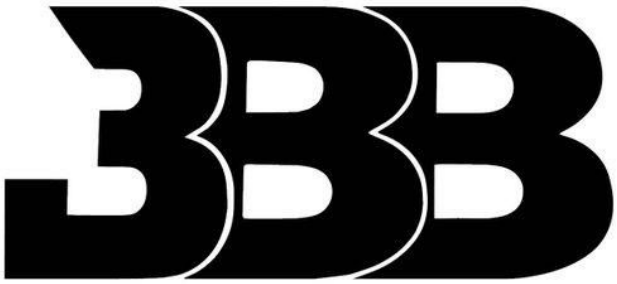
TA Helper

Requirements Specifications



Big Bug Busters

Nolen Young

Ben Matthews

Zack Snoen

Course: CptS 322 - Software Engineering Principles I

Instructor: Sakire Arslan Ay

TABLE OF CONTENTS

I. Introduction

Every semester, engineers at WSU manually pick TAs for their courses and labs which we can make easier for them. This task is not trivial and will save man hours and therefore, money for the school. We will solve this problem through a web-based application where students who are interested in becoming a TA fill out a survey to specify their course preferences and enter contact information. The faculty will then choose TAs among the students who showed interest in their courses.

The objectives of this project is to create an easy-to-use web app that allows potential TAs and faculty to easily find each other. We will need an easy to understand UI that users can use without getting confused. To accompany this, we will need a strong backend that accurately matches teachers to students in order to be more effective than doing this project manually.

Section II includes descriptions of the features, functions and anything else that is required for our project. Section III includes descriptions of our user interfaces, what we are doing to make it user friendly and easily navigable. We also have mockups for the main parts of our interface. Section IV includes our references.

Document Revision History

Rev 0.1 - 10/3/2018 Read over project details together, put together cover page and started on the Introduction.
Rev 0.2 - 10/5/2018 Finished Sections 1, and 2 as a group, will finish up rough draft this weekend.
Rev 0.3 - 10/7/2018 Finished rough draft, we will go over it tomorrow before turning it in.
Rev 1.0 - 10/8/2018 Completed final draft.

II. Requirements Specification

II.1. Customer, Users, and Stakeholders

Our customer is specifically the eecs department at Washington State University. They are looking for software to easily assign teaching assistants to courses and lab sections. Meanwhile, the users will be those in the eecs department, students and teachers alike. The students are looking for a fast and simple way to sign up to be a teaching assistant for a course they have already taken and excelled in. Teachers in the eecs department are looking for a clean interface showing students with interest in being a teaching assistant in the courses they teach, as well as an easy way to choose students from that list. Our stakeholders are those that benefit from our system, which is everybody in the eecs department and Washington State University staff. This will save them time and provide an easy way to assign those qualified to be teaching assistants.

II.2. Use Cases

The actors involved in this program are both faculty and students. Faculty will need to be easily able to find students to TA for their classes. And likewise students will need to be able easily apply to TA positions.

Use case # 1

Name	Login existing user
Users	All Users(Student and Faculty)
Rationale	Students and faculty should be able to login to their correct page with their correct information stored previously. They should already have registered an account. The user enters their username with their WSU email address and their password the one they created when registering. Once logged in will take to the user, either student, or faculty's correct page.
Triggers	Hitting a 'submit' button on the login page
Preconditions	The user's account is created and the information entered matches one in the system
Actions	1. User enters their email and password to login 2. The system responds by checking the database for a user with the email and password 3. The system goes to either a faculty or student page 4. If the user has entered a email and pw that is not in the system then pops up an error message
Postconditions	The system goes to a page with a student or faculty
Acceptance Tests	Make sure that a user information is created and when
Iteration	1

