Software Engineering
6905

Graduate Recruitment System

By (Group 3):

- Dalia Ibrahim
- Yamini Kosaraju
- Hailong Feng
- Xiaolong Liu

Contents

Graduate Recruitment System

1. Introduction

Graduate Recruitment (GradRec) system is a web based project. It is developed to recruit the graduate students for the suitable project with required supervisor (project faculty). The projects are administered by a project manager.

2. Current system

The existing system is manually. The students contact the university and ask for available positions. The secretary gives them brochure with projects which updated annually.

3. Proposed system

3.1. Overview

The proposed system is allowing students to sign up and fill their profile accounts with (education, research interests, work experience, intended start date, need for financial support, etc.). The system matching the students based on their profile with the available projects. The projects are managed by a project manager who can add or modify project details like (Project title, description, project participants etc.). Moreover, he sets the matching criteria which will be used in matching step.

The student can communicate with project manager of respective projects for more information and apply to a project.

The project manager can accept this student or decline. If project manager accepts the student, he will send the admission offer and the student may accept/decline the offer.

If the student is rejected or declines the admission, then the student can apply to a new project.

3.2. Functional requirements

- Student:
  - Student creates and fills his profile.
  - The list of matched projects appears to student only after submitting his profile
  - Communication channel between student and project manager is established related to the projects. The student can communicate for more information.
  - Student can apply to only one project at a time.
  - Student makes the decision to accept or decline the offer letter.
  - If he declines the offer he can apply for a new project.
- Project Manager:
  - Project manager adds new projects and provides their description. Also he sets the matching criteria for those projects.
  - The system will notify the project manager if any student is matching with the existing projects and he can proactive and send an offer to this student.
  - Project manager makes assessment and sends the admission offer letter to any students apply for specific projects
  - A project manager can manage multiple projects and view all matched, applied and enrolled students.

3.3. Nonfunctional requirements

- The response time to show the matched project list to student within 2 sec.
- Multiple students (up to 1000) can access the online system at same time.
- The webpages have to be responsive.
- The online system has to be available 24 hours and 7 days a week.

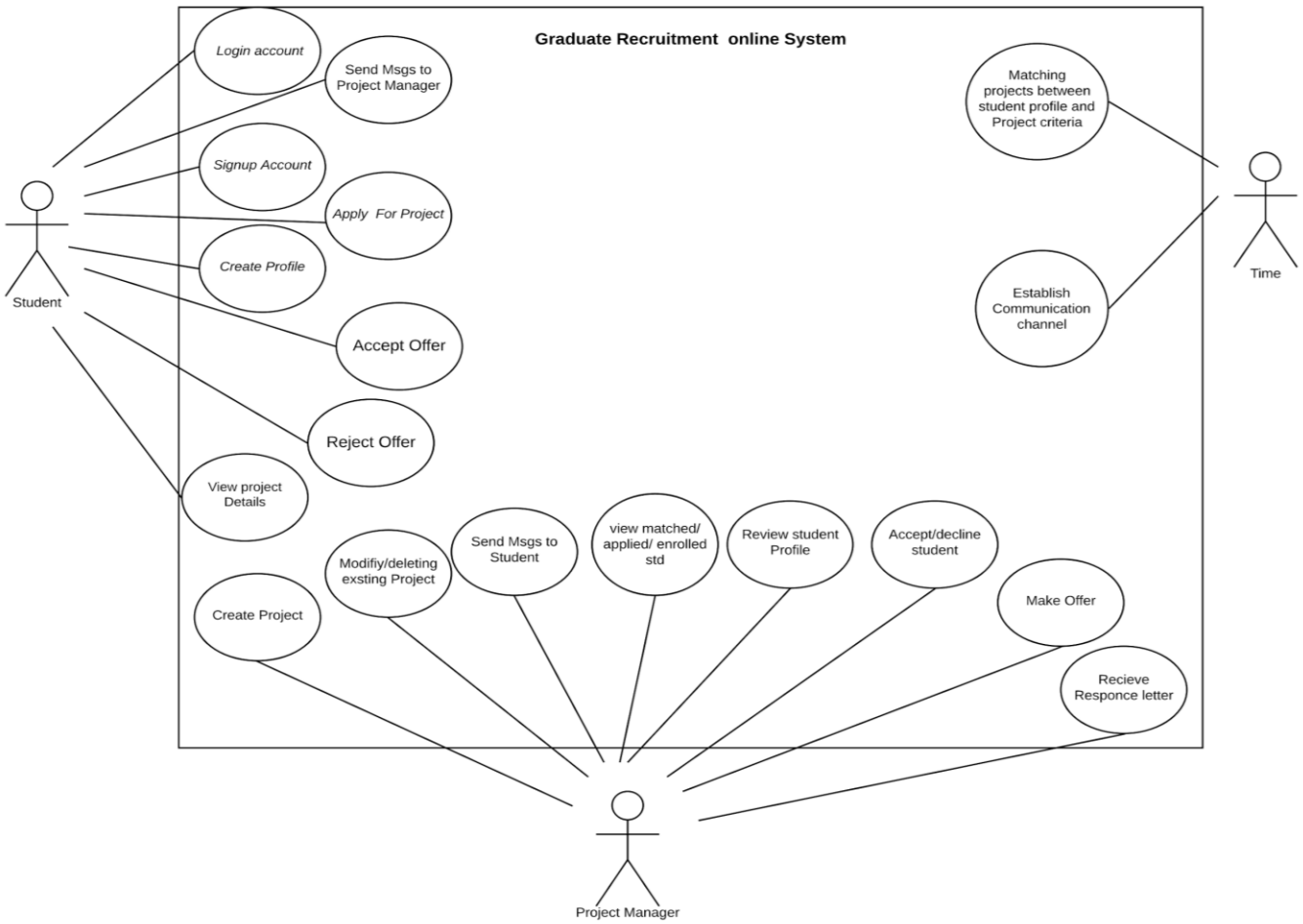
3.4. Constraints (“Pseudo requirements”)

For this full stack web application project, we use Node.js and JavaScript for our front end and back end development. Node.js is a JavaScript runtime environment established at server side which enable use implement service using JavaScript, which is consistent throughout entire stack development. For front end development, we use JavaScript,

HTML and CSS to develop the interface web pages presented to client. And using Eclipse as IDE.

