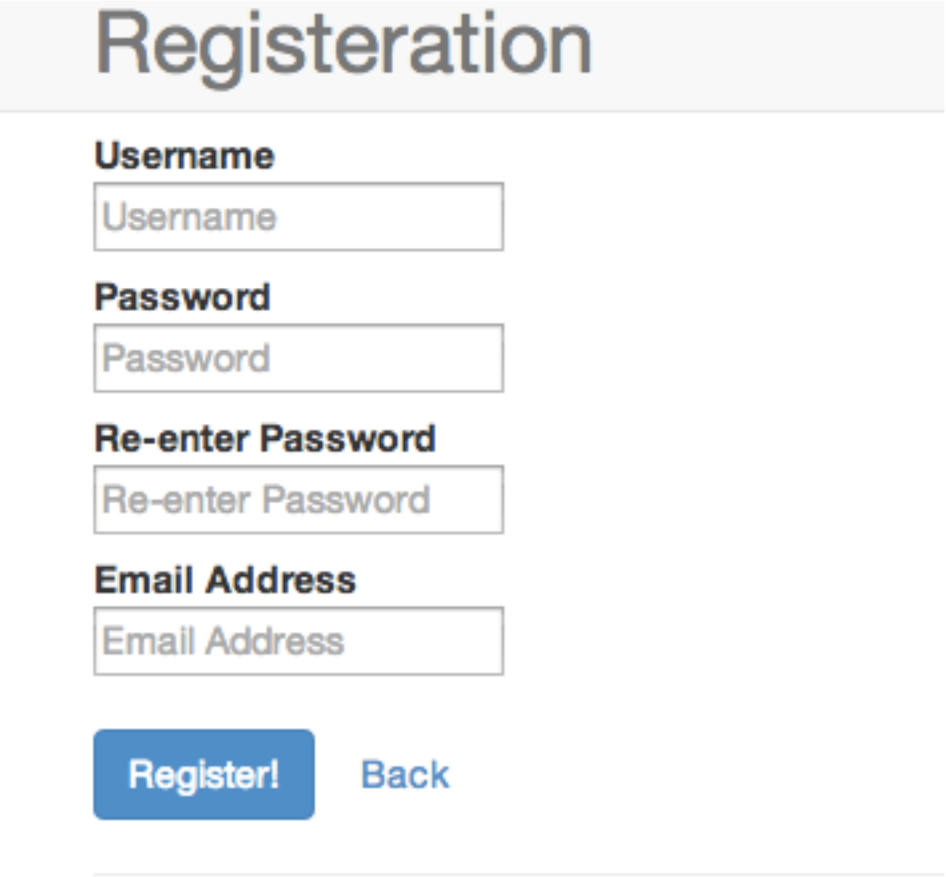
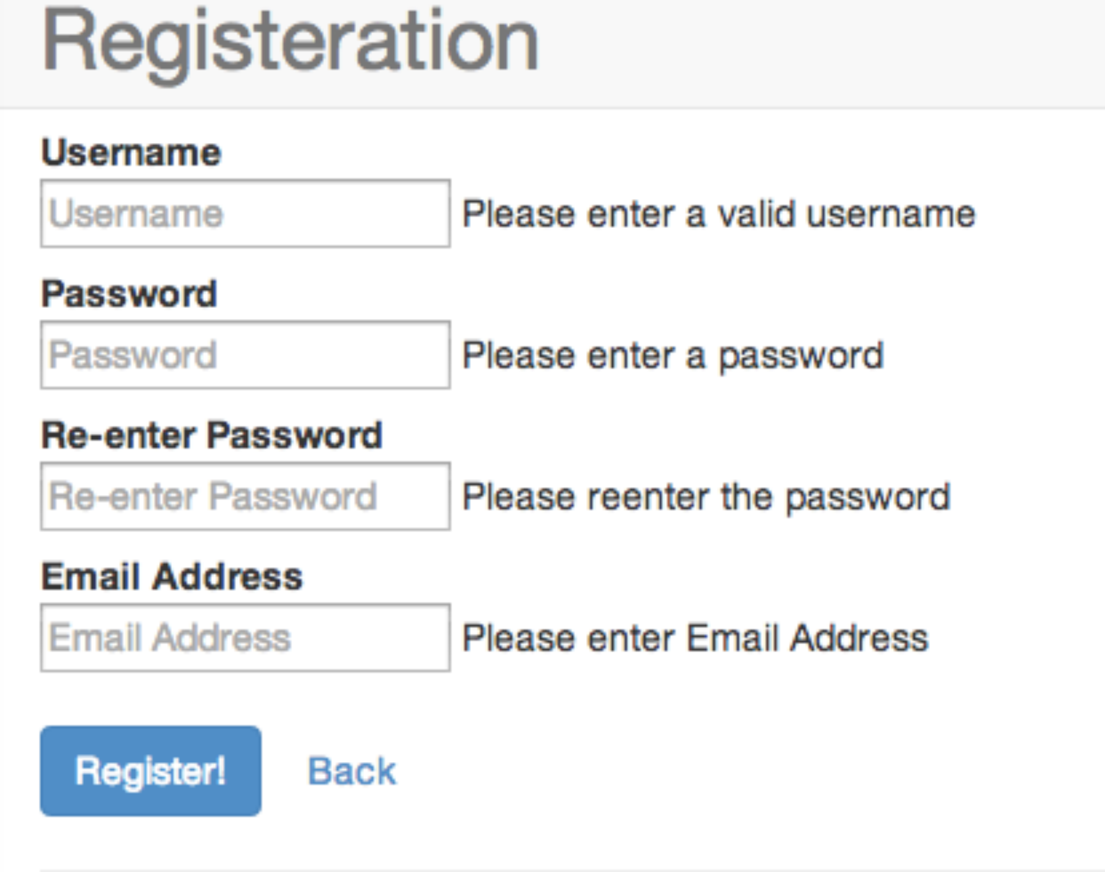
Final Project Manual Test-Plan