Use case # 2

Name	Register New User
Users	All
Rationale	New users trying to use the site will need to enter some information in order to be added as a user to the website. On this page we will take information about whether the user is a faculty student, and their username and password.
Triggers	The user hits 'Register or 'Cancel' on the register page.
Preconditions	None
Actions	1. The user can hit a 'submit' button which will add the information provided in the form into the system, and return the user to the login page. 2. The user can hit a 'cancel' button and be taken back to the login page without their information being added into the system. 3. If the user information entered is invalid or already taken, an error message pops up and doesn't enter the user into the system.
Postconditions	The user's information is added into our system and can now login.
Acceptance Tests	All of the form is filled out and all the information given is valid.
Iteration	1

Use case # 3

Name	Save Student Information
Users	Student
Rationale	Once a student logs in to the document their name email will be displayed at the top of the page. Information of name, last name, WSU ID, email, phone. Information of major, cumulative GPA, expected graduation date, previous TA courses taught. The user can also save their course preferences which each course having a course number, grade received, year/semester taken, year/semester applied, and whether student has been a TA before. All of this information will be displayed by default and all editable.
Triggers	The user clicks a save button.
Preconditions	The user information is valid.
Actions	1. The user can edit any information and update course preferences 2. The user hits a 'save' button that updates any information in the system 3. If any input is invalid with wrong characters ect. Then the system will pop up an error message 4. If the input is valid the page will be refreshed and the information that was there when the button was clicked will be displayed
Postconditions	The page is refreshed
Acceptance Tests	Information will be edited and when the submit button is hit, the new information will displayed
Iteration	2

Use case # 4

Name	Save Faculty Information
Users	Faculty
Rationale	Once a faculty logs into the web app the faculty should be able to edit any information linked to their profile. Their information is displayed when the web page is opened, and the user can change and edit that information
Triggers	The user will hit a 'save' button to change the information on their profile
Preconditions	All the information entered into the user profile form is valid.
Actions	1. The user can edit and change any of the information that gets pulled up by the system and displayed at login 2. The user can save any changes they made to their information after they edit it. 3. If they new information added and is invalid then an error message appears on screen
Postconditions	The user information is updated in the system and the page is refreshed.
Acceptance Tests	We check to make sure the information was actually updated in the system. We also make sure that all the information changed is valid.

Iteration	2

Use case # 5

Name	Assigning TAs to classes
Users	Faculty
Rationale	The program needs to be able to allow faculty to choose students to TA for them. This is the main objective of the web app so it is crucial that faculty can easily find students to TA for them.
Triggers	User clicks on 'Manage TAs' button
Preconditions	The faculty must have at least 1 course registered that they are teaching.
Actions	1. The user selects a course to add or remove TAs from. 2. A table of available TA's for that course is loaded and displayed to the user. 3. The user can select from among those TA's with a checkbox. 4. The user can hit an 'add TAs button' to add the student to TA their course. 5. The user can also hit a 'Cancel' button to return to their profile page 6. If the form is invalid an error message appears on screen
Postconditions	The TA profiles are updated to show that they are now TAing for the selected class.
Acceptance Tests	The information is successfully updated and all the information entered is valid.
Iteration	3