4. System models

4.1. Use case model



4.2. Use case brief description :

4.2.1 Use Case Description for Project Manager

Use case name	*Description*
Create Project	Project Manager adds new project and fills up details like description, funding, department etc. Project Manager can publish the project which gets added to the project catalog
Modify Or Delete Existing Project	Project Manager can modify the existing project details or he can delete the existing project.
Review Student Profile	Project Manager can view the matched student profile for assessment.
Send Messages To Student	Project Manager can communicate with students who are matched/applied/enrolled for a project.
View Matched Applied Enrolled Std	Project Manager can view list of students who matched, applied and enrolled for each project.
Accept Or Decline Student	Project Manager can make decision by accepting or declining student's application based on assessment.
MakeOffer	If the student is accepted, the Project Manager will send the admission offer letter to the student.
ReceiveResponseLetter	The project manager will receive the response from the student whether he accepts or declines admission.

4.2.2 Use Case Description for Student

Use case name	*Description*
SignUpAccount	For the first time student signup and creates an account with credentials. An empty profile is initialized when an account is created.
LoginAccount	If the student has already signup then student needs to login in order access his account.
CreateProfile	Student can fill/modify the fields and save his/her profile draft as long as the profile has not been submitted. Once all the fields are filled, student can submit his/her profile to the system. This submit action will trigger the matching function which will return the matched projects if there is any.
ViewProjectDetails	Once the profile is submitted, student can view the matched projects and their description.
SendMsgToPM	Student can send message to the corresponding project manager for requesting any further information or negotiate the financial aid.
ApplyForProject	Once the student has decided to take the project, he/she can apply for the project. During the application of one project, the student cannot apply for another project.
ResponseToOffer	If the student receives an offer letter, the student can choose whether to accept the offer or not. If the student receives a decline letter, the student can apply for other matched projects.

4.2.3 Use Case Description for Time

Use case name	*Description*
Matching Projects between Student Profile and Project Criteria	The matching between student profile and project has to be done and displayed after student submits the profile.
Establish Communication Channel	Once the matched projects gets listed the communication channel between student and project manager has to be established .The channel is established through a webmail and communication is done through emailing. This is applicable for both student and project manager.

4.3. Detailed description of selected use case :

4.3.1 Create profile use case:

Name: Create profile
Participating actor: student
Entry condition: Student successfully registered and logged in to the online system.
Exit condition: Student fill all required fields in the online form and click on submit button or save button
Flow of events:

- **Scenario 1:**
  1. Student choose from department list which department he wants to join
  2. The student fills his education background, research interests, and work experience.
  3. The student selects the intended start date from a calendar
  4. The student determines if he needs financial support from a checkbox.
  5. The student can click save button to save his data and continue later on.
  6. The student clicks the submit button.
  7. The system should do the matching process between student profile and project criteria.
- **Scenario 2:**
  1. The student does not complete all required fields.
  2. The student clicks the submit button.
  3. The system should stay on the same webpage and Print message to the student all required fields should be filled."
- **Scenario 3:**
  1. The student does not complete all required fields.
  2. The student clicks the save button.
  3. System save all modification.

4. System prints “All modification is saved.”
5. The system stays on the same page until student clicks the submit button.

- **Scenario 4:**

1. The student does not complete all required fields.
2. The student clicks the save button.
3. Student closes the webpage.
4. The system should keep all modification in its database.

4.3.2 Create Project use case:

Name: Create Project.
Participating actor: Project manager.
Entry condition: Project manager successfully logged in to the online system and click on create project button
Exit condition: Project manager fills all required fields in the online form and clicks on the submit button or save button.
Flow of events:

- **Scenario 1:**

1. Project manager selects from checklist which department provides this project.
2. Project manager fills the project’s title and description.
3. Project manager checks if the project will be funded or not and select fund range.
4. Project manager adds project participants (professors) which will be supervisors on this project.
5. Project manager set matching criteria of this project.
6. The project manager can click save button to save his data and continue later on.
7. Project manager clicks submit button.
8. The system should save all fields on the database.
9. The System prints “Submitted Successfully.”

- **Scenario 2:**

1. The project manager does not complete all required fields.
2. Project manager clicks submit button.
3. The system should stay on the same webpage and Print message to the student “Complete all required fields.”

- **Scenario 3:**

1. The project manager does not complete all required fields.
2. The Project manager clicks on the save button.
3. The System save all modification.
4. The system stays on the same page until Project manager clicks the submit button.

- **Scenario 4:**

1. The project manager does not complete all required fields.
2. Project manager clicks the save button.
3. Project manager closes the webpage.
4. The project manager should keep all modification in its database.

4.3.3 Set channel between student and project manager use case:

Name: Set channel between student and project manager.
Participating actor: Time.
Entry condition: System will be notified that the student submit his profile and get the list of matched projects.
Exit condition: The project manager will able to communicate with the new student and student can see the contact button to communicate with the project manager if he wants to ask for additional information about a certain project
Flow of events:

- **Scenario 1:**
  1. The student submitted his profile
  2. On the submission of profile the matching process is automatically called which returns all matched project to the student
  3. Then the channel between the student and the project manager is established.
  4. The student can see the contact button to communicate with the project manager.
- **Scenario 2:**
  1. The student submitted his profile and close the online page before getting his matching list.
  2. The system should save the matched list and a status flag that student successfully submits his profile.
  3. When student login again all matched projects loaded and the communication channel will be established for him.

