CourseNebula

Software Requirements Specification Document

Group 2
Tanner Marino, Brandon Geren, Michael Shanahan, Ayush Manish Agrawal

Date	Changes	Version
01/31/17	Initial document	1.0
02/08/17	Revisions from feedback	1.1
02/10/17	Further revisions from feedback; final editing pass	1.2

Table of Contents

1. Introduction

1.1. Purpose

The purpose of the document is to explain the requirements of the application CourseNebula, along with its features.

1.2. Scope

The project, CourseNebula, will be a web-based forum application where students can find accurate information regarding the UNL Computer Science and Computer Engineering classes they plan to sign up for. The information will be provided by TAs along with their peers who have already taken the course. They will also be able to contribute their own information regarding classes and professors.

1.3. Definitions, acronyms, and abbreviations

1.3.1. CourseNebula is the name of the application.

1.3.2. A modern web browser is any web browser that is currently supported and is capable of running HTML5 and CSS3.

1.4. References

1.5. Overview

The following sections will provide a description of the project, CourseNebula. Section 3 will provide a set of specific requirements that go into detail regarding the exact design of the project.

2. Overall Description

2.1. Product perspective

2.1.1. System interfaces

The project will interface with modern desktop web browsers, such as Firefox and Google Chrome.

2.1.2. User interfaces

The UI will be designed to be navigated through mouse and keyboard with a style suited to desktop monitors.

2.1.3. Hardware Interfaces

All hardware capable of running a modern web browser will be capable of running the project.

2.1.4. Software interfaces

The only software necessary to run the project is a modern web browser.

2.1.5. Communications interfaces

Connection to the internet will be necessary to run the project.

2.1.6. Memory constraints

The memory constraints will be most impacted by the back-end software and the MySQL database. As more classes are added, more server space will be necessary.

2.1.7. Operations

Users will have four primary modes of operation. The first mode is creating an account. The second mode is viewing and leaving feedback on classes they are interested in. The third mode is asking and answering questions regarding classes. The fourth mode is rating feedback and questions they have viewed.

2.1.8. Site adaptation requirements

No site adaptations will be necessary.

2.2. Product functions

The web app is independent.

2.3. User characteristics

Users will be CSE students at UNL. The application will be most useful to freshmen, sophomore, and juniors since these students will be most interested in learning about their professor and what the class is like.

2.4. Constraints

As the project is a web app, it will only run on web browsers. In the first phase, the project will be designed for desktop use and may not perform well on mobile browsers.

2.5. Assumptions and dependencies

On the front end, the only dependence is a modern web browser that supports HTML5 and CSS3, which all do. The MySQL database and backend software will depend on the server hosting them.

2.6. Apportioning of requirements

A mobile-friendly user interface will be required in subsequent releases. Non-CSE classes can be added into the system in subsequent releases as well.

3. Specific Requirements

3.1. External interface requirements

3.1.1. User interfaces

- 3.1.1.1. Account Registration Form
- 3.1.1.2. Course Selection Page
  - 3.1.1.2.1. A button for each Course in the system
- 3.1.1.3. Course Page
  - 3.1.1.3.1. A section for Course Info
  - 3.1.1.3.2. A section for Feedback (3.2.2)
    - 3.1.1.3.2.1. A button to Leave Feedback
  - 3.1.1.3.3. A section for Questions (3.2.3)
    - 3.1.1.3.3.1. A button to Ask Question

3.1.2. Hardware interfaces

The project will not require any particular hardware considerations.

3.1.3. Software interfaces

- 3.1.3.1. The CourseNebula application will be able to communicate with modern-day web browsers, such as Google Chrome and Mozilla Firefox. In theory, any device capable of using those browsers should be able to use this website.

3.1.4. Communications interfaces

3.1.4.1. This application will communicate over the Internet to users with computers and other internet-capable devices.

3.2. System features

3.2.1. User Registration

3.2.1.1. Introduction/Purpose of feature

The project will be semi-anonymous, allowing users to register under pseudonyms which they can be identified by thereafter. User registration allows us to keep tabs on user activity and require permission to access certain parts of the site.

