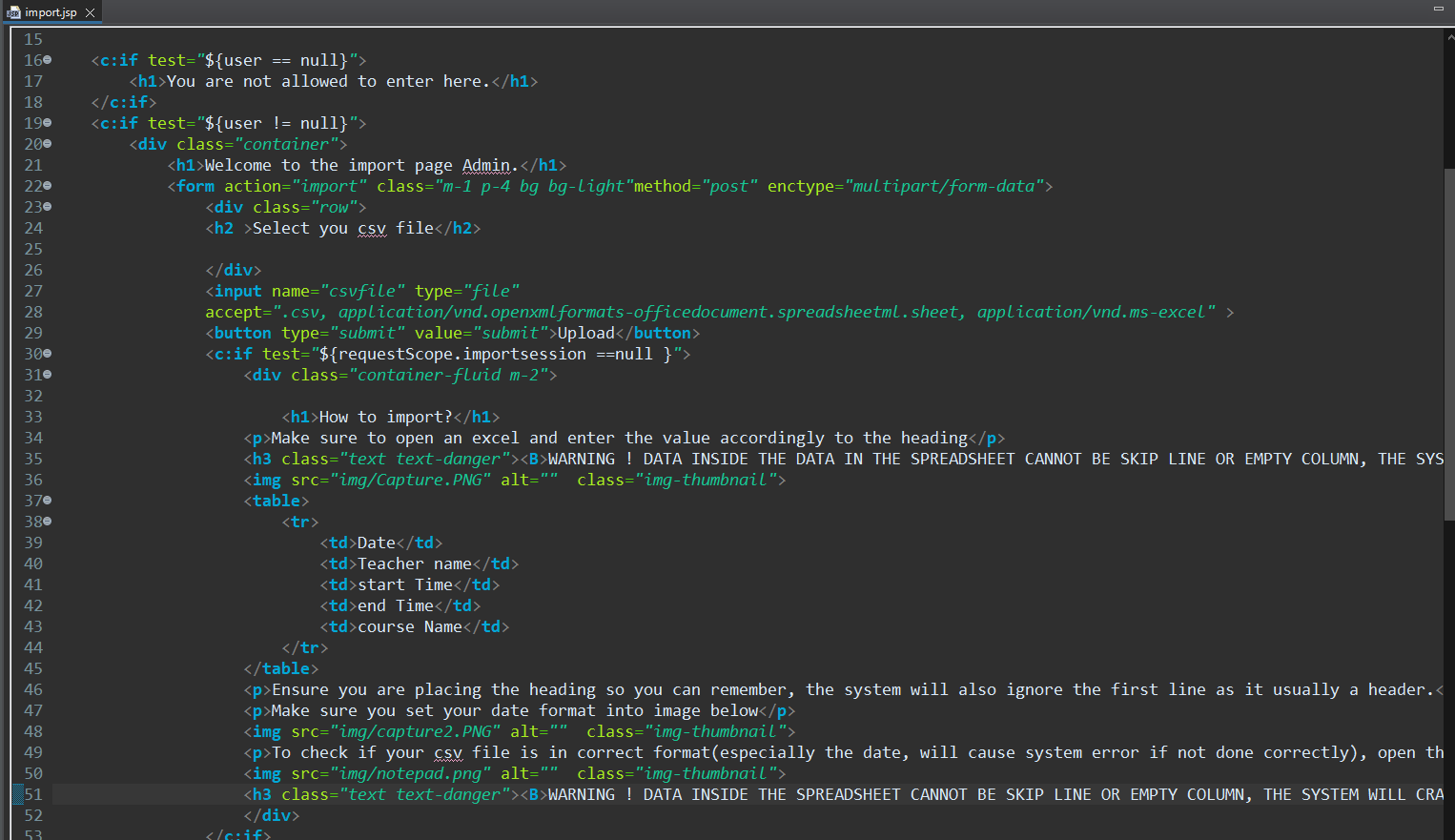
## Login.jsp

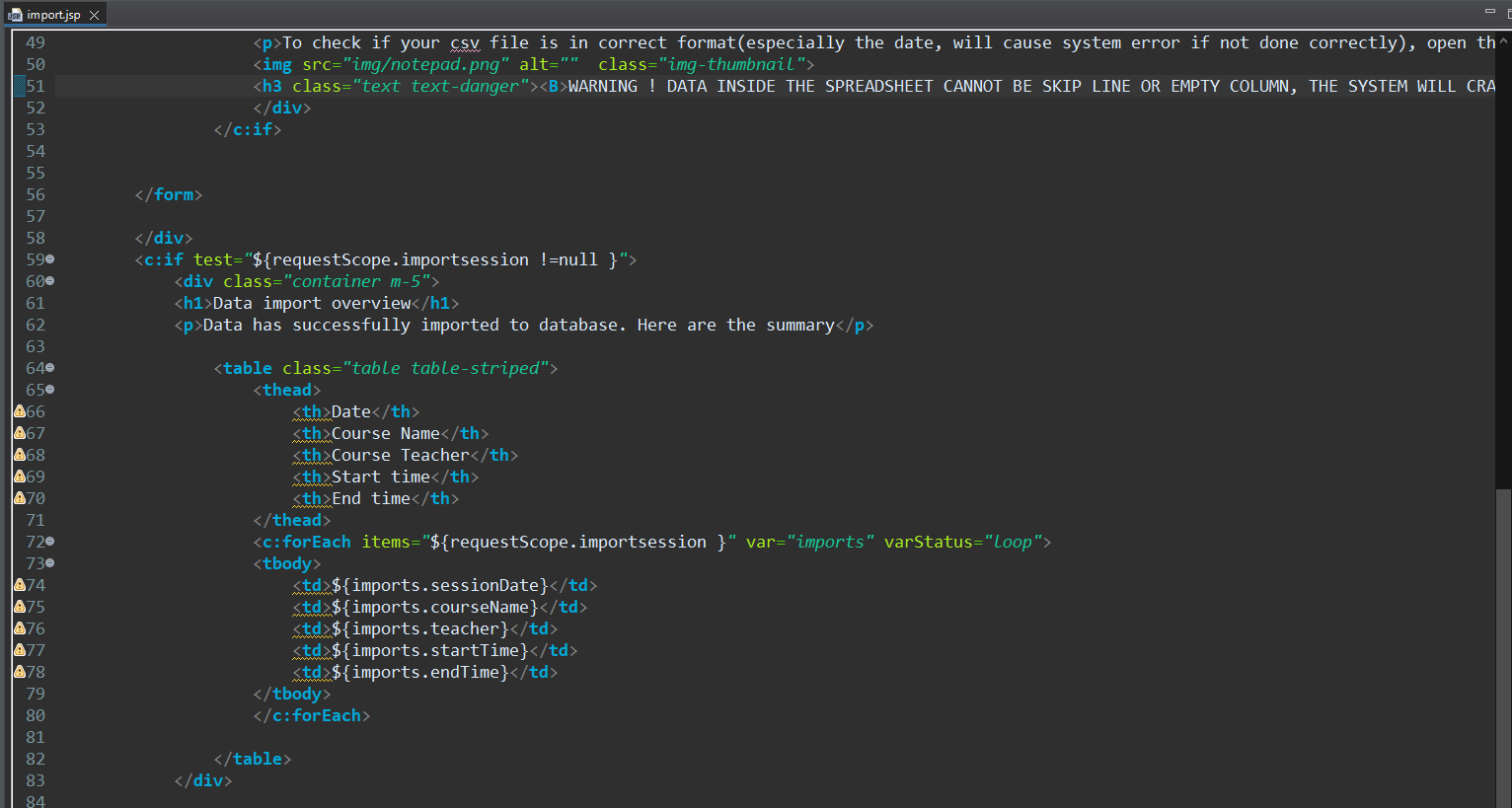


## Register.jsp

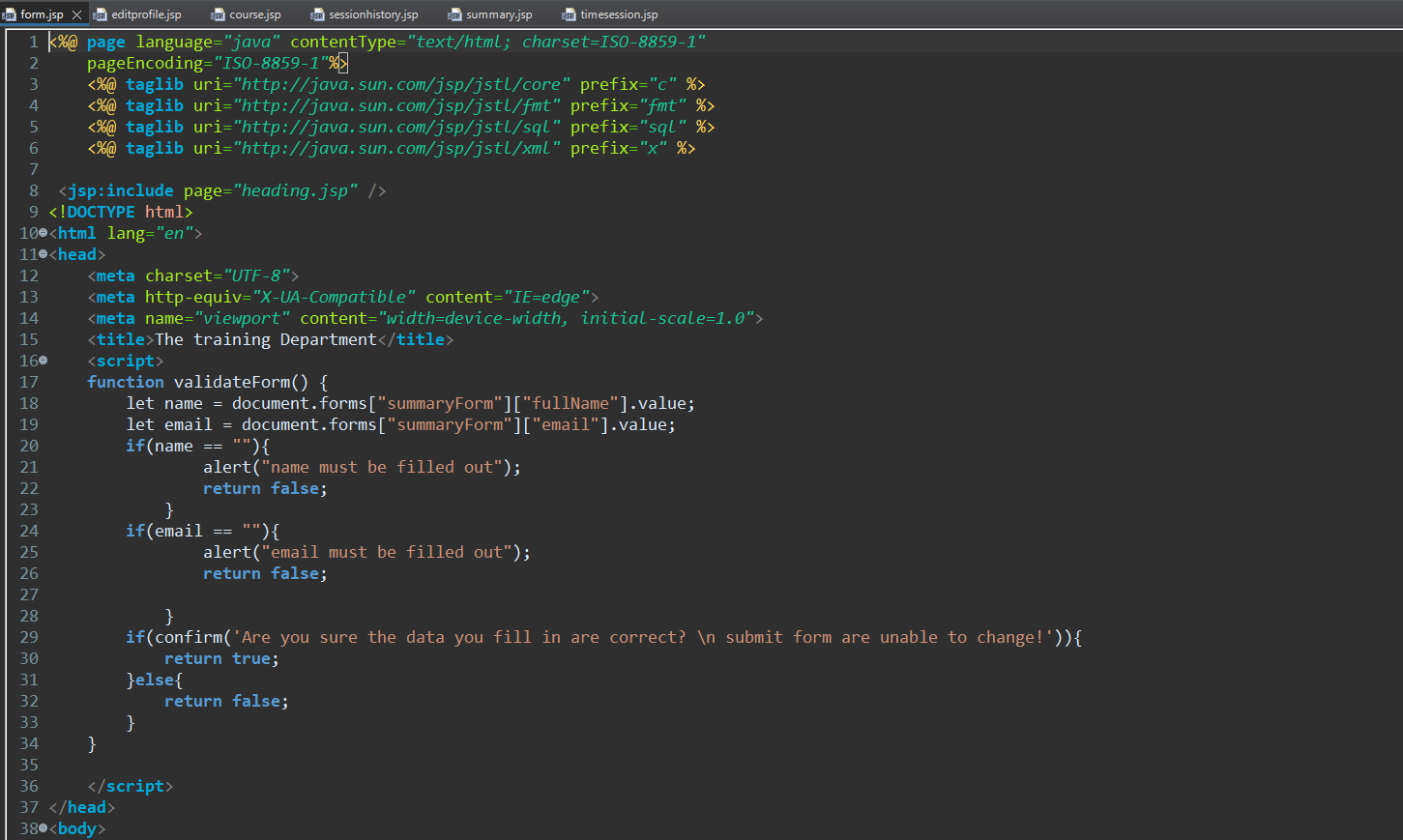


## Import.jsp

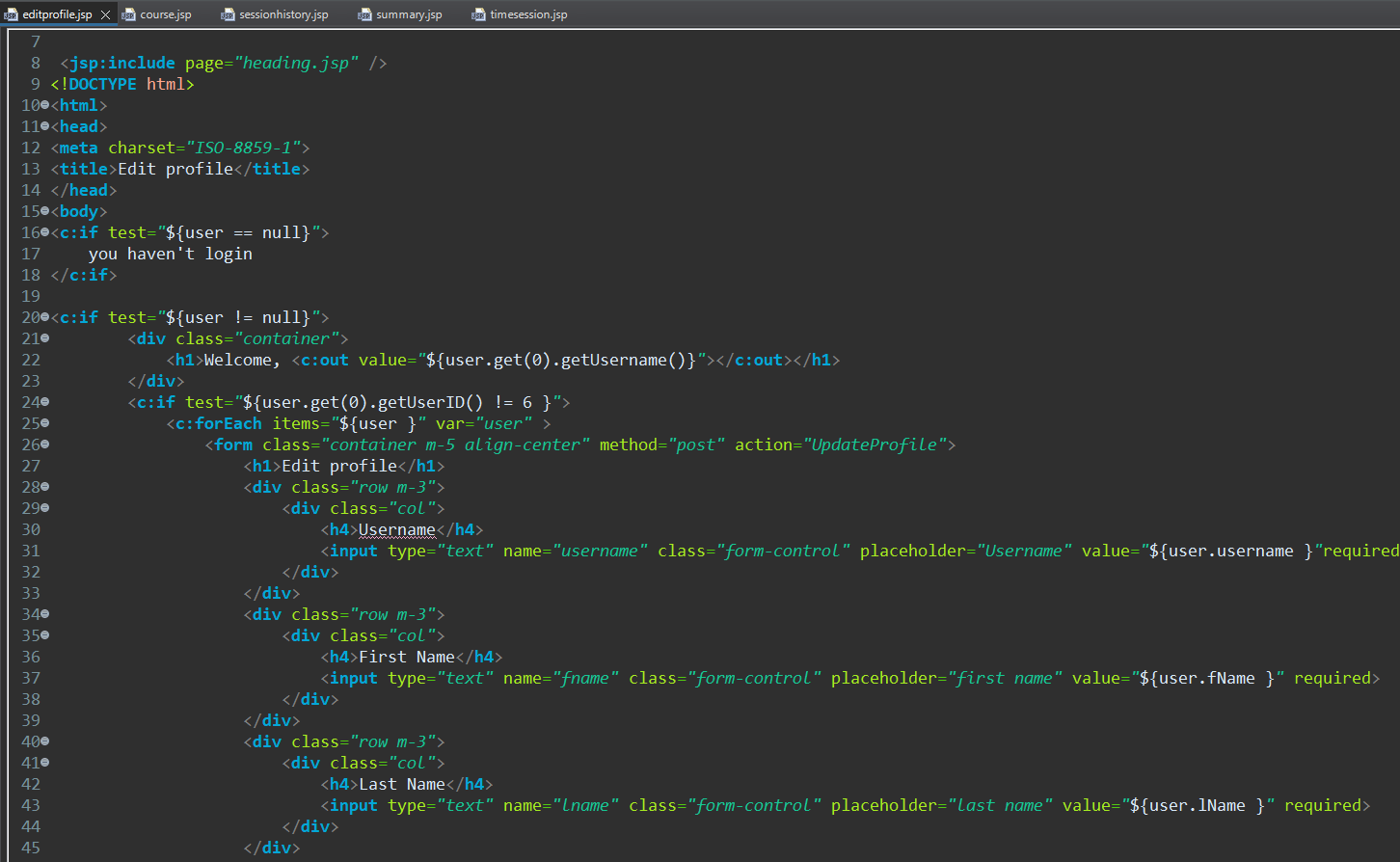