4.3.4 Make offer use case:

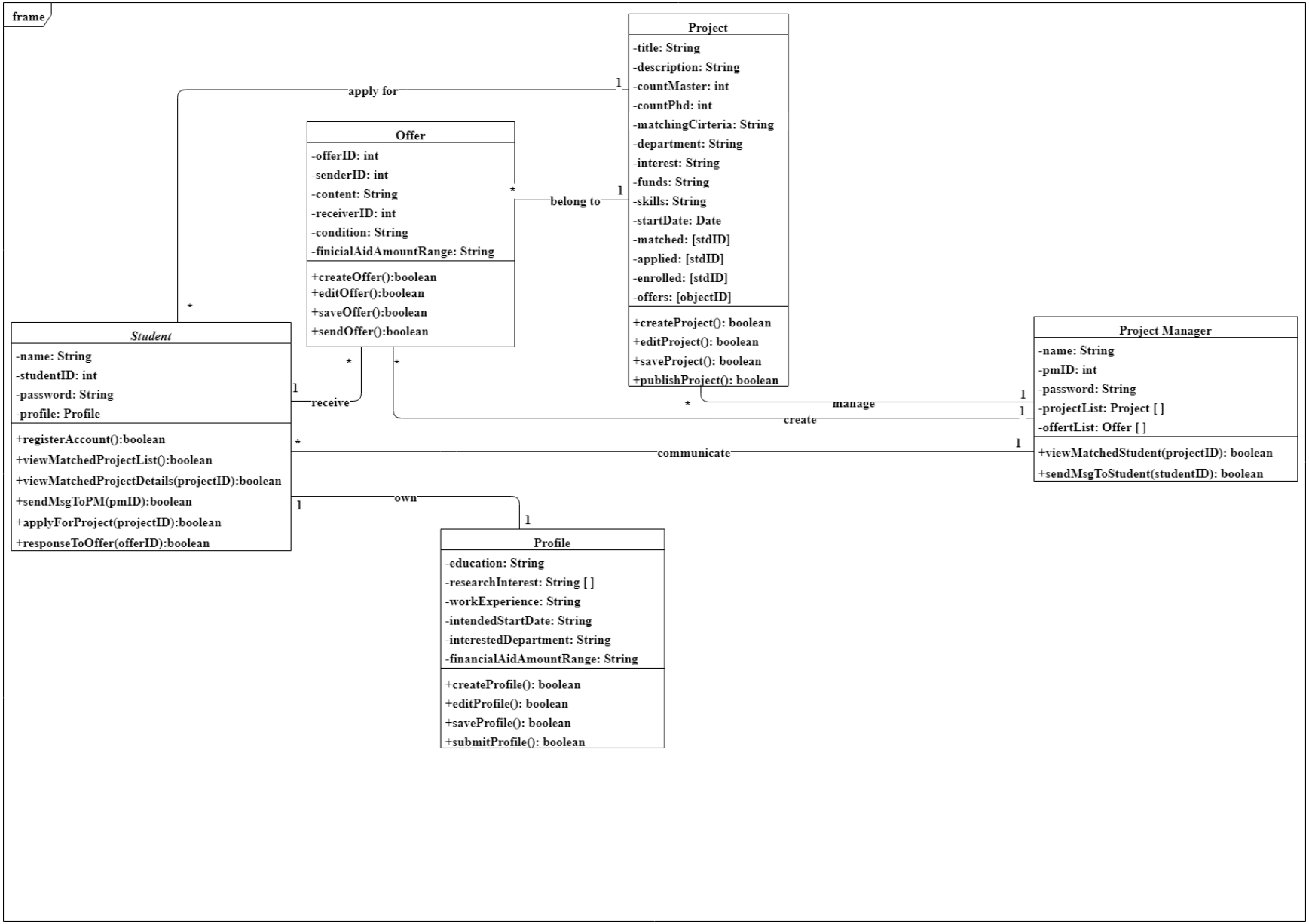
Name: Make Offer
Participating actor: Project manager
Entry condition: Project manager has a student list of matched for a project. So the project manager can click on send offer button for matched students.
Exit condition: The Offer is successfully submitted.
Flow of events:
• **Scenario 1:**
1. The project manager can review the student profile and see which project the student applied for.
2. Project manager can submit an offer for this student by Browsing the offers list for this project
3. The template is generated automatically and also he can modify the text template of offer before sending it.
4. He presses send button.

• **Scenario 2:**
1. The project manager can review the student profile and see which project the student want to apply in it
2. There is no offering list exist for this project.
3. He can automatically generate a new offer by
4. Opening new offer form
5. Selecting start date, duration of the project
6. Add budget for this project
7. System sends notification message to student.

4.3.5 Accept Offer use case:

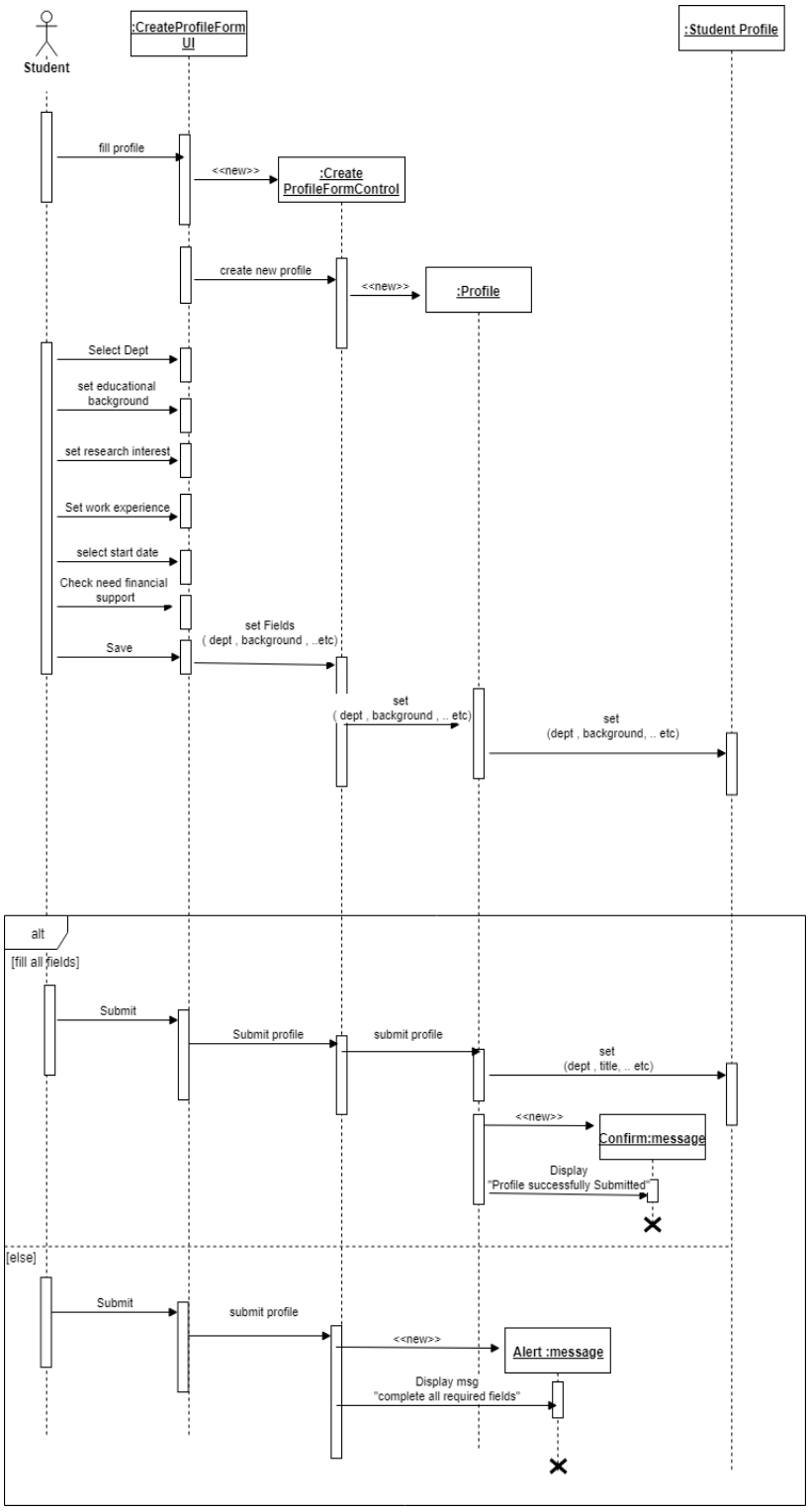
Name: Accept Offer
Participating actor: Student
Entry condition: Student receive the notification message. And the communication channel is already established so he can communicate with Project manager
Exit condition: The student Acceptance is successfully submitted.
Flow of events:
• **Scenario 1:**
1. The Student can open the offer and read it.
2. The student can send message to Project manager for more details and discuss the offer.
3. And click on accept button.
4. System appear Popup Message " The accept offer letter is successfully submitted"