3.2.1.2. Stimulus/Response sequence

Stimulus: Clicking a ‘Create Account’ button
Response: Being taken to ‘Account Registration Form’
Stimulus: Filling account registration form
Response: Being presented with a ‘Complete Account Creation’ button
Stimulus: Clicking ‘Complete Account Creation’ button
Response: Send verification email and inform user
Stimulus: User clicks verification link
Response: User is told account is verified and has access to site

3.2.1.3. Associated functional requirements

3.2.1.3.1. The system shall allow users to register new accounts.

3.2.1.3.2. The system shall block banned users from registering new accounts by checking IP.

3.2.1.3.3. The system shall store passwords securely.

3.2.2. Feedback

3.2.2.1. Introduction/Purpose of feature

The project will help students to choose CSE courses for future semesters through providing them with feedback left by students who have taken the course before. Feedback is distinct from questions (3.2.3) because they are a general impression of the course rather than an answer to a prospective student’s specific question.

3.2.2.2. Stimulus/Response sequence

Stimulus: User sees course they are interested in on ‘Course Selection Page’

Response: User clicks the course on ‘Course Selection Page’

Stimulus: User is presented with course info and feedback in ‘Course Page’

Response: User reads course information and remaining feedback

Stimulus: User runs out of feedback on page

Response: User clicks the ‘Next Page’ button to view more feedback

Stimulus: User runs out of feedback on course

Response: User clicks the ‘Return to Course List’ button to return to ‘Course Selection Page’

Stimulus: User wants to leave feedback on a course they’ve taken before

Response: User clicks the course on ‘Course Selection Page’

Stimulus: User is presented with ‘Leave Feedback’ button

Response: User clicks or ignores ‘Leave Feedback’ button

3.2.2.3. Associated functional requirements

3.2.2.3.1. The system shall provide users with comments and documents left by other users.

3.2.2.3.2. The system shall allow users to leave feedback on classes they’ve previously taken.

3.2.2.3.3. The system shall allow users to rate feedback left by other users in a five-star rating method.

3.2.3. Questions

3.2.3.1. Introduction/Purpose of feature

The purpose of this feature is to allow users to ask a public question and then receive a notification when answered by other users. All questions asked in a class will be viewable to registered users.

Stimulus/Response sequence

Stimulus: User looks at the top rated questions
Response: User clicks on a question
Stimulus: Answers to those questions appear
Response: User returns to the Question page
Stimulus: User clicks the “Add Question” button
Response: User asks a question that he/she wants an answer for
Stimulus: Pop-up “An email will be sent as soon as someone replies”
Response: User clicks the ‘Return to Course List’ button to view more courses

3.2.3.2. Associated functional requirements

3.2.3.2.1. The system shall allow users to click on previous questions and view its answers

3.2.3.2.2. The system should allow users to rate the question and answers

3.2.3.2.3. The system should allow users to ask a new question

3.3. Performance requirements

The project will support an arbitrary number of users. Interactions with the project will take little time to complete.

3.4. Design constraints

The project will be entirely web-based. This means that all interactions will need to be completed quickly to prevent the site from feeling unresponsive. The site must be very minimalistic and avoid resource-intensive operations.

3.5. Software system attributes

3.5.1. Reliability

Pulling the correct data from the database will make the functioning of the website more reliable at the time of delivery.

3.5.2. Availability

Initially the website will be available to anyone who registers with their email. Users must verify their email address to access the website.

3.5.3. Security

Email verification to verify students. Sanitize inputs to prevent database manipulation.

3.5.4. Maintainability

In the event classes are added or removed by the University, they will need to adjusted manually in the application. This with our MySQL database.

3.5.5. Portability

Very portable since only a modern browser is required, allowing all popular operating systems to access the application.

3.6. Other requirements

No further requirements are necessary.