

<< Software Requirements Specification >>

ResearchNetwork - Web App

Prepared by:

- Boxiang Lin
- Shutian Wang
- Yi Chou
- Wanting Wu

Course: Cpts 322 - Software Engineering Principles I

Instructor: Sakire Arslan Ay

Table of Contents

- [<< Software Requirements Specification >>](#)
 - [## ResearchNetwork - Web App](#)
 - [Table of Contents](#)
 - [Document Revision History](#)
- [1. Introduction](#)
 - [1.1 Document Purpose](#)
 - [1.2 Product Scope](#)
 - [1.3 Document Overview](#)
- [2. Requirements Specification](#)
 - [2.1 Customer, Users, and Stakeholders](#)
 - [2.2 Use Cases](#)
 - [2.3 Non-Functional Requirements](#)
- [3. User Interface](#)
- [4. References](#)
- [Appendix: Grading Rubric](#)

Document Revision History

Name	Date	Changes	Version
Revision 1	2021-10-05	Initial draft	1.0

1. Introduction

ResearchNetwork is a platform that provides smooth connections to research opportunities for students at Washington State University. Professors or faculties can present their research opportunities on this platform. Students will be able to see all currently available research opportunities and be able to apply for them.

1.1 Document Purpose

The purpose of this document is to presents a concise description of the Web System. It will explain the features and the interface of the system. This document intends to build an online system to manage the undergraduate research opportunities between the faculty's and students to expose the opportunities to all users.

1.2 Product Scope

Research Network is a platform that provides services for WSU students and faculty. Our purpose is to link students with faculty. On this platform, students can easily find research positions with their interests. WSU EECS faculty can advertise their research opportunities, review student qualifications and select candidates that they would like to interview with the position.

1.3 Document Overview

The remaining sections of this document provide a general description, including characteristics of the users and the functional requirements of the system. The specific software requirements of the project is discussed in section 2 of this document, including the Customer, Users and Stakeholders, Use cases, and non-functional requirements. Section 3 provides the sketches or mock-ups for the main parts of the interface. Section 4 is the References page for supporting information.

2. Requirements Specification

This section specifies the software product's requirements. Specify all of the software requirements to a level of detail sufficient to enable designers to design a software system to satisfy those requirements, and to enable testers to test that the software system satisfies those requirements.

2.1 Customer, Users, and Stakeholders

The customer is specifically the faculty members for the EECS department at WSU(Research Assistant, Professors, etc). They are looking for a online platform that will allow the faculty advertise their research position and connect them with the qualified undergraduate students. So the user will be the students and the faculty members, after students enter their contact information, completed coursework, research interests, and prior research experience. They can apply for the research position. For faculty members, they can advertise research opportunities for undergraduate students and select the candidates that they would like to interview with for the position. The stakeholders are those that benefit from our platform witch is every faculty members in the EECS department and also the students who use this platform. For faculty members, they have more opportunities advertise their research positions and faster way to connect them with the qualified undergraduate students. For students, they have more chance the get into the research they are interested and participating the research earlier.

2.2 Use Cases

Use case # 1	
Name	Create a Faculty Account and Enter Profile Information
Users	Faculty User
Rationale	The Users have the access to create account in faculty page by filling the required profile information. The Create a Faculty Account and Enter Profile Information should allows Users to create a login and become a Registered User.
Triggers	The faculty user select the "Create Account" option
Preconditions	The form and the model of the database have been successfully created
Actions	<ol style="list-style-type: none">1. The Users enter the required Faculty Account information values and requests that the system saves the entered values.2. The system validates the entered Faculty Account information and Profile Information3. The entered data for the Users information are stored in the User's account.4. The system notifies the User that the account has been created
Alternative paths	<ol style="list-style-type: none">1. The User can choose to cancel create the account at any time.2. In step 2, the system describes which entered data was invalid and presents the User with suggestions for entering valid data.3. In step 2, the User re-enters the information and the system re-validates it.4. In step 2, if the invalid values is entered, the entered information is invalid alternative flow is executed again. This continues until the users chose to cancel create the account.
Postconditions	The User entered data is stored in the user account.
Acceptance tests	Make sure that all the User entered data is valid and successfully stored in the system.
Iteration	#1