Test 1:
In our registration page, we allow users to fill out their username, password and email and register to the website. If the user submits invalid input, inline error messages are shown. On submitting valid input, the users are now allowed to login to the website.



When the user submits a blank form, the registration page should print out inline error messages for each of the inputs:



Registeration

Username

Username

Please enter a valid username

Password

Password

Please enter a password

Re-enter Password

Re-enter Password

Please reenter the password

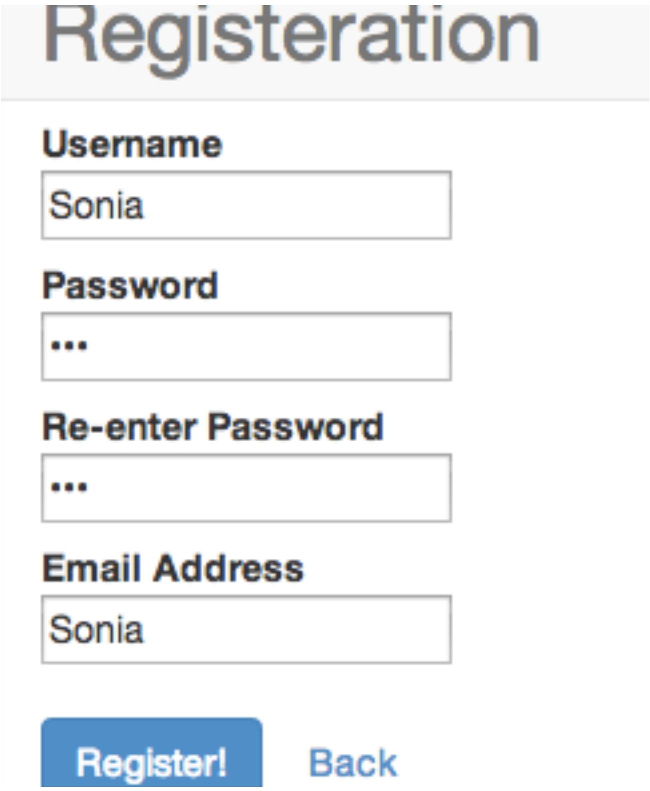
Email Address

Email Address

Please enter Email Address

Test 2:
Incase the user enters invalid input, say text instead of email address, the page should still display an error message next to the email address.