4.4. Class diagrams

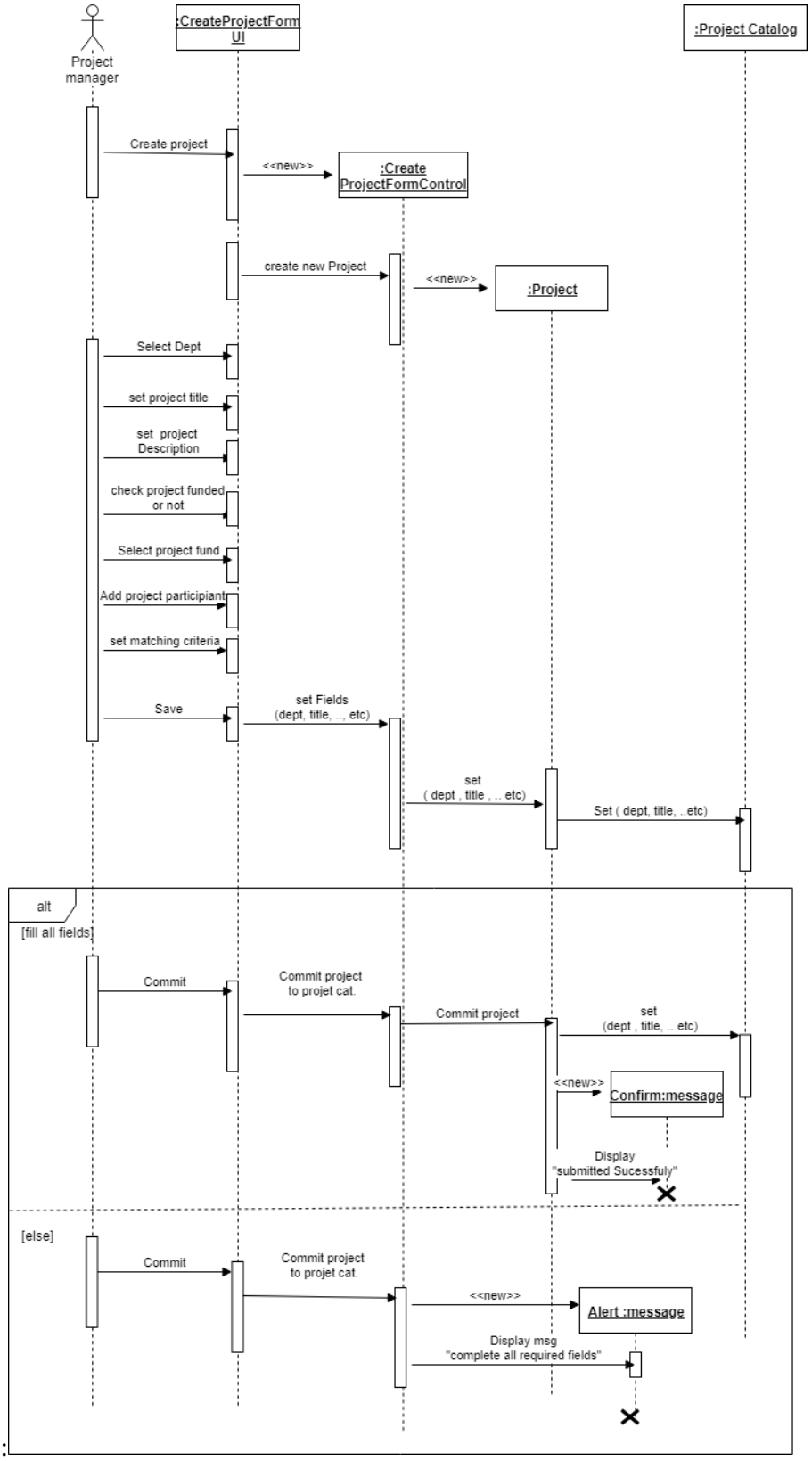


4.5. Sequence diagram

4.5.1 Create profile

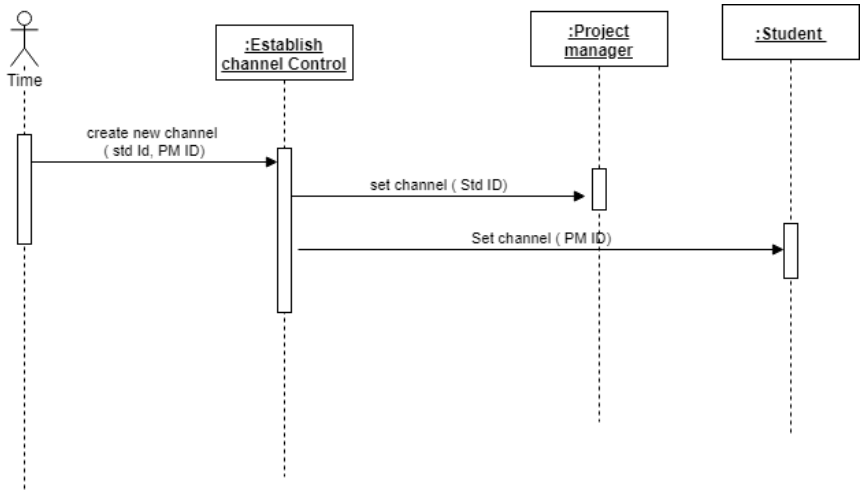


4.5.2 Create Project:

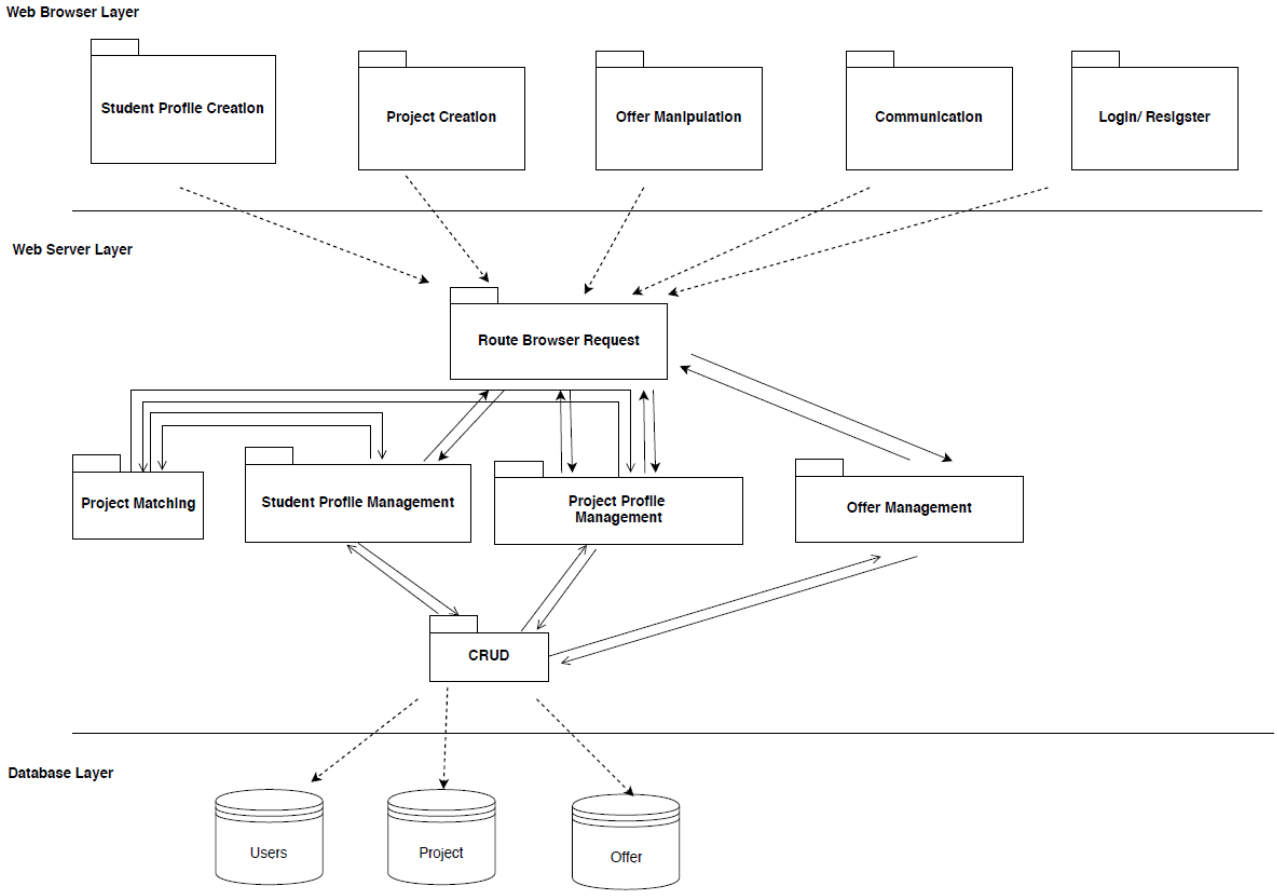


■ :

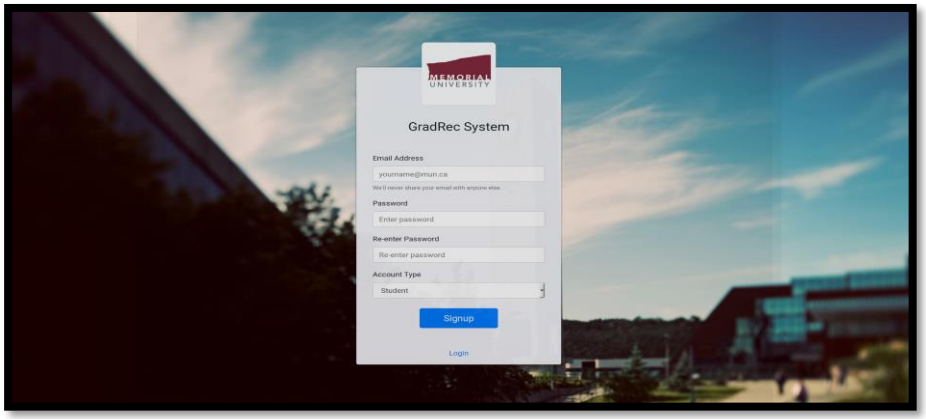
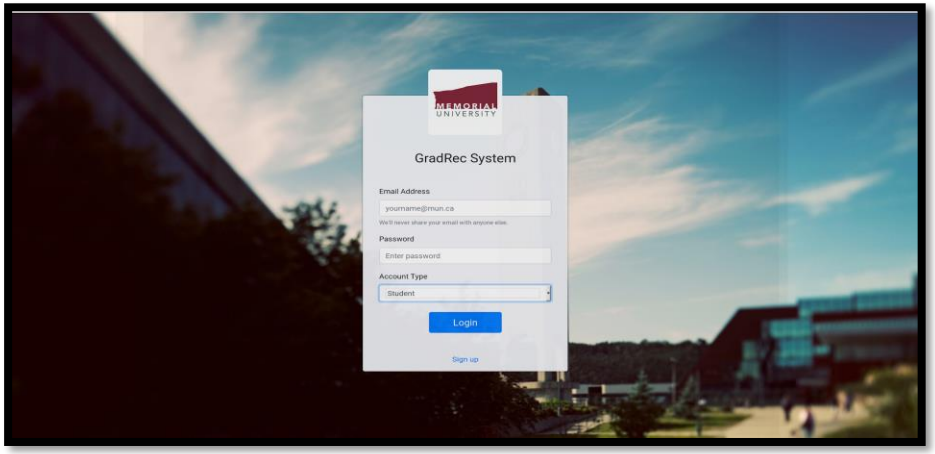
4.5.3 Set channel between student and project manager