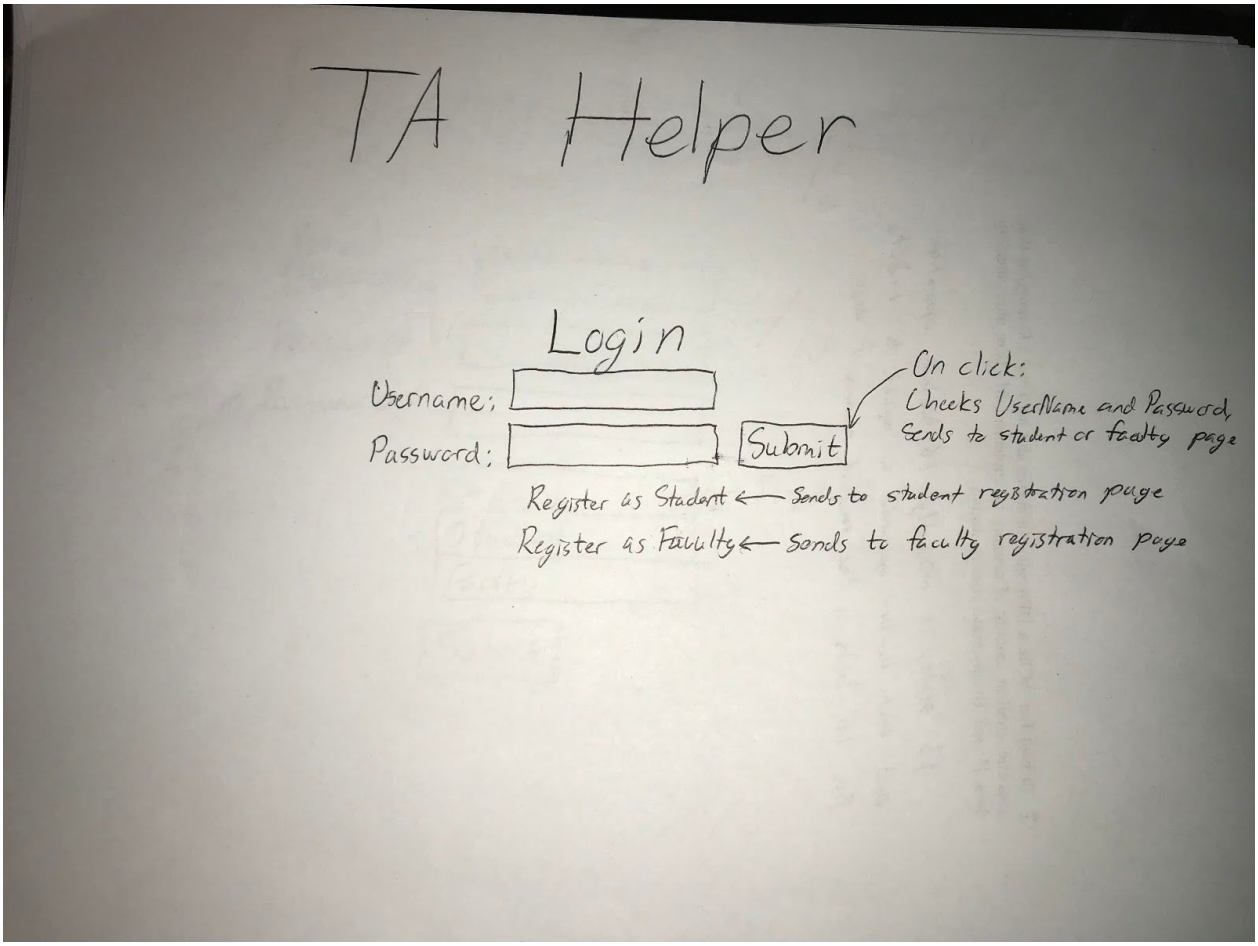
II.3. Non-Functional Requirements

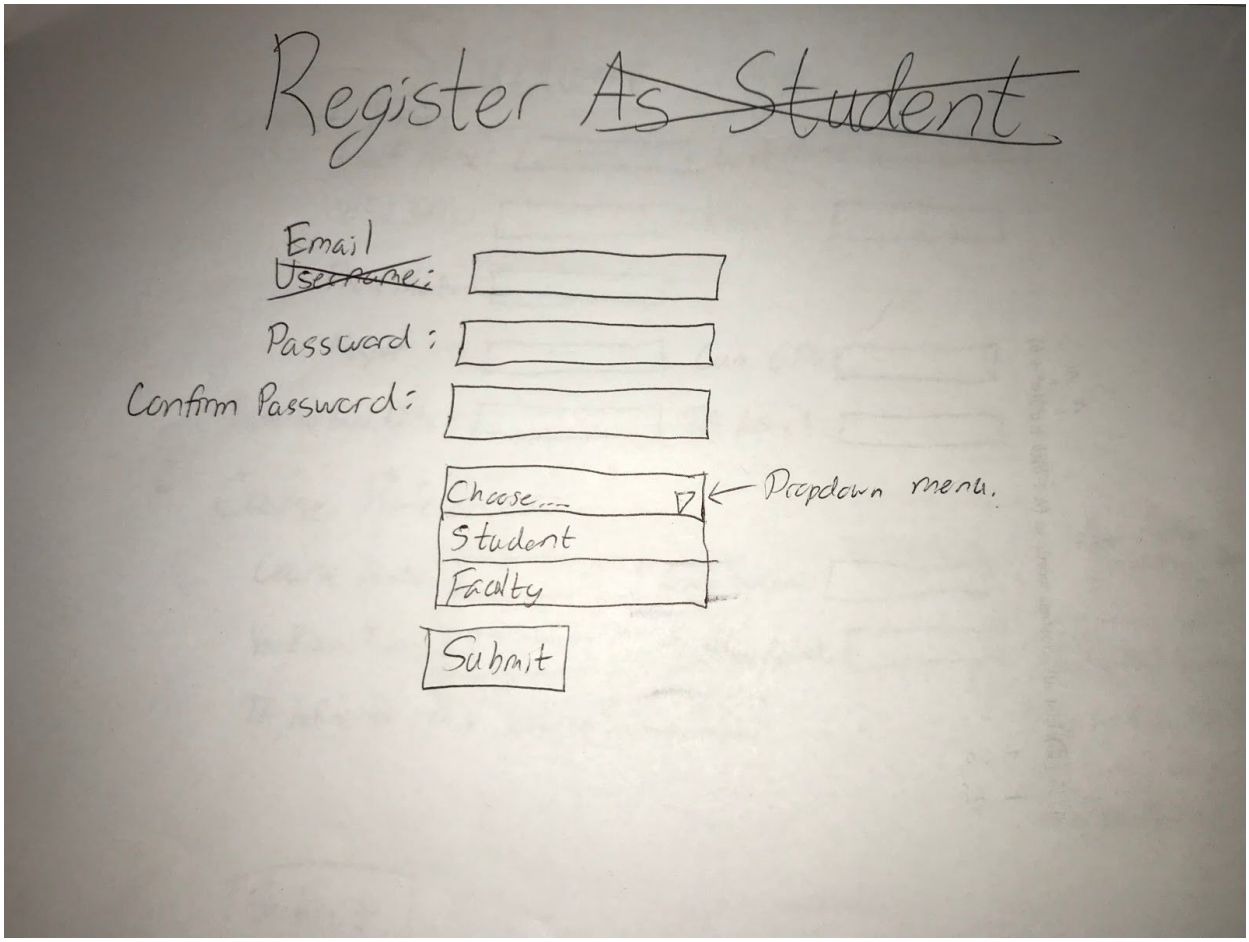
1. Performance: The performance time should be efficient and without unnecessary long response time
2. Security: The application should not allow breaches of data or the mishandling of data to outside users
3. Reusability: A user should be able to have a login for multiple semesters. The user can then update their preferences by logging in and editing. Faculty can do the same with their login.
4. Usability: Interface is easy to use and understand. If there is ever an error, that error should be displaced and the results of the handling displayed.
5. Portability: The web app should function on all major web browsers. IE Chrome, Edge, Safari, Firefox, Opera.
6. Scalability: The system should work quickly with and accurately with a large amount of users.
7. Dependability: The web app should work at all times, and should be resistant to crashing and other errors.