Ex:



Registeration

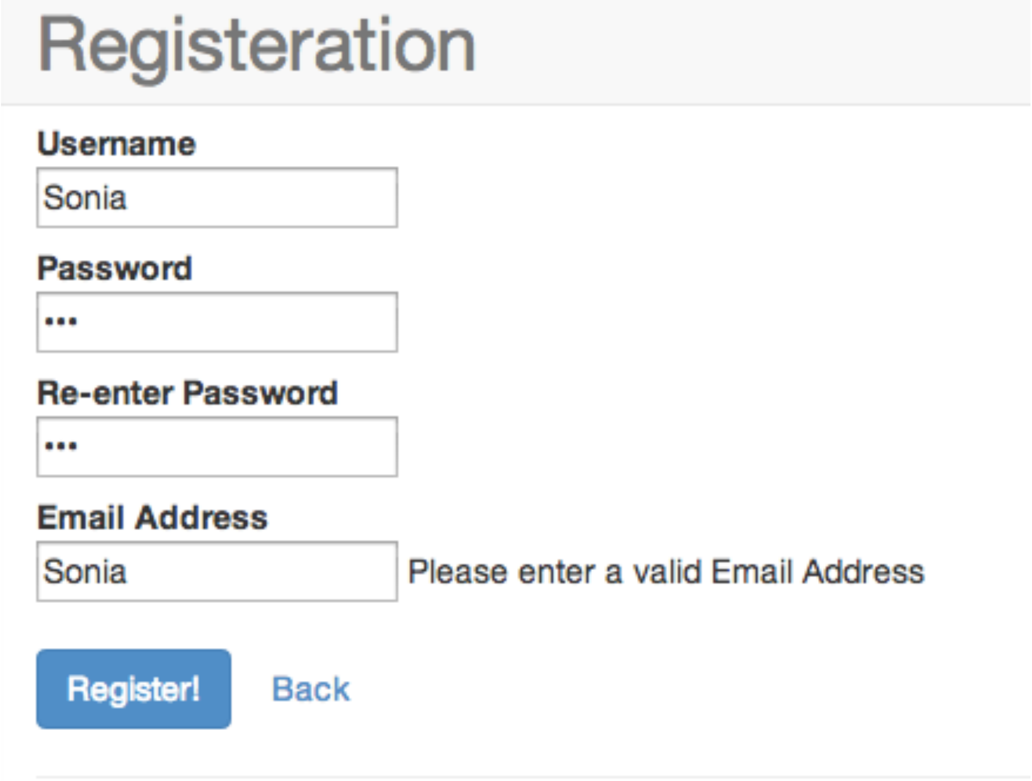
Username
Sonia

Password
•••

Re-enter Password
•••

Email Address
Sonia

After clicking “Register” the page should display an error message for the email address:



Registeration

Username
Sonia

Password
•••

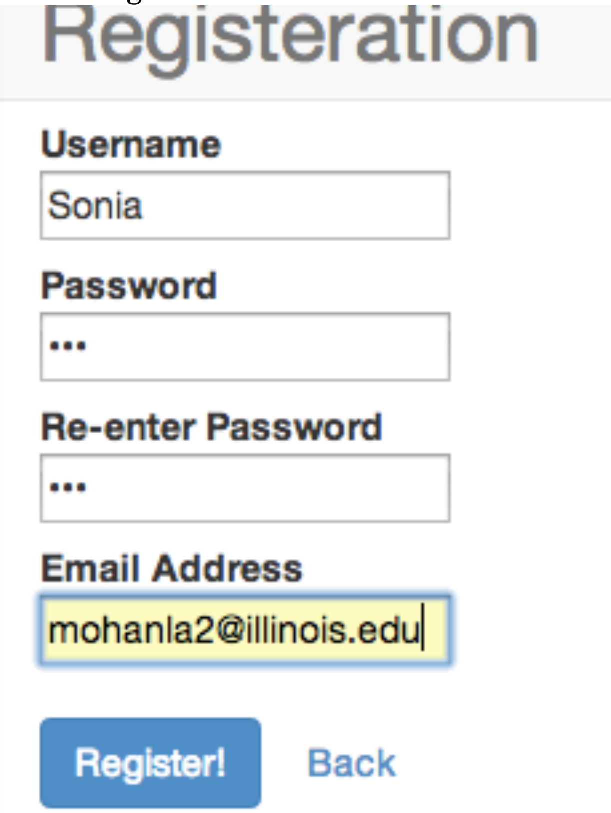
Re-enter Password
•••

Email Address
Sonia

Please enter a valid Email Address

Test 3:
If the user puts in valid values for all the textboxes, the user should be redirected to the main page and registered to the website!