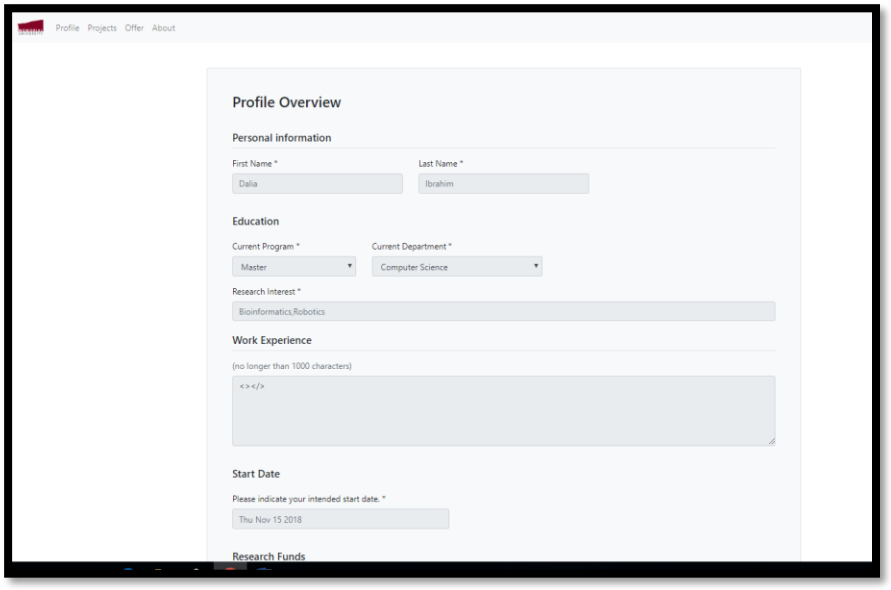
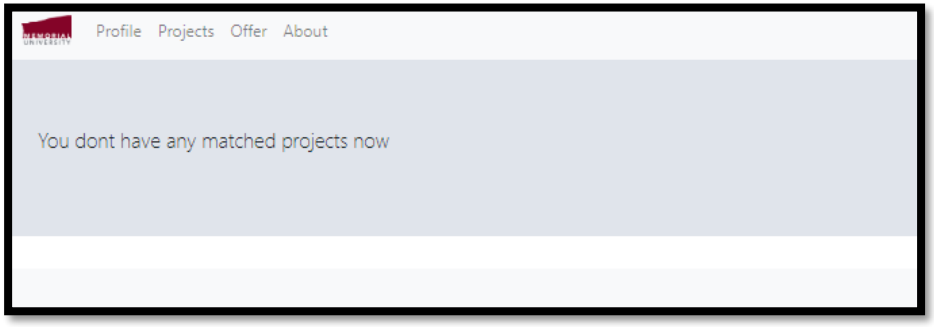


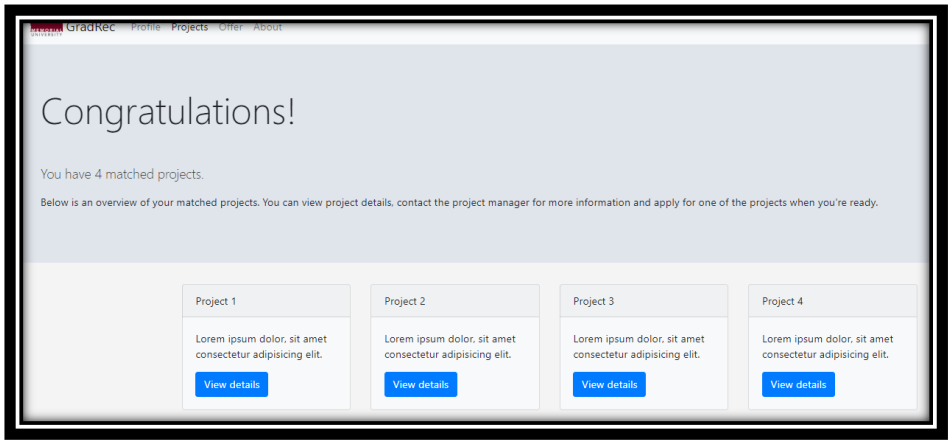
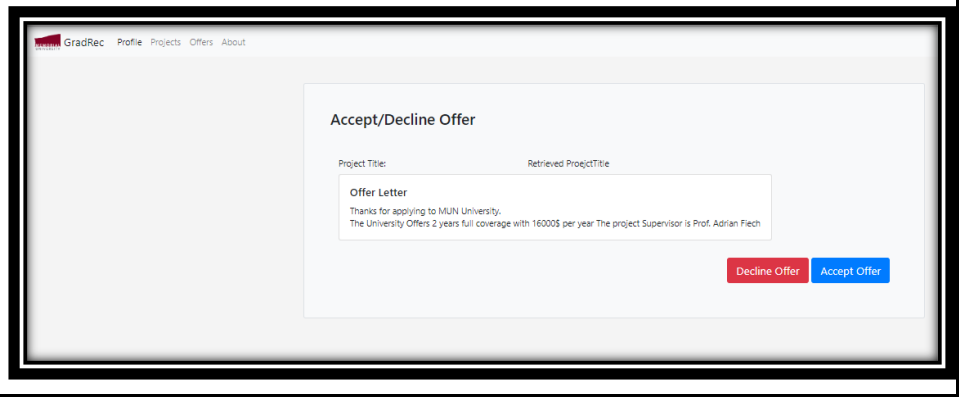
5. 3-layers-architecture