Use case # 2	
Name	Faculty Login with Username and Password
Users	Faculty User
Rationale	The Users have the access to login account in faculty page. The Login with Username and Password allows the Users login to his/her account page and have the access to the features.
Triggers	The faculty user select the "Login" option
Preconditions	A faculty account has been created and stored in system
Actions	<ol style="list-style-type: none"> 1. The system prompts the User for his/her username abd password. 2. The User enters his/her username and password. 3. The system validates the entered information.
Alternative paths	<ol style="list-style-type: none"> 1. In step 3, inform the Users if the entered information is incorrect. 2. In step 3, if the Users does not have an account, the system will inform the User to register an account.
Postconditions	<ol style="list-style-type: none"> 1. The User is successfully log in and the system display all features available for the role the user is associated with. 2. The User fails to log in and was informed the invalid sign in information.
Acceptance tests	Make sure that all the valid User Accounts can log in successfully.
Iteration	#1

Use case # 3	
Name	Create Research Positions
Users	Faculty User
Rationale	The Users will be able to create the undergraduate research positions. The Create Research Positions allows the Users to enter and post the details of the position and qualifications needed.
Triggers	The User select the "Create Research Positions" option in the faculty page
Preconditions	The User is signed in and user's profile exists
Actions	<ol style="list-style-type: none"> 1. The Users enter the details of the position and qualifications for the research position and requests that the system saves entered values. 2. The edited post from the User is display in student page and under the User's account.
Alternative paths	<ol style="list-style-type: none"> 1. The User can choose to cancel create the post at any time. 2. In step 1, the system informs the User if any field is missing and presents the User with suggestions.
Postconditions	The User successfully created a new post including the details of the position and the system displays a new post.
Acceptance tests	Make sure that a new post is added to the database
Iteration	#1

Use case # 4	
Name	Delete the Existing Research Positions
Users	Faculty User
Rationale	The Users will be able to delete the posted position under his/her account in any time. The Delete the Existing Research Positions allows the Users to remove the existing post.
Triggers	The faculty user select "Delete Research Positions" option
Preconditions	A research position has been created and posted
Actions	<ol style="list-style-type: none"> 1. The Users delete the posted research positions. 2. The deleted post from the User is removed from the student page. 3. Once deleted, all applications to the position receive updates as "Position is not available"
Alternative paths	In step 1, the Users can only delete his/her own post.
Postconditions	The User successfully delete the post and the post is removed from the page. All applications to the deleted position should update as "Position is not available"
Acceptance tests	Make sure that the post is deleted from the database
Iteration	#1

Use case # 5	
Name	View Open Research Positions
Users	Student, Faculty
Rationale	The open research positions are what users of this platform are looking for. So when students or faculty log in or click their home page, they should see a list of open research positions.
Triggers	The user click the Home Page (index page) or redirect to the Home Page route.
Preconditions	Student or Faculty account login
Actions	<ol style="list-style-type: none"> 1. Home page (index page) clicked. 2. Home page (index page) route directed 3. Database table query 4. HTML render the database information (Only render the Research Project Title, brief description of the required qualifications, research filed, and start and end date)
Alternative paths	For step 1, when users login their account, the platform will automatically load the Home page (index page).
Postconditions	All open research opportunities are shown on the Homepage (index page).
Acceptance tests	Make sure total number of open research opportunities are shown on the home page.
Iteration	#2

Use case # 6	
Name	Recommended Research Positions
Users	Student
Rationale	Now students that use this platform are in different majors, and among a major, they could be interested in various fields of studies. An algorithm that filters out the matching positions to the student will significantly improve the research job look-up efficiencies for the students.
Triggers	The user submit the select field form component.
Preconditions	Student account login and is on the Home Page route.
Actions	<ol style="list-style-type: none"> 1. Select field clicked. 2. Home page (index page) route directed and select filed element retrieved 3. Database table query the corresponding positions with match tag 4. HTML render the database information
Alternative paths	For step 1, the select field will always place the "All" at the first position, so it will always load the all open research position. Unless otherwise, other interest tags are selected.
Postconditions	The corresponding matched research opportunities are shown on the Homepage (index page).
Acceptance tests	Make sure total number of matched open research opportunities are shown on the home page.
Iteration	#2

