

BigDaddy.com

Requirements Specifications



BigDaddy

Puthypor Sengkeo

Nick Lamos

Daniel Chia

Course: CptS 322 - Software Engineering Principles I

Instructor: Sakire Arslan Ay

TABLE OF CONTENTS

I.	Introduction.....	3
II.	Requirements Specification.....	3
III.	Customers, Users, and Stakeholders.....	4
IV.	Use Cases.....	4
V.	Non-Functional Requirements.....	7
VI.	User Interface Requirements.....	7

I. Introduction

The school of EECS at WSU has been recruiting undergraduate teaching assistants, who show interest in teaching assistantship, for introductory level courses and lab sections. This process has been carried out mostly manually, which is a very tedious process and time consuming for the recruiters. The aim of this project is to create a web application that allows students to enter their contact information and course preferences for teaching assistantships, and allows instructors to choose their teaching assistants among the students who showed interest in their courses.

Section II includes Requirements Specification

Section III includes Customer, Users, and Stakeholders

Section IV includes Use Cases

Section V includes Non-Functional Requirements

Section VI includes User Interfaces Requirements

Section VII includes References

Document Revision History

Rev 0.0 2019-10-07 Initial Version

II. Requirements Specification

The application will have two pages; a “student page” and an “instructor page”.

On the student page, a student user can:

1. Create a student account and set the account password (user name should be the WSU email)
2. Login with username and password
3. Enter contact information (name, last name, WSU ID, email, phone)
4. Enter additional information (major, cumulative GPA, expected graduation date, whether s/he served as a TA before)
5. Enter course preferences for the TAship. A student can specify more than one course for her/his preferences. For each course, the student should enter:
 - a. The course number (i.e. CptS 121)
 - b. The grade s/he earned when s/he took the course
 - c. The year and semester s/he took the course
 - d. The year and semester s/he applies for TAship
 - e. Whether s/he served as a TA for this course before (yes/no)

Student users can add/remove courses to/from their list of preferred courses

6. Save the entered information

On the instructor page, a faculty user can:

1. Create an instructor account and set the account password (user name should be the WSU email)
2. Login with username and password
3. Enter contact information (name, last name, WSU ID, email, phone)
4. Enter the courses s/he teaches currently. Of course an instructor can teach more than one course and/or lab section during a semester. (If a particular course has more than one lab

section, each of those sections should be entered as a separate course. For example:
CptS121-Lab1, CptS121-Lab2...etc.)

5. See the list of students who applied for TAship. The list should show whether the students have already assigned to a course.
6. The instructor can select a student (who is not assigned to a course yet) and assign it to one of the courses/labs s/he teaches. S/he may also remove the current TA assignments for his courses/labs.

III. Customer, Users, and Stakeholders

Regular User: Is responsible for applying to Teaching Assistant positions.

- a. Search for class or lab
- b. Apply for Teaching Assistant (TA) position
- c. Accept/Decline offer for TA position
- d. View class schedule, syllabus and other information

Administrator: Is responsible for assigning and approving a Teaching Assistant to a class or lab.

- a. Create open Teaching Assistant position for class/lab
- b. Review applications
- c. Approve or Deny application
- d. Upload class schedule, syllabus or other documents

IV. Use Cases

Use case # 1

Name	Create open Teaching Assistant (TA) position
Users	Administrator
Rationale	To allow Administrator to communicate to regular users what TA positons are available
Triggers	User creates TA positons
Preconditions	Class/lab exists
Actions	System will prompt the user to input options for fields such as GPA requirements or previous class grades Form box will be created to display open position and allow regular users to

	apply for TA position
Postconditions	Administrator interface will display open TA position and list of applicants Regular User interface will display form box to apply for open position as well as a list of requirements
Acceptance Test	Form box with administrator input details exists after creation Regular user interface displays form box
Iteration	1st Iteration

Use case # 2

Name	Approve or Deny application
Users	Administrator
Rationale	Allows administrator to choose and assign TAs for a class/lab
Triggers	Administrator selects a class/lab details
Preconditions	Class/lab exists and at least one regular user has submitted an application
Actions	System will display a list of applicants and give the option to approve or deny an applicant System will update application status from pending to either approved or denied on the regular user interface
Postconditions	Administrator interface will display list of approved TAs Regular User interface will display either approved or denied application
Acceptance Test	System updates both the administrator and regular user interfaces
Iteration	1st Iteration

Use case # 3

Name	Accept/Decline offer for TA position
Users	Regular user
Rationale	To allow regular user to accept or decline an offer for TA position

Triggers	Administrator approves or denies TA application
Preconditions	Class/lab exists, administrator has approved or denied application for TA position
Actions	System will register regular user input store it and update the class/lab to either show a user that has accepted a position or remove a user from the class/lab if the user declined the position
Postconditions	Administrator interface will display list of regular users that have accepted the TA position Regular User interface will display confirmation of selection (accept/decline)
Acceptance Test	System outputs correct recorded data (determined by what the regular user inputs (accept/decline offer))
Iteration	2nd Iteration