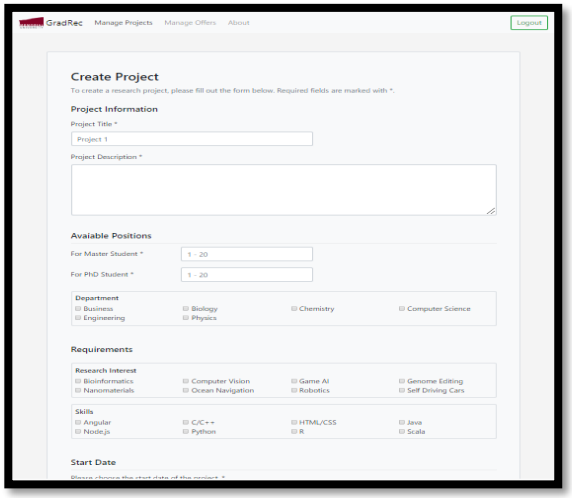


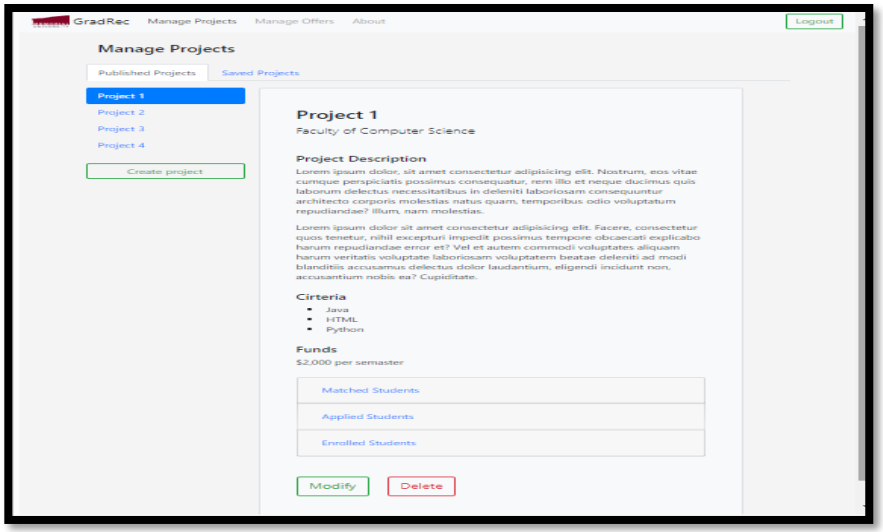
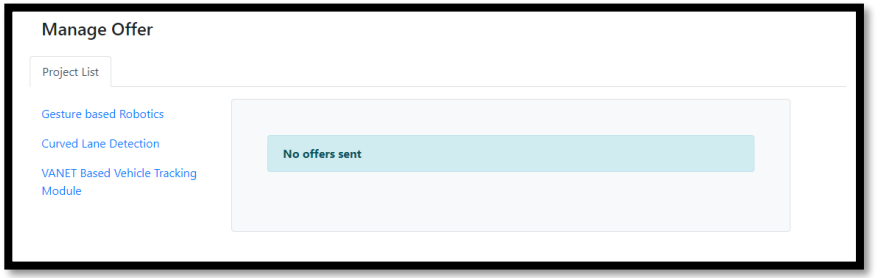
6. *User interface "screenshot"*

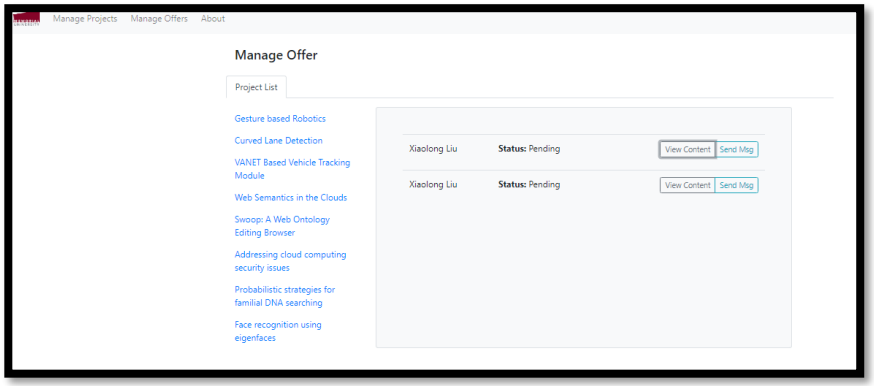
Students Views	Description	screenshot
Signup	**The student can sign up for first time he visit the website**	
login	**Student can enter his email and password and choose from dropdown list his role as a student.**	

Create profile	**This view for collecting information about student. His education, skills, ..etc.**	
Matched Projects	**If there are not any matched projects, it shows there are no matched projects.**	

	Student can view the list of matched projects and description details of each matched project.	
Offers	**When student try to open offer tab before he complete his profile information or if the application is in processing , it shows "No offers yet"**	
	When he receive offer letter he can read it and decide if he accept or decline this offer	

Project manager Views		
login	**Project Manager can login with his credentials**	
Create project	**He can create a new project and fill it's details and setting the required skills. How many position opened for this project. The project manager has capability to publish this project after completing its required fields or save it and complete later on.**	

Manage Projects	**Project Manager can view list of all the projects and their details. He can edit the details by clicking on modify and he can delete by clicking on delete.** **The project manager can view all matched students and view their profile. He can send an offer by clicking on send offer button which appears popup window for easy submitting the offer.** **He also can see the list of applied students and he can view their profile and approve of decline this student.** **The project manager can also see the list of applied students for this project**	
Manage Offers	**If no offer is sent, it shows no offers are sent.**	

	The project manager can see all submitted offers to all students for all projects	