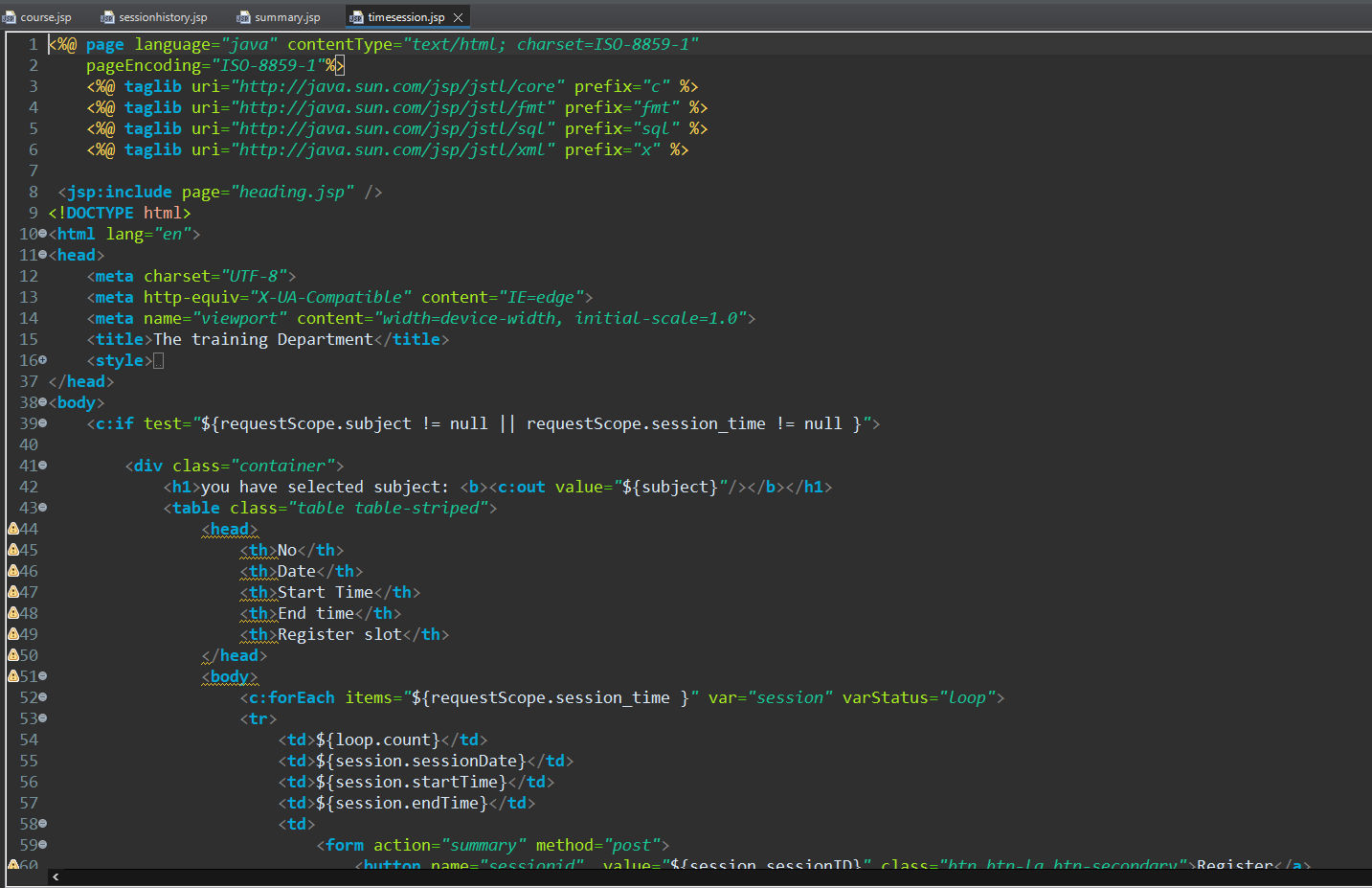
## Form.jsp



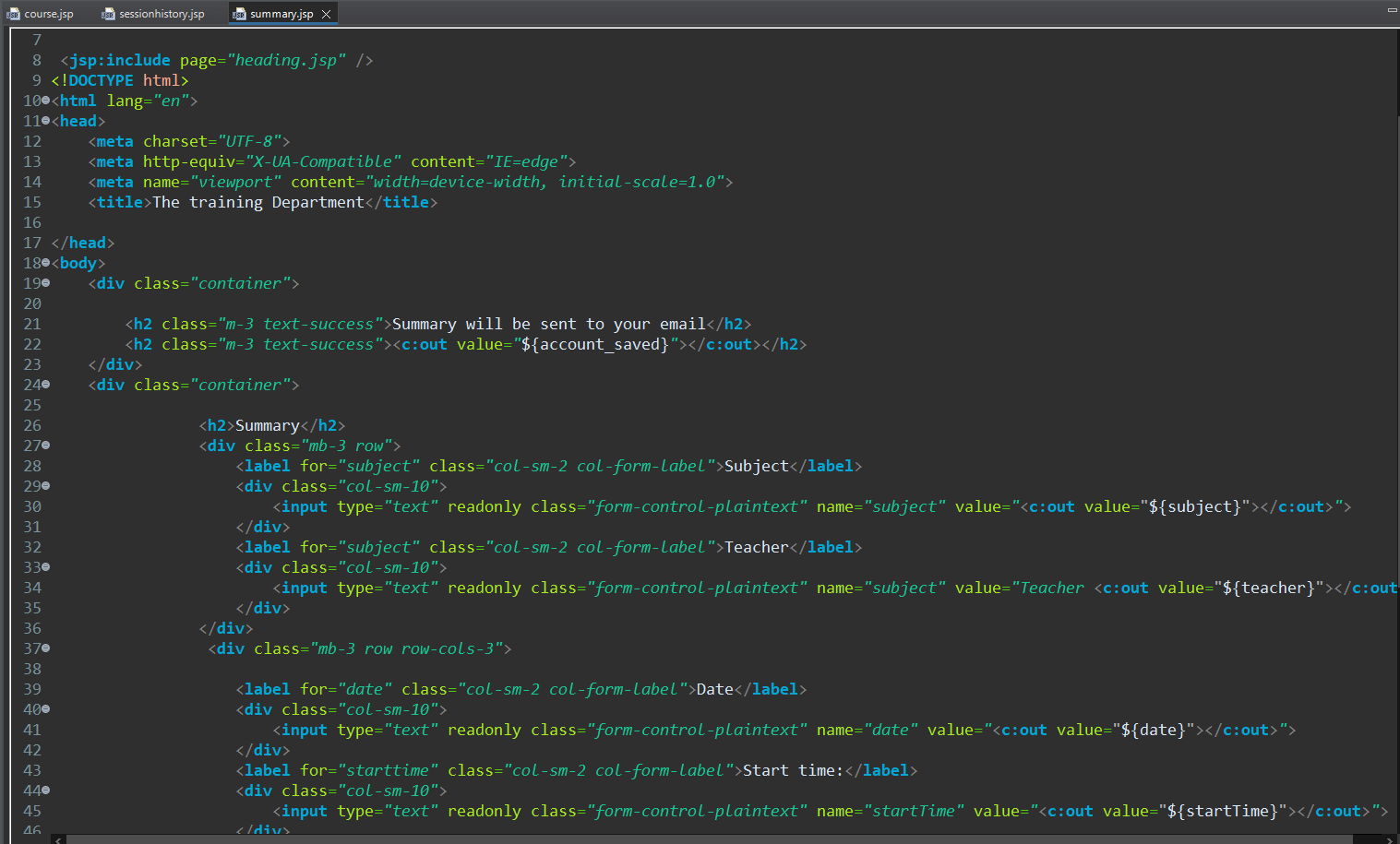
## Editprofile.jsp



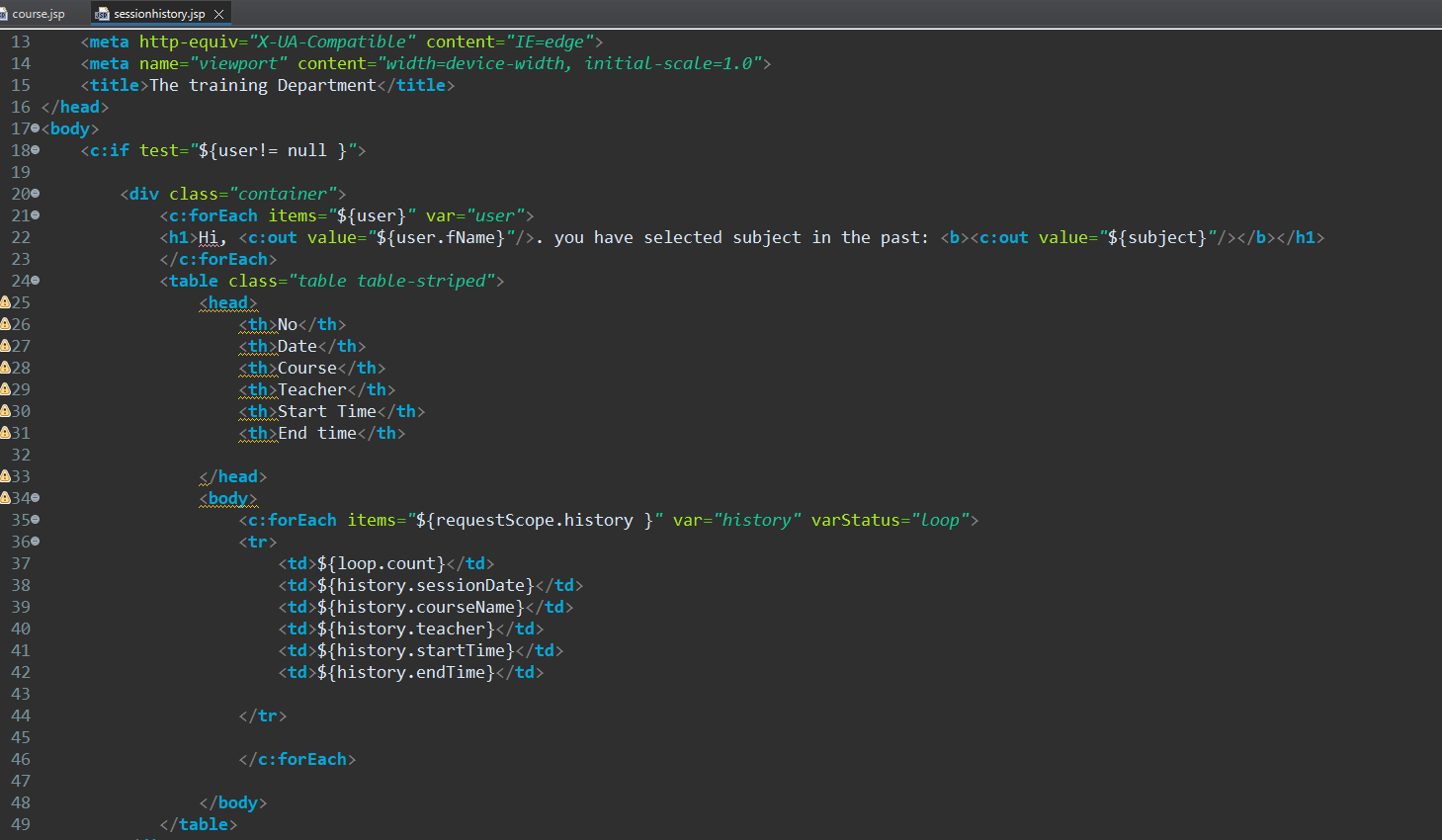
## Timesession.jsp



## Summary.jsp



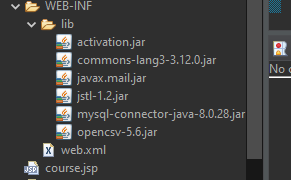