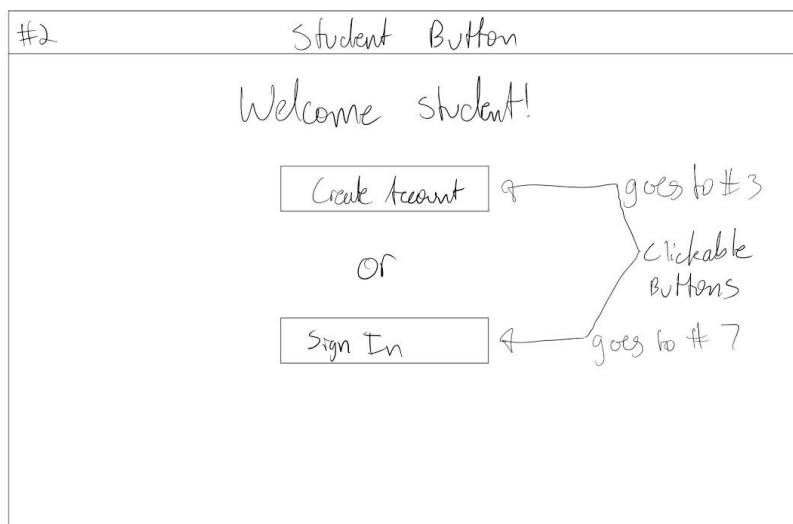
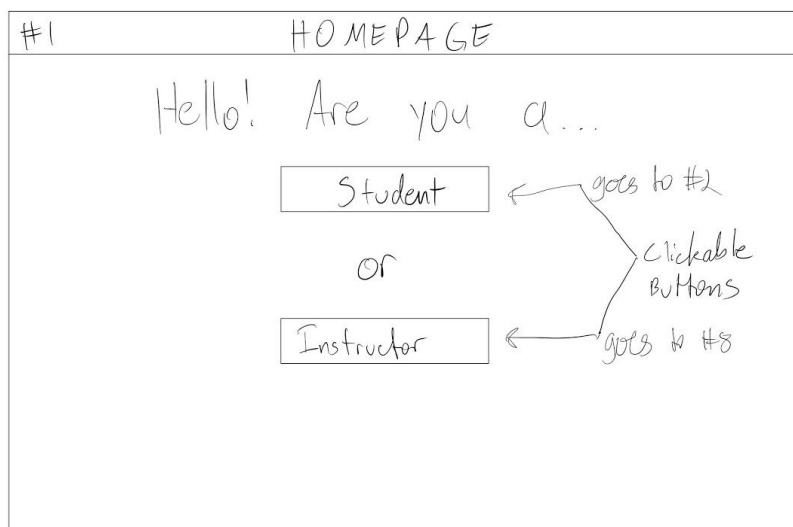
Use case # 4

Name	Upload Documents
Users	Administrator and Regular user
Rationale	To allow administrator and/or regular user to upload and/or view documents related to a specific class such as syllabus, schedules etc
Triggers	Administrator/Regular user selects a class/lab
Preconditions	Class/lab exists, regular user has access to selected class/lab
Actions	System will prompt to choose documents to be uploaded System will store documents in database System will display a list of available documents for download
Postconditions	Administrator interface will display list of documents Regular User interface will display list of documents
Acceptance Test	System shows documents exists after successful upload
Iteration	2nd Iteration

V. Non-Functional Requirements

1. Performance Requirements: Making sure the site runs fast and smooth with good response and processing time.
2. Interface Requirements: Having all elements of our website easy to find and laid out in a clear manner.
3. Security Requirements: Encrypting username and passwords so an intruder would be unable to get in and expose our customers sensitive data.
4. Maintainability Requirements: Being able to easily upgrade our website and make changes in a reliable way to keep it free from bugs.

VI. User Interface Requirements



#3 Student Create Account Button

First name:

Last name:

WSU ID:

Email:

Phone:

Major:

GPA:

Expected Graduation Date: ← options

Have you been a TA before? : ☐ Yes ☐ No ← check boxes

← button

Text fields

#4 Student Create Account Next Button

Course Preferences ← Title

Course number (ex. CptS 121): ← text field

Grade for the course: ← option

Year and semester the course was taken: ← both options

Year and semester you want to be a TA: ← both options

Have you been a TA for this class before: ☐ Yes ☐ No ← check boxes

↑
button to save current information to user, then clear all fields

↑
button to save current data to user

#5

Student Create Account Save Button

Thank you so much for applying

_____!
(student name)

An instructor will reach out to you shortly.

Go to Account Page

← button
(goes to
#6)

#6

Student Account Page

Welcome back _____!
(student name)

Courses you've applied to:

(list courses here)

Apply for another course?

APPLY

↑
button to take them to
the account creation page
(# 4)

7

Student Sign In Button

Please Sign In!

Email :

Password:

} Text fields

Sign In

↖ button (goes to #6)

8

Instructor Button

Welcome instructor!

Create Account

or

Sign In

↖ goes to #9

clickable buttons

↖ goes to #11

#9

Instructor Create Account Button

First name:

Last name:

WSU ID:

Email:

Phone:

} Text fields

& button

#10

Instructor Create Account Next Button

Courses you teach ← Title

Course number (ex. CptS 121): ← text field

Lab number (if applicable) ← text field

↑
button to save current information to
user, then clear all fields

↑
button to save current
data to user (goes
to #12)

#11

Instructor Sign In Button

Please Sign in!

Email: Password:

} Text fields

↔ button (goes to #12)

#12

Instructor Account Page

Welcome back _____!
(Instructor name)

Courses you teach:

(list courses here
with a select
button)Selected course: _____
(course name)

Students applying for TAs

<input type="text"/>	Selected student:
<input type="text"/>	↳ list all info about student (Major, GPA, etc.)
<input type="text"/>	Confirm _____ (student name)
(list of students w/ select button)	to be your TA? <input type="checkbox"/> Yes <input type="checkbox"/> No ← <small>(go to next check boxes)</small>

#13

Instructor Student Chosen Page

Great! You have chosen _____
(student name)
to be your TA for _____
(class)

UNDO

↑
button to remove
student from TA
(goes back to #12)

SAVE

↑
button to save
student as TA
(goes back to #12)