Use case # 7	
Name	Open Research Position Information Display
Users	Student, Faculty
Rationale	Since users can see all the open research positions on the Home Page, displaying all information of each available position on the home page will occupy a large pixels amount of space. So, we will need a unique route that to display the information of that open position.
Triggers	The user click a specific research position post.
Preconditions	Home page is loaded.
Actions	<ol style="list-style-type: none"> 1. Click on a specific research position post. 2. information route directed 3. Database information query and pass to HTML 4. HTML render all information for the research position post)
Alternative paths	A faculty user can only display other research position posts posted by other faculty, for the post from themselves will be handled with additional functionality that provide for modifications.
Postconditions	A new page that shows all the details about the research position that user clicked.
Acceptance tests	Make sure all the information about the current research position is on the page.
Iteration	#2

Use case # 8	
Name	Student Account Registration
Users	Anonymous User / Student
Rationale	We want students can be able to log in to this platform, view and apply for the research position. To ensure that the student account has all the necessary information, we will require students to fill out their profiles at the time of registration.
Triggers	User select create student account
Preconditions	A user has a valid wsu@edu email that is not already tied to an account in the application.
Actions	<ol style="list-style-type: none"> 1. Click register for a student account. 2. System direct to a registration route and display the form for fill out. 3. The student user supplies information (email, username, password) and profile information to the system. 4. The system validates the supplied email is a valid and unused email and username. 5. The student user is directed into the login page
Alternative paths	<ol style="list-style-type: none"> 1. A user is asked to enter a different email if the email is already associated with an account or the email is not from the wsu.edu domain. 2. A user is asked to enter a different username if the username is already associated with an account.
Postconditions	An account for the email supplied by the user has been created, storing email, username, password, and profile information.
Acceptance tests	Make sure that the user account is stored by the system with identical information to what the user supplied.
Iteration	#2

Use case # 9	
Name	Student Account Login
Users	Student User
Rationale	For student to be able to access features that require a student user account
Triggers	A user selects the student login page
Preconditions	A student user already has a created account
Actions	<ol style="list-style-type: none"> 1. A student user should be able to supply the information (email and password) that is associated with their account. 2. The system will validate that the information given by the user is correct. 3. The user will now have a validated login session created. 4. The user will be directed to the home page.
Alternative paths	<ol style="list-style-type: none"> 1. A student user is asked if they want to sign up for an account if an account does not exist with their email. 2. A student user is asked to re-enter the password if the one entered is incorrect.
Postconditions	A student user successfully logs into their account and a login session is created.
Acceptance tests	Make sure that student users have an established login session with validated student user capabilities.
Iteration	#2

Use case # 10	
Name	Apply research position
Users	Student members
Rationale	students can apply for more than one research position. But colleges need to know why this student wants to apply for this research position. So students need to write a brief statement describing why they are interested in this research topic and what they want to gain after participating in this project. They also need to provide the name and the email of one faculty who can provide them a reference for the position.
Triggers	when students select “Apply research position” option
Preconditions	need to login in to the student account
Actions	<ol style="list-style-type: none"> 1. A student user should be able to supply the information (email and password) that is associated with their account. 2. The system will validate that the information given by the user is correct. 3. The user will now have a validated login session created. 4. The user will be directed to the home page.
Alternative paths	<ol style="list-style-type: none"> 1. If faculty name and email is not from wsu.edu then ask the user to enter a different email and name. 2. The statement cannot be empty and the number of words cannot be less than 200 or 300.
Postconditions	The statement, name and email of a faculty member are stored into the database.
Acceptance tests	Make sure the data students enter is stored into the database and faculty members can also see the data that students enter.
Iteration	#2

Use case # 11	
Name	View the research positions statuses
Users	Students, faculty members
Rationale	When the student submits their application, the statuses should show "pending". After a faculty accepts this application, the status should be updated as "Approved for Interview". And when the statuses are changed students can contact faculty for the Interview. Depending on the Interview result the statuses should change to the "Hired" or "Not hired".
Triggers	When the students submit their application and select "View the research position statuses" option
Preconditions	Need to login to the student account and need to submit the application.
Actions	<ol style="list-style-type: none"> 1. After students submit the application the statuses should be "" Pending". 2. If faculty accepts this application, the status should be updated as "Approved for Interview" 3. After the interview depending on the result faculty can change statuses to "Hired" or "Not hired".
Alternative paths	<ol style="list-style-type: none"> 1. Students and faculty both can see the statuses, but only faculty can change the statuses and students can see the change immediately. 2. For step1, If a student did not submit the application, this page should show "You did not submit the application yet!" or do not show this button before students submit the application.
Postconditions	When the faculty members change the statuses.
Acceptance tests	Make sure that the students and faculty statuses data are synchronous.
Iteration	#3