## Sessionhistory.jsp



## Course.jsp



# Library integrated in the system 3.0



Jstl 1.2.jar

Mysql. JDBC- 8.0.28

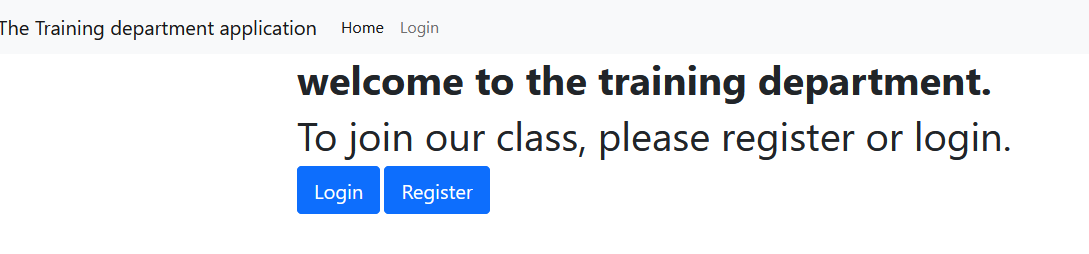
Opencsv- 5.6

Javax.mail

Commons-lang3

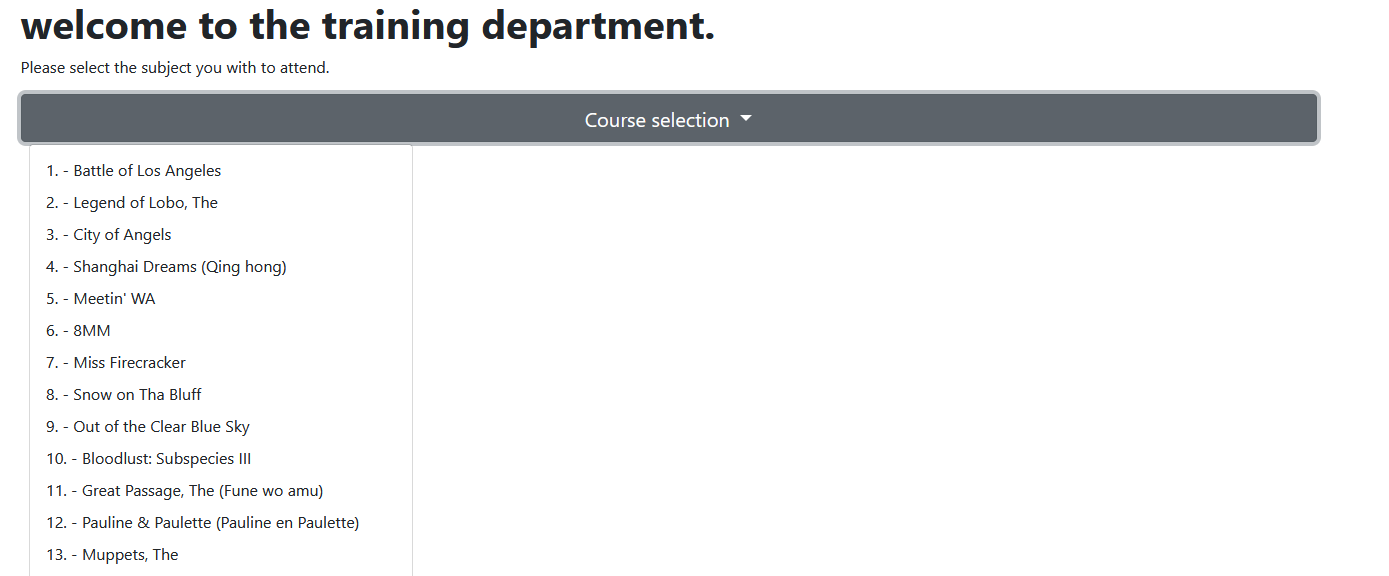
Activation

# Justify decisions made how the system will result in an effective solution.



The reason I would like to design a login/register system is I assumed that the company employees