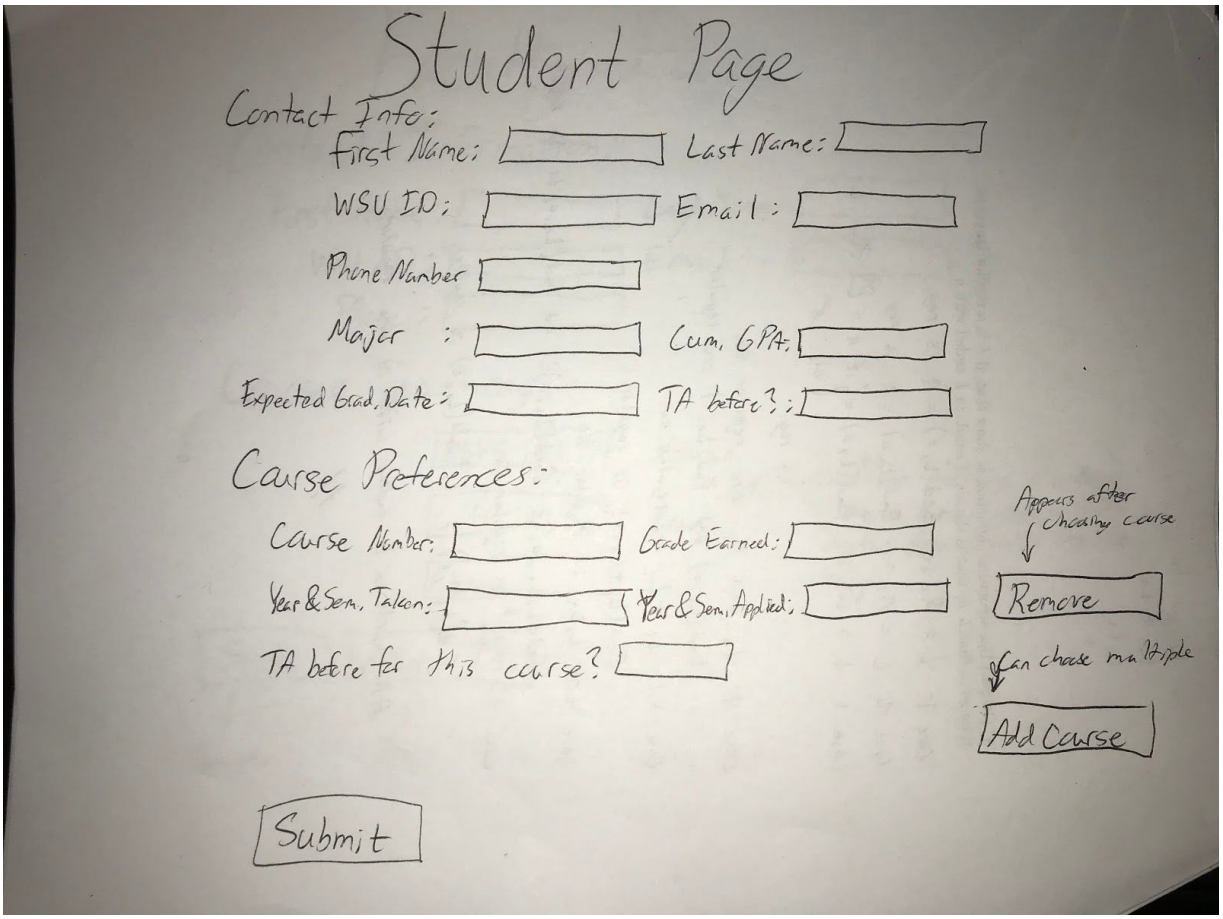
III. User Interface Requirements

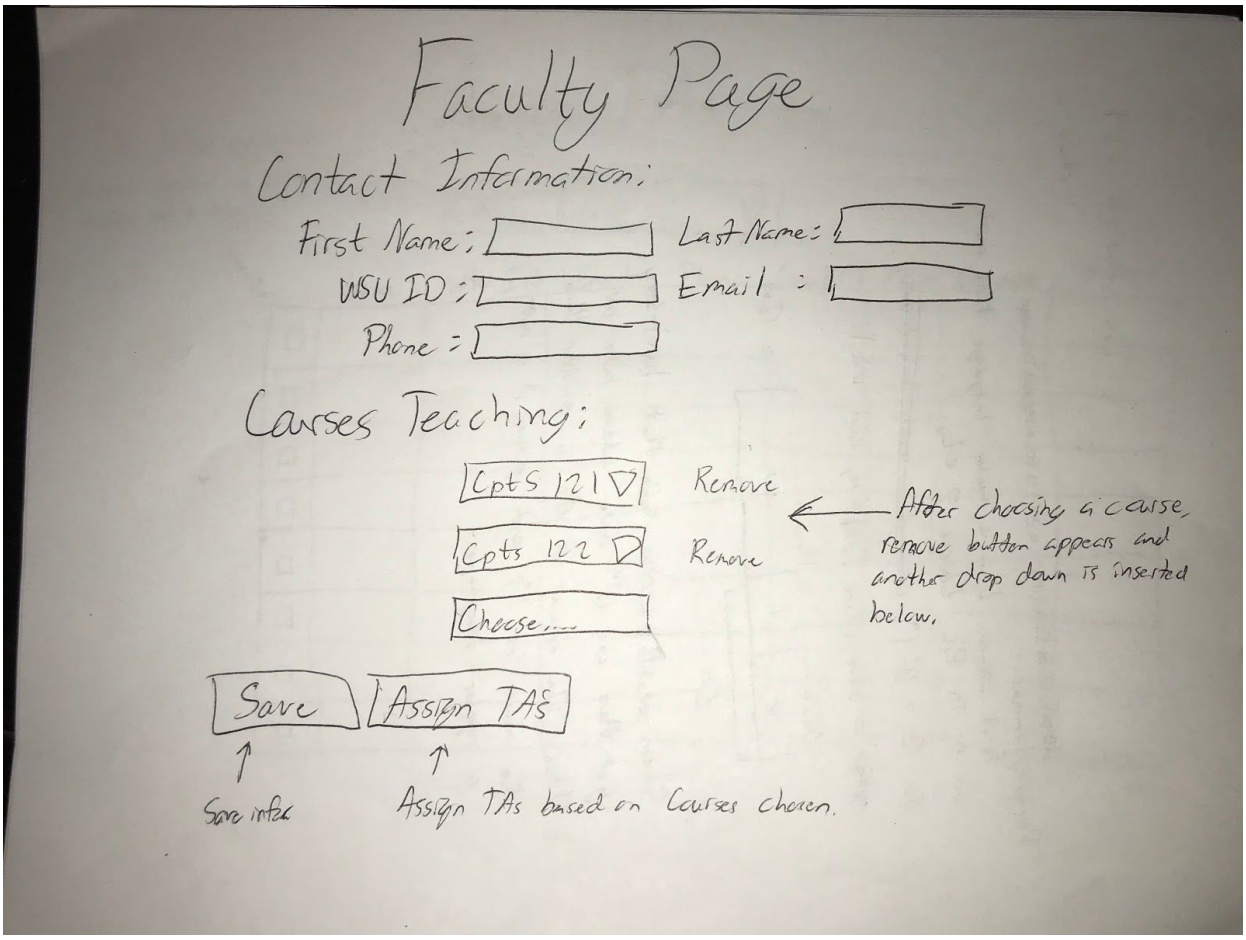
Our users must not struggle to work around our website, it will be simple and clear in leading the user to what they need. For students that is signing up and logging in as a student, as well as registering for a teaching assistant role in a course. Teachers should be able to easily sign up and log in, as well as search through users that have applies to be a teacher assistant in their courses and see their qualifications.