Use case # 12	
Name	Withdraw pending applications
Users	Students
Rationale	Students can withdraw their pending application. But if the statuses change to the “Approved for Interview”, “Hired” or “Not hired”. Students cannot withdraw anymore.
Triggers	select “Withdraw pending application” option and this option will only show for the applications that are still pending.
Preconditions	Need to login to the student account and need to submit the application. And the application should still be in the pending statuses.
Actions	<ol style="list-style-type: none"> 1. If students press this button it will delete their application data(only for the application they choose) from the database. 2.faculty shouldn't see the data after students withdraw their application.
Alternative paths	For step1, this button will show after you submit the application but when the statuses of your application change to the “Approved for Interview”, “Hired” or “Not hired”, this button will disappear.
Postconditions	When students click the “withdraw” button.
Acceptance tests	Make sure after students press this button the data should be deleted from the database.
Iteration	#3

Use case # 13	
Name	Show student applied on all position
Users	Student, Faculty
Rationale	Faculties use this page to review undetermined students list. Student who has been approved for interview or hired by another position should not be displayed.
Triggers	Faculties click the Responded Applicants button on nav bar or redirect to the Responded Applicants (this) page.
Preconditions	Faculty login; Position exist;
Actions	<ol style="list-style-type: none"> 1. Responded Applicants button clicked, or this page has been redirected. 2. Check if the faculty credentialed. 3. Fetch student list in database.
Alternative paths	In step 2, If the faculty not credentialed, redirect to the login page and notice the faculty.
Postconditions	All student applied this position who not be marked as approved for interview or hired by another position are shown on Responded Applicants (this) page.
Acceptance tests	<p>Make sure student shows on this page same as the student applied.</p> <p>Student approved for interview or hired by another position should not be on this list.</p>
Iteration	#3

Use case # 14	
Name	View student qualification
Users	Student, Faculty
Rationale	Faculties use this page to review student's qualification (like GPA/Course taken/Interested topic/Programming language) who applied to the designated position.
Triggers	Faculties click Details button on Responded Applicants page or redirect to the Applicant Details (this) page.
Preconditions	Faculty login; Position exist;
Actions	<ol style="list-style-type: none"> 1. Responded Applicants button clicked, or this page has been redirected. 2. Check if the position/student exist, is the student applied this position and faculty credentialed. 3. Fetch student information.
Alternative paths	In step 2, if the student does not exist or student did not apply this position, redirect to the Responded Applicants page and notice the faculty. If the position does not exist, redirect to the index page and notice the faculty. If faculty not credentialed, redirect to the login page and notice the faculty.
Postconditions	All student information details (like GPA/Course taken/Interested topic/Programming language) should be shown same Applicant Details (this) page.
Acceptance tests	Make sure this page can fetch correct student information and matched exactly.
Iteration	#3

Use case # 15	
Name	Faculty approve application
Users	Student, Faculty
Rationale	Faculty approve student who applied this designated position.
Triggers	Faculty click approve button.
Preconditions	Faculty login; Student applied; Student exist; Position exist
Actions	<ol style="list-style-type: none"> 1. Faculty click the button Approve for interview. 2. Check if the position and student exist, student applied this position, faculty credentialed. 3. Change student status to approved for interview. 4. Notice faculty database changed successfully!
Alternative paths	In step 2, if student does not exist or student did not apply this position, redirect to the view Responded Applicants page and notice the faculty. If position does not exist, redirect to the index page and notice the faculty. If faculty not credentialed, redirect to the login page and notice the faculty.
Postconditions	Student status changed to approved for interview if faculty clicked the approve button.
Acceptance tests	Make sure the faculty press the Approve for interview button, the student status changed to approved for interview.
Iteration	#3