have their own emails. Hence, it is better to use this authorization system to track the users name and emails without retyping it if the course they want to join a lot of courses.

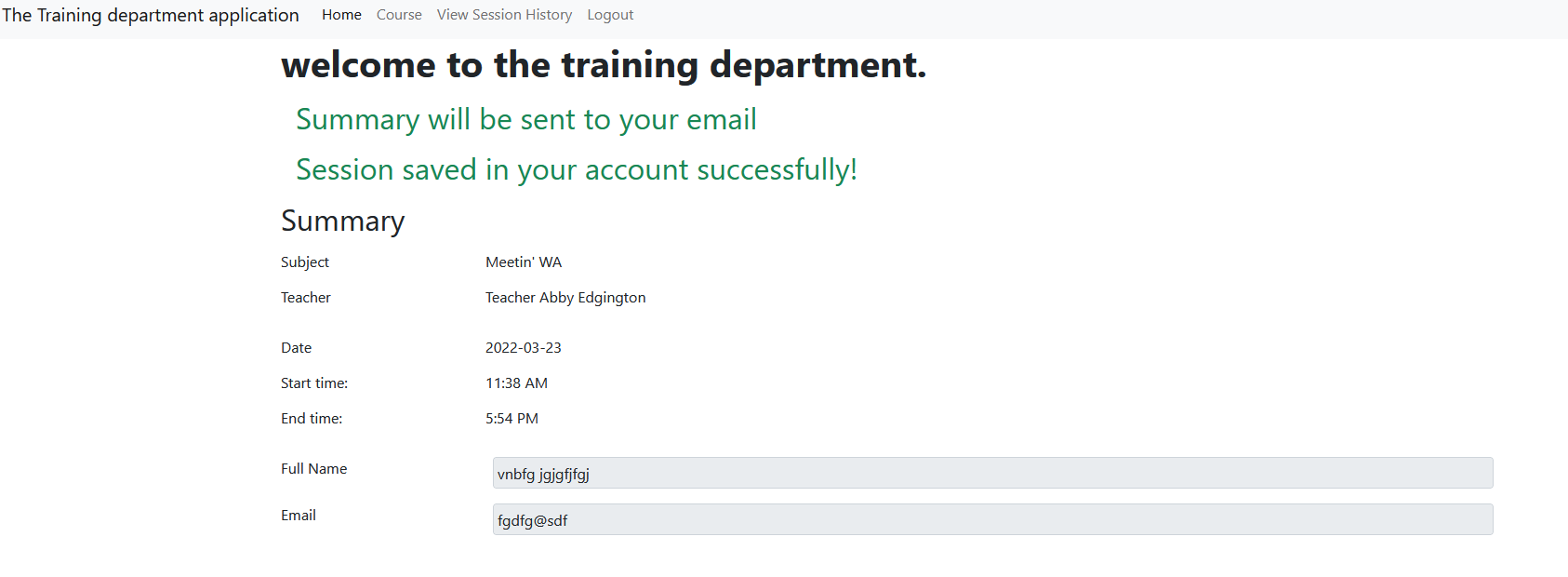
I decided to make users to select from **subject first** before selecting the time session is to make sure users that decided to join the **course subject** to register quickly without surfing around the website.