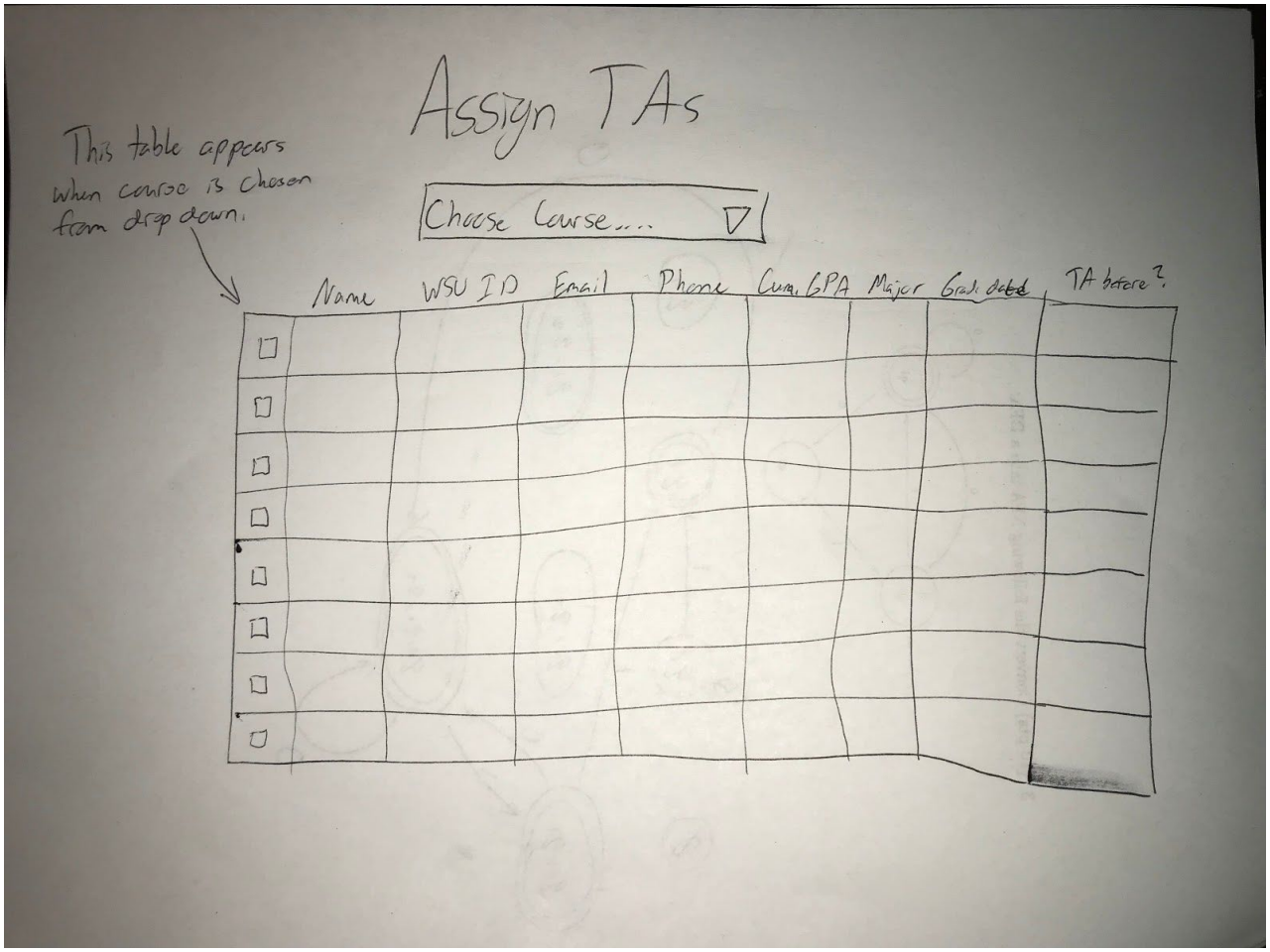
Below are some sketches of our main user interfaces.




Register As Student
Email
Username:
Password:
Confirm Password:
Choose...
Student
Faculty
Dropdown menu.
Submit


Student Page
Contact Info:
First Name:
Last Name:
WSU ID:
Email:
Phone Number
Major :
Cum. GPA:
Expected Grad. Date:
TA before?:
Course Preferences:
Course Number:
Grade Earned:
Year & Sem. Taken:
Year & Sem. Applied:
TA before for this course?
Appears after choosing course
Remove
can choose multiple
Add Course
Submit


Faculty Page
Contact Information:
First Name:
Last Name:
WSU ID:
Email :
Phone :
Courses Teaching:
Cpts 121
Remove
Cpts 122
Remove
Choose....
After choosing a course, remove button appears and another drop down is inserted below.
Save
Assign TAs
Save info
Assign TAs based on Courses chosen.


Assign TAs
This table appears when course is chosen from drop down.
Choose Course....
Name
WSU ID
Email
Phone
Cum. GPA
Major
Grad. debt
TA before?