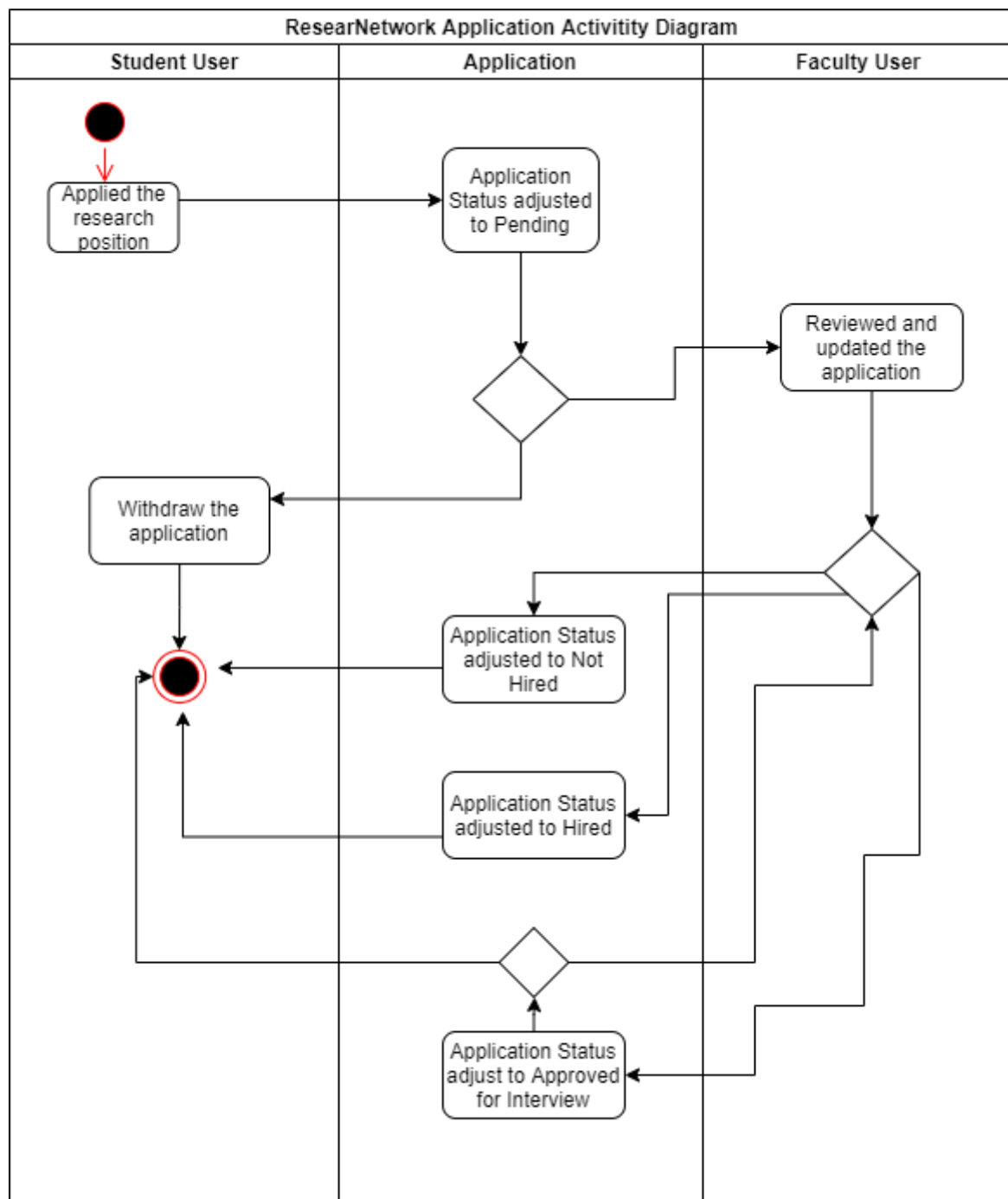
Use case # 16	
Name	Hire or not hire
Users	Student, Faculty
Rationale	Faculty decide hire or not hire student.
Triggers	Faculty click the hire/not hire button.
Preconditions	Faculty login; Student approved for interview; Student exist; Student applied; Position exist
Actions	<ol style="list-style-type: none"> 1. Faculty click the button hire/not hire 2. Check if the position and student exist, student applied this position, faculty credentialed. 3. If hire, change the student status to Hired. 4. If not hire, release the student status. 5. Notice faculty database changed successfully!
Alternative paths	In step 2, if student does not exist or the student did not apply this position, redirect to the view Responded Applicants page and notice the faculty. If position does not exist, redirect to the index page and notice the faculty. If the faculty not credentialed, redirect to the login page and notice the faculty.
Postconditions	Student status changed to hire/normal if faculty clicked the approve button.
Acceptance tests	Make sure the faculty press the hire/not hire button and the student status changed to hire/normal.
Iteration	#3

Include a swim-lane diagram that illustrates the message flow and activities for following scenario:

"A student applies to a research position; initially its status will appear as "Pending". The faculty who created that position reviews the application and updates the application status to either "Approved for Interview", or "Hired", or "Not hired". The updated status of the application is displayed on the student view.