Before registration:



Registeration

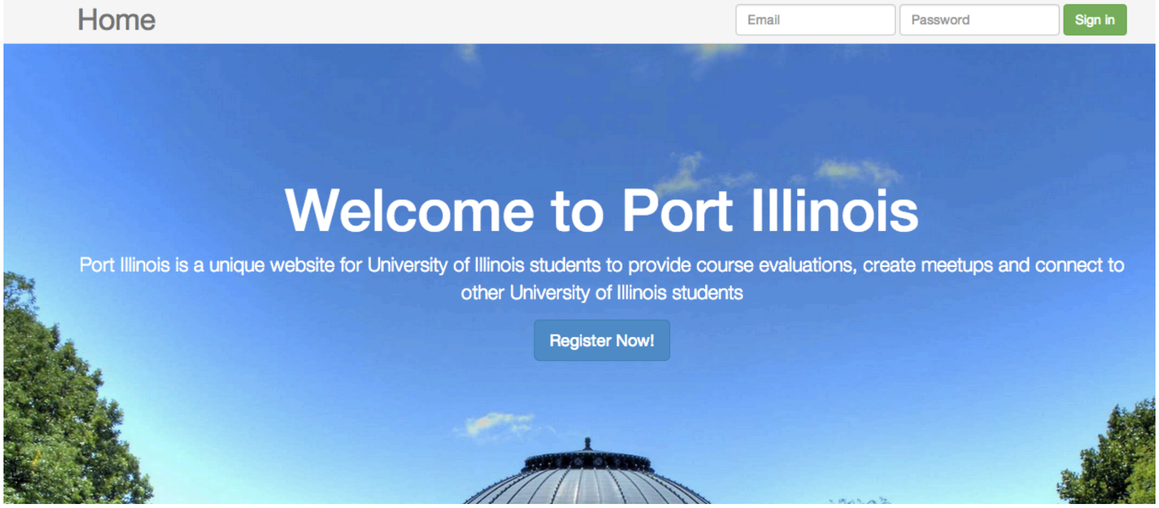
Username
Sonia

Password
•••

Re-enter Password
•••

Email Address
mohanla2@illinois.edu

After registration:



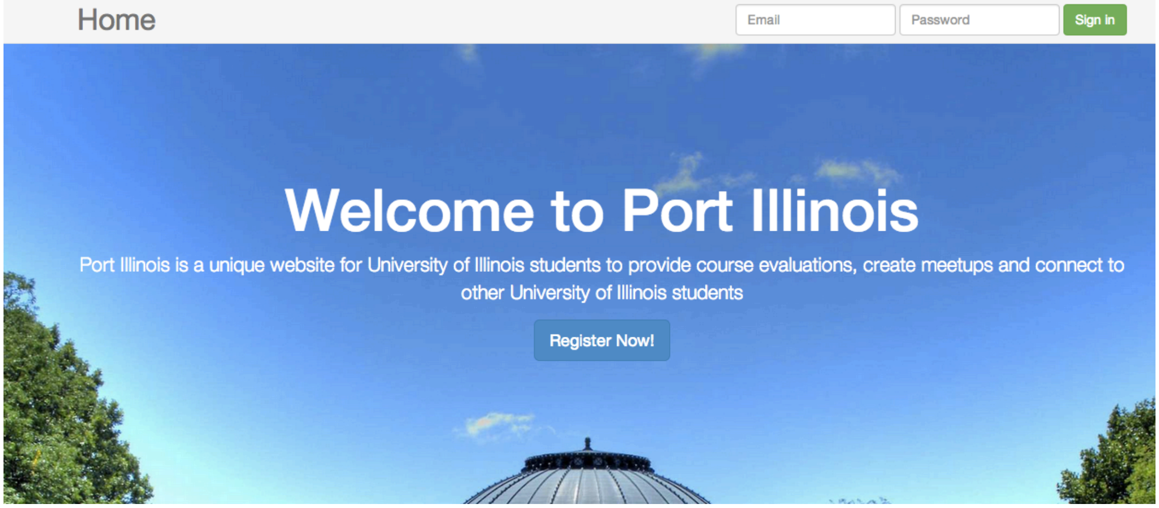
Welcome to Port Illinois

Port Illinois is a unique website for University of Illinois students to provide course evaluations, create meetups and connect to other University of Illinois students

Test 4:

On our main home page, when a user inputs either empty or incorrect email/password combination and clicks the signin button, the webpage should display an error message:

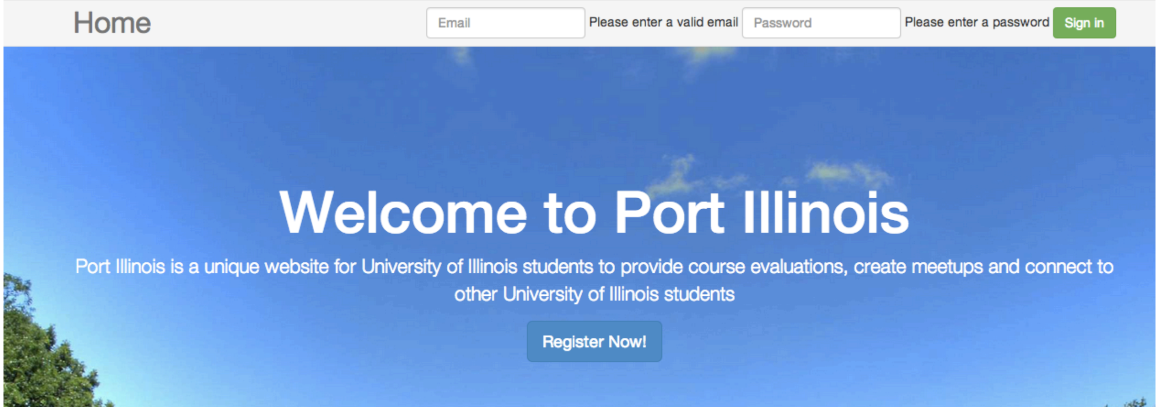
Before login:



Welcome to Port Illinois

Port Illinois is a unique website for University of Illinois students to provide course evaluations, create meetups and connect to other University of Illinois students

After login (with empty input):



Please enter a valid email

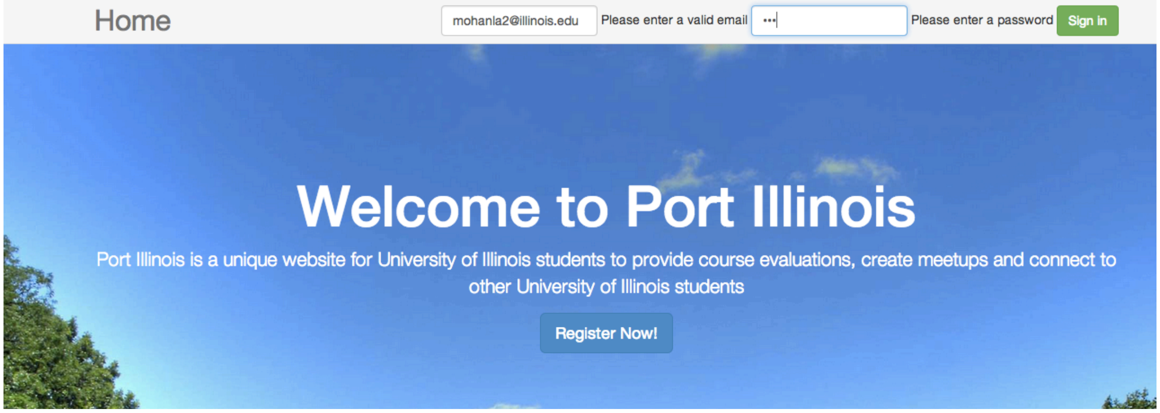
Please enter a password

Welcome to Port Illinois

Port Illinois is a unique website for University of Illinois students to provide course evaluations, create meetups and connect to other University of Illinois students

Test5:
If the user enters correct login-info and clicks the sign-in button, he should be redirected to the courses page with a header on the top welcoming him!