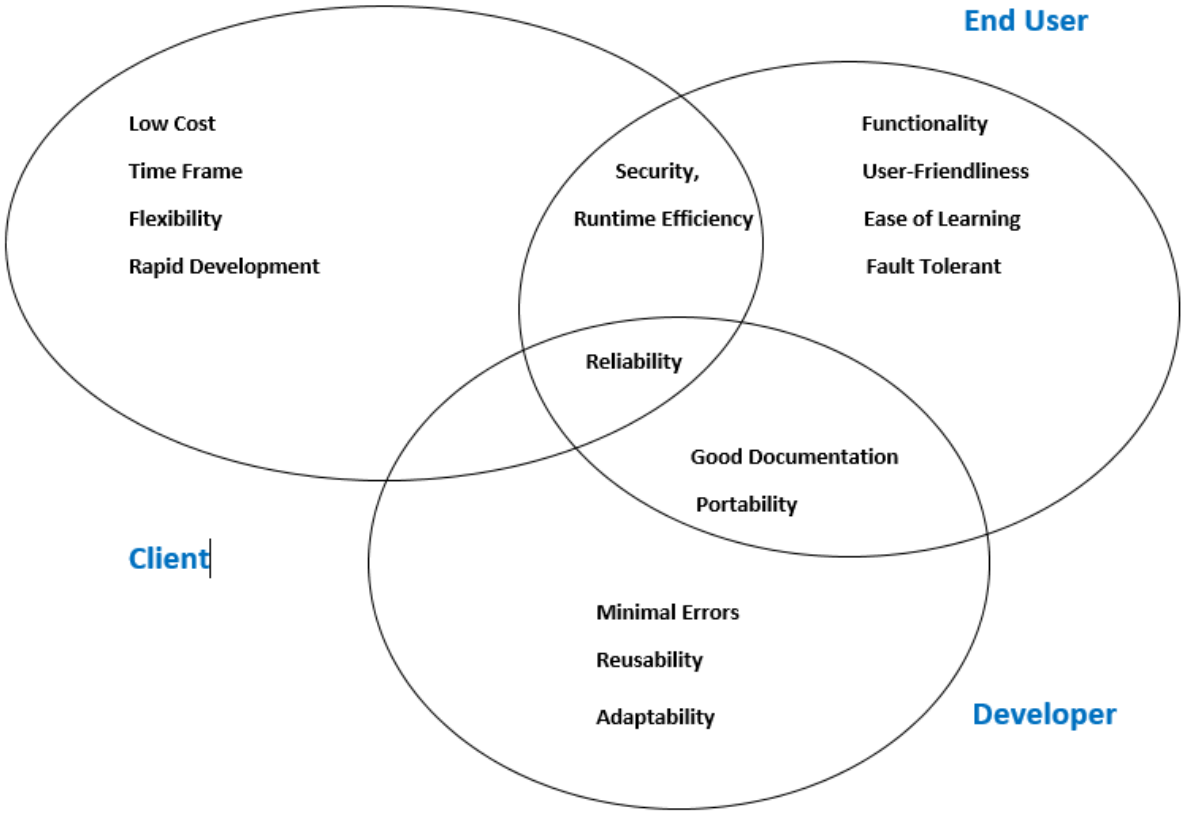
7. Design goals:

Design goals is the first step of system design. It identifies the qualities that our system should focus on. Many design goals can be inferred from the non-functional requirements or from the application domain. Here are some common goals followed by specific goals from client, end user and developer perspective.

- **Common goals for client and end user:**
  - *Security:* The communication between the project manager and student should be encrypted.
  - *Runtime efficiency* is the common goal for end user and client so that the response time is very less without any errors.
- **Common goals for client, end user and developer:**
  - *Reliability:* is the common goal that everyone expects the system has to be failure free at least for certain time
- **Common goal for end user and developer:**
  - *Good Documentation* is the goal that is expected by end user and developer so that the system is easy to understand.
  - *Portability*: The system should be run different web browser (Firefox and internet explorer), operating systems. Moreover, it will run on Android mobiles in future.

- **Design goals for client:**
  - *Low cost:* The cost of the project should not exceed $10,000.
  - *Time frame:* The first prototype of the project has to be delivered in 3 weeks since beginning of the project.
  - *Flexibility:* The product has to be flexible for future additional feature requests.
  - *Rapid development:* The product development has to be made rapidly, and produce the incremental prototypes.
- **Design goals for end user:**
  - *Functionality:* The end user expects all the functionalities are implemented
  - *User-friendliness:* The user expects the system has friendly user interface with proper colors and easy to use
  - *Ease of learning:* The developed system has to be easy to learn and the simple words have to be used in documentation and user interfaces.
  - *Fault tolerant:* The system should have proper validation mechanism to detect fault input.
- **Design goals for Developer:**
  - *Minimal number of errors:* The system has to have minimal errors to avoid the system failure.
  - *Reusability:* By using *npm(node package manager)*, we can use some of the off-the-shelf functions. Decompose the system into reusable subsystems using design patterns.
  - Adaptability: The system should be adapted the change and tolerate changes in its environment without external intervention.

Putting all the listed goals in a diagrammatic representation.



8. *System scenario example:*

Student point of view	Project administrator point of view
1. They can login to the system through the login page	
● If student do not have account yet, he should create a new account by hit the signup button ● Fill out the form and create a new account ● Login with your email and password	● He can login by this account: ○ Email : pm@mun.ca ○ Password: admin ● Pm can choose his account type in the dropdown menu
● Once the student logged in, the system will first show his profile page ○ Before the student submit his profile, the student cannot see any projects ○ Student provide his profile information, save his profile for future editing or hit submit button ○ Once submit, the student's profile can no longer be edited	● Create a project ○ Show responsive design ○ Fill the project information ○ Skills and interest areas are some predefined checkboxes, which will be considered as matching criteria. ○ PM can specify the range of available for each student ○ Show validation ○ He can publish the project by hitting the publishing button
● In projects tab, now the page will give an overview of all the matched projects. ○ It has responsive design ○ The student can click view	● Pm can switch between the list of all published projects or saved projects by the tab ● For published projects, pm can check the current matched, applied

project button to see the detailed description of the project ○ If the student need any further information, he can send PM a message ○ Once student is ready, he can click the apply button	and enrolled students.
● Now student can see his offer in the offer section, and decide whether to accept the offer or not	● Once there is a student apply, it will be shown in the applied list, ○ PM can view the student's profile and decide whether to approve his application or reject. ○ If approve, PM will send the student an offer letter.