The student may delete the pending applications (i.e., whose status is still "Pending".)"

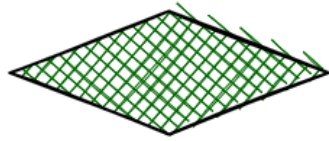


Swim-lane Diagram for application status



2.3 Non-Functional Requirements

1. Security : This platform should not allow breaches of data to the outside user.
2. Efficiency : the performance time should be efficient and without unnecessary long response time.
3. Reusability : students can update their information by logging and edit it. Faculty members can also do that by login their account.
4. Portability: This platform should function on all major web browsers.
5. Stability : This platform should work all times and should not jump out any bug or crashing.
6. Usability : The interface of the platform is easy to learn and the platform is efficient for the frequent user. No matter students or faculty members the platform will satisfied with the platform.

3. User Interface

Homepage	Current User:	
<div data-bbox="325 315 655 454"></div> <div data-bbox="240 454 769 553">ResearchNetwork @ Washington State University</div>		<div data-bbox="1066 163 1134 241"></div> <div data-bbox="1134 179 1394 226">Student Login</div> <div data-bbox="1066 250 1134 329"></div> <div data-bbox="1134 275 1394 320">Faculty Login</div> <div data-bbox="1074 353 1177 383"><p>New User?</p></div> <div data-bbox="1074 387 1378 439"><ul style="list-style-type: none">• Click here to register as a student• Click here to register as a faculty</div>

Student Login

Username:

Password:



Login

No student account? [Click here to register](#)

Faculty Login

Username:

Password:



Login

No faculty account, [click here to register](#)

Student Registration

Username:

Expect Gradulation:

Research Experience

WSU ID:

GPA:

Email:

Technical Elective Courses:

First Name:

Last Name

Major:

Field of Interest

Phone Number

Programming Language



Register

Faculty Registration

Username:

WSU ID:

Email:

First Name:

Last Name

Phone Number



Register

Homepage

Your Applications

Login as: Name (Student)

Log Out

Recommand Positions (Select your interested field):

Research Position XXX

- Brif description

- start and end date

- Fields

Apply

Research Position XXX

- Brif description

- start and end date

- Fields

Apply

Research Position XXX

- Brif description

- start and end date

- Fields

Apply

Research Position XXX

- Brif description

- start and end date

- Fields

Apply

Research Position XXX

- Brif description

- start and end date

- Fields

Apply

Research Position XXX

- Brif description

- start and end date

- Fields

Apply

Research Position XXX

- Brif description

- start and end date

- Fields

Apply

Research Position XXX

- Brif description

- start and end date

- Fields

Apply

Homepage	Post New Position	Responded Applicants	Login as: Name (Faculty)	Log Out
----------	-------------------	----------------------	--------------------------	---------

Research Position XXX

- Brif description

- start and end date

- Fields

Modify

Research Position XXX

- Brif description

- start and end date

- Fields

Research Position XXX

- Brif description

- start and end date

- Fields

Modify

Research Position XXX

- Brif description

- start and end date

- Fields

Research Position XXX

- Brif description

- start and end date

- Fields

Research Position XXX

- Brif description

- start and end date

- Fields

Research Position XXX

- Brif description

- start and end date

- Fields

Modify

Research Position XXX

- Brif description

- start and end date

- Fields

Research Position XXX

- Brif description

- start and end date

- Fields

Homepage	Post New Position	Responded Applicants	Login as: Name (Faculty)	Log Out
----------	-------------------	----------------------	--------------------------	---------

Research Project Title

Required Time Commitment

Project Brif Description:

Research Filed

Applicant Qualification

Start date and end date

POST

Homepage	Post New Position	Responded Applicants	Login as: Name (Faculty)	Log Out
<div><div><div>John John</div><div>AppliedResearch ABC</div><div>Detail</div></div><div><div>John John</div><div>AppliedResearch ABC</div><div>Detail</div></div><div><div>John John</div><div>AppliedResearch ABC</div><div>Detail</div></div><div><div>John John</div><div>AppliedResearch ABC</div><div>Detail</div></div><div><div>John John</div><div>AppliedResearch ABC</div><div>Detail</div></div><div><div>John John</div><div>AppliedResearch ABC</div><div>Detail</div></div></div>				

4. References

Cite your references here.

https://en.wikipedia.org/wiki/Swim_lane