Before pressing “sign in”:

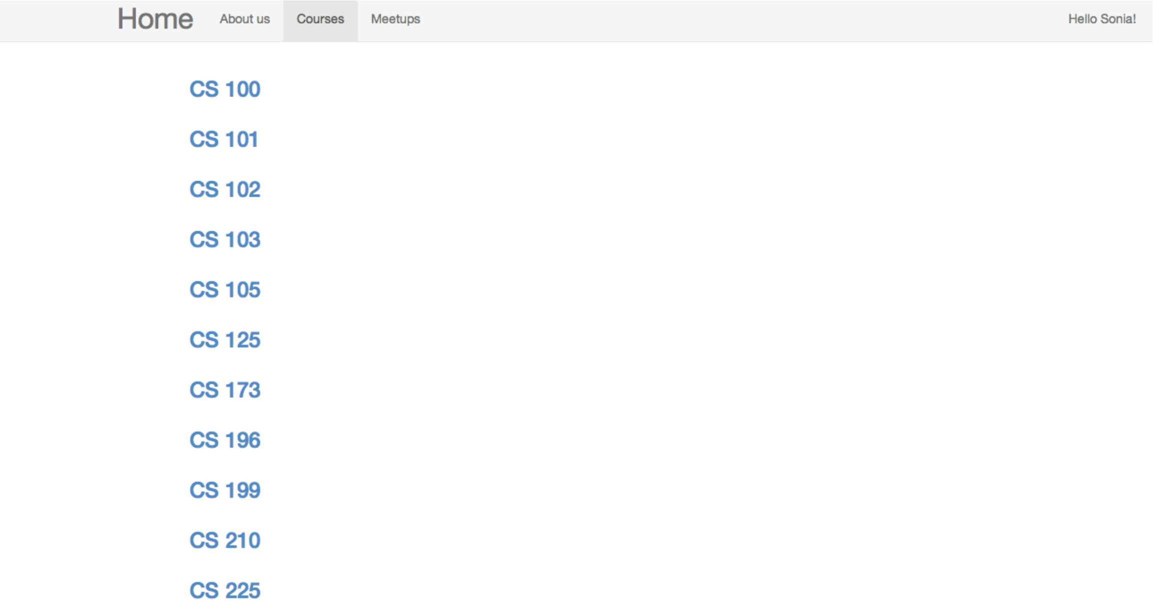


mohania2@illinois.edu Please enter a valid email ••• Please enter a password

Welcome to Port Illinois

Port Illinois is a unique website for University of Illinois students to provide course evaluations, create meetups and connect to other University of Illinois students

After pressing sign-in:



CS 100

CS 101

CS 102

CS 103

CS 105

CS 125

CS 173

CS 196

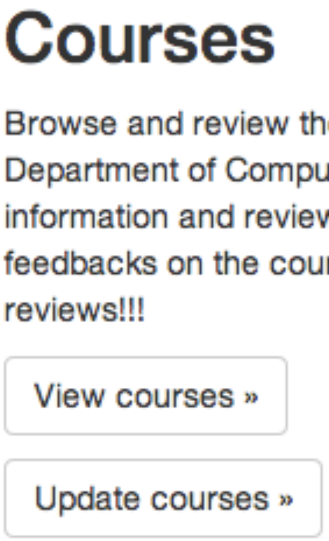
CS 199

CS 210

CS 225

Test 6:
On the main homepage, when you click on the “view courses” button, you get redirected to a page containing the list of all CS classes:

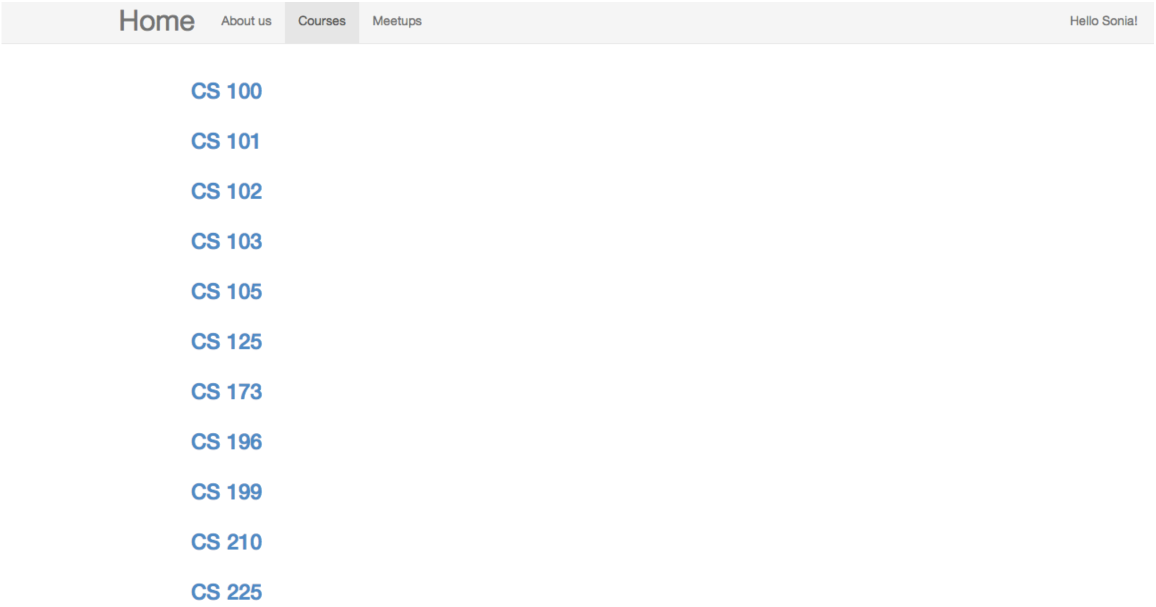
Before pressing the “View courses” button:



Courses

Browse and review the classes offered in the Department of Computer Science. Get course information and reviews posted by your peers to get feedbacks on the course. Don't forget to add your own reviews!!!

After pressing the "view courses" button:



CS 100

CS 101

CS 102

CS 103

CS 105

CS 125

CS 173

CS 196

CS 199

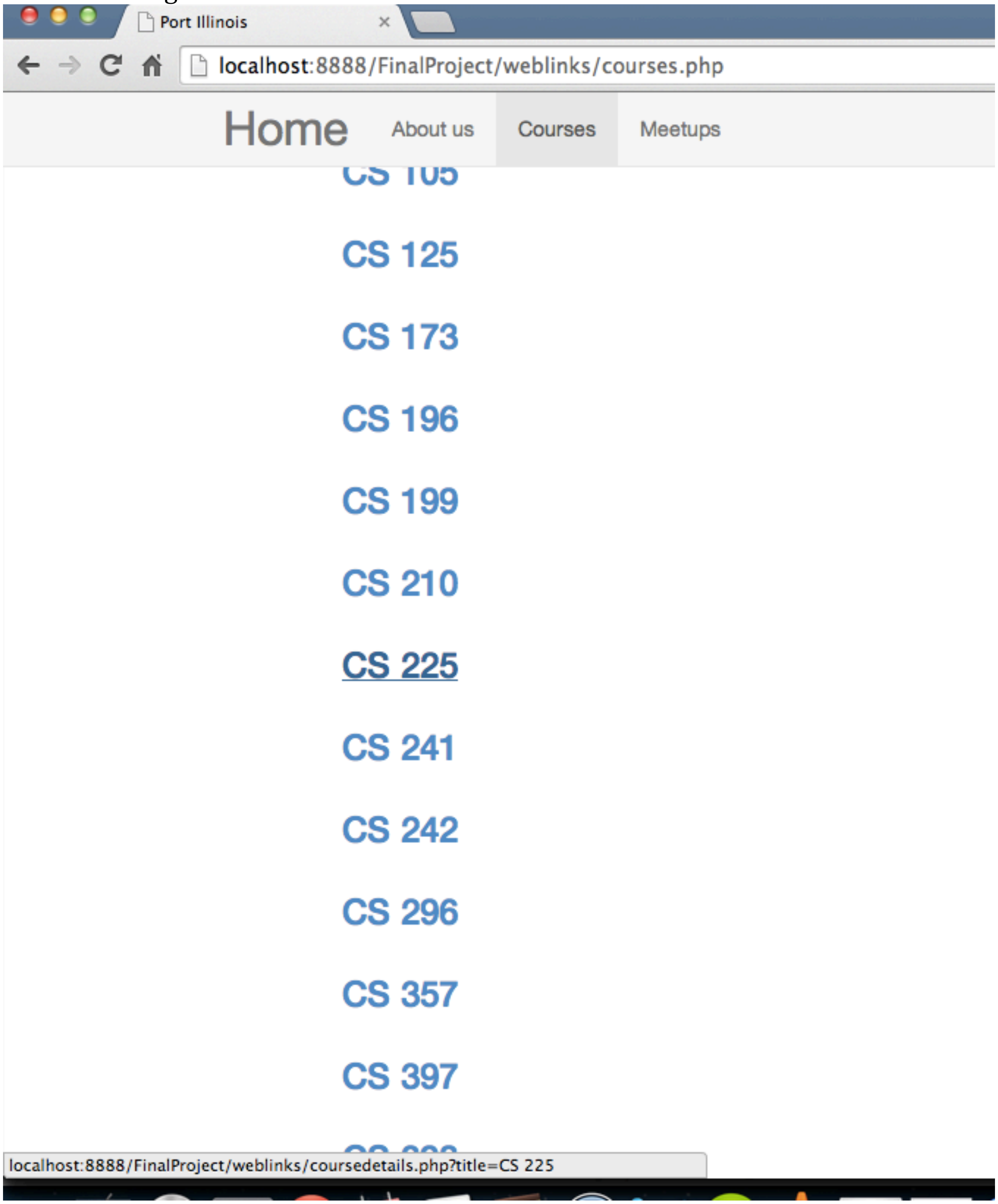
CS 210

CS 225

Test7:
Each course is a link, which when clicked, should display 3 things:

- Course Title
- Course Description
- Course Credit Hour
- An "add review" section

Before clicking “CS 225”:



localhost:8888/FinalProject/weblinks/courses.php

CS 105

CS 125

CS 173

CS 196

CS 199

CS 210

CS 225

CS 241

CS 242

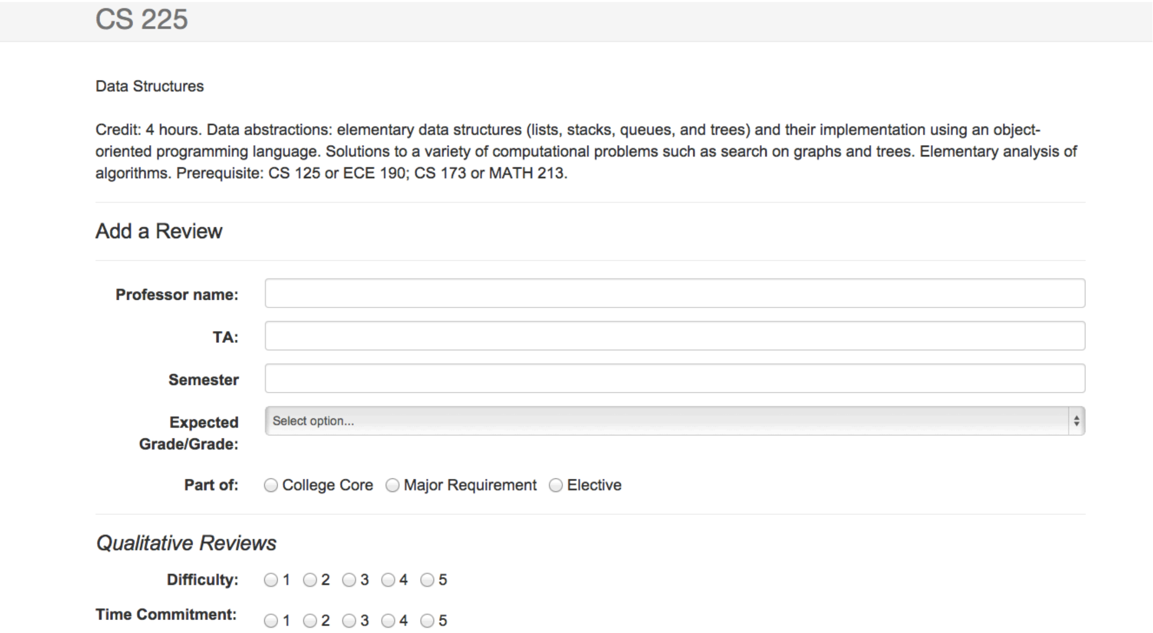
CS 296

CS 357

CS 397

localhost:8888/FinalProject/weblinks/coursedetails.php?title=CS 225

After clicking “CS 225”:



CS 225

Data Structures

Credit: 4 hours. Data abstractions: elementary data structures (lists, stacks, queues, and trees) and their implementation using an object-oriented programming language. Solutions to a variety of computational problems such as search on graphs and trees. Elementary analysis of algorithms. Prerequisite: CS 125 or ECE 190; CS 173 or MATH 213.

Add a Review

Professor name:

TA:

Semester

Expected Grade/Grade: Select option...

Part of: ○ College Core ○ Major Requirement ○ Elective

Qualitative Reviews

Difficulty: ○ 1 ○ 2 ○ 3 ○ 4 ○ 5

Time Commitment: ○ 1 ○ 2 ○ 3 ○ 4 ○ 5