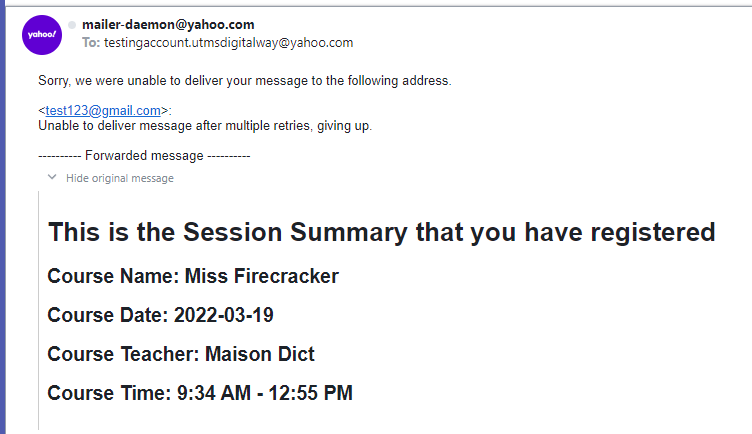
For the selection time session, I make sure the user is **really selected** the course they want. Hence, I show the selected subject in the interface.



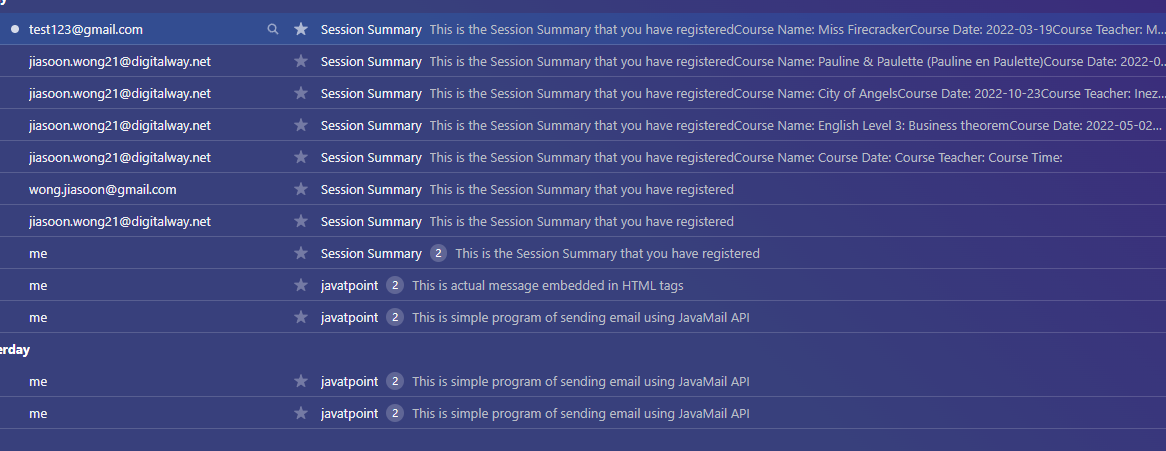
The date is sorted in ascending order so that the user will not be confused by the time showing on the table. I also separate out the start time and the end time to let user to read easily while making their time decision. The register button is made big to allow user to press easily.

For the summary page, page is made to show user the subject they will be choosing, date, time and the summary will be sent the user email. 

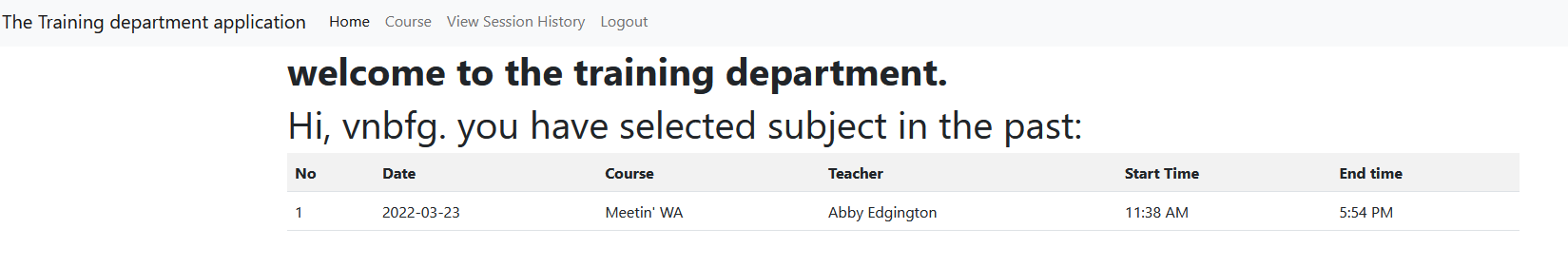
User can use it to double check the course they choose. In the email, the result will show like this:



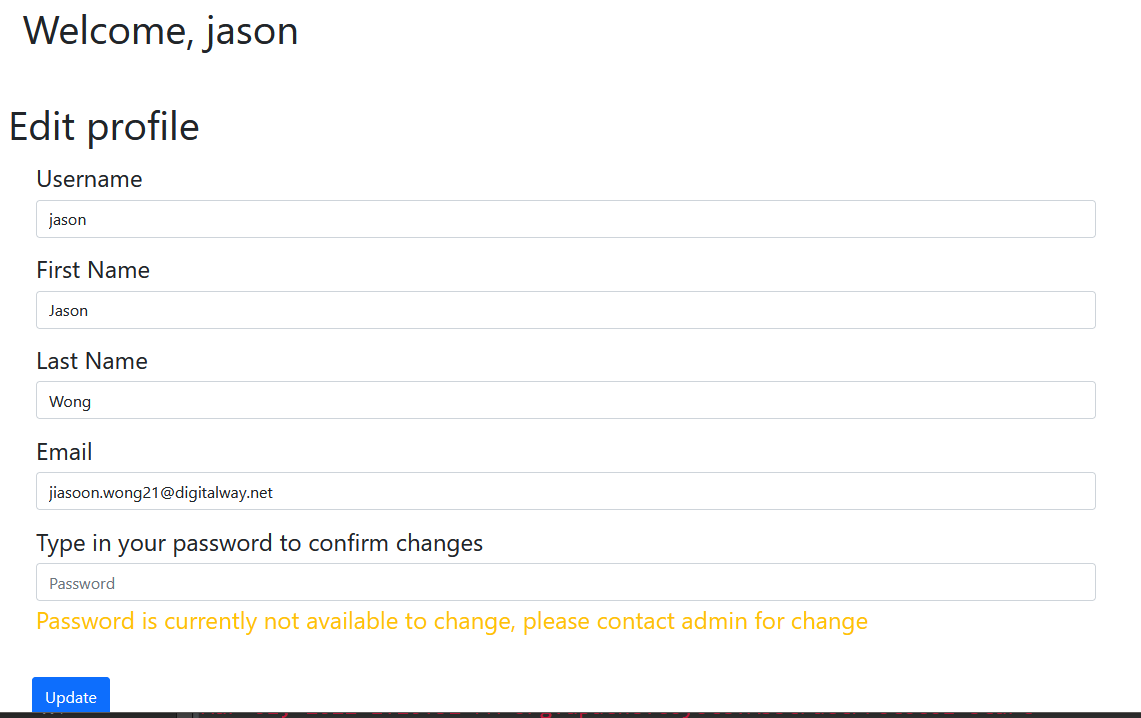
The summary is sent from yahoo email to others



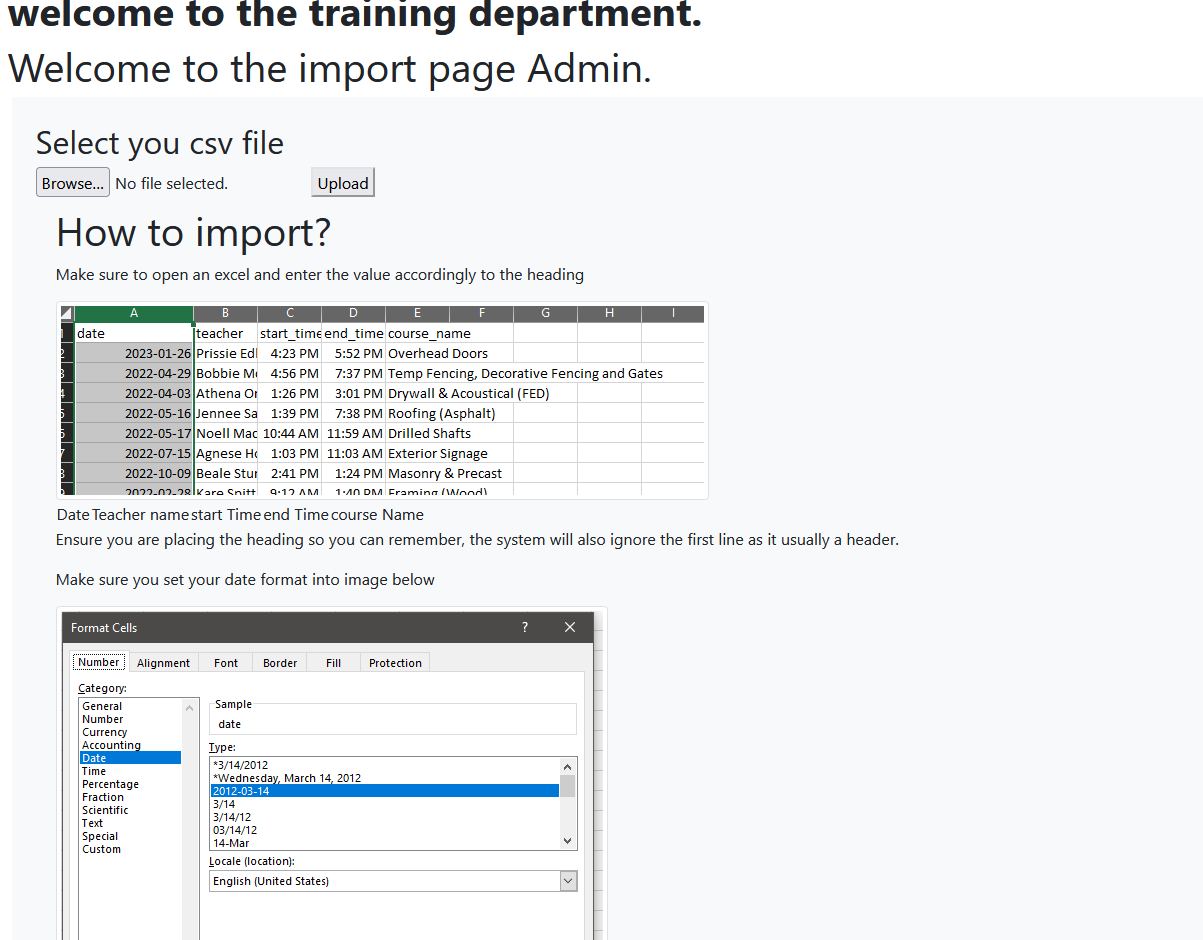
To revisit the summary they picked. User can visit the view session history page.



User can also edit their profile.



In the admin page, admin can import the csv into the database.



There are also tutorial on how to prepare the spreadsheet file to import to the database

## Future planning for the system

In future design and upgrade I would like to:

* Add admin system
  + Create session
  + Edit session
  + Delete session
* Add edit user profile
  + Add gender, age, department area
* Add class learners quantity to the session time
* Add function to filter the class that is already finish
  + View history session have the option to hide the course that are dismissed or done.
* Add status on the session class for **complete, done, cancelled**
* If possible, add some company update news to the system.