SW Engineering CSC648/848 Spring 2019

GatorList

Team – 04

Karuna Nayak (knayak@mail.sfsu.edu)

Victor Muñoz

Huawei Gao

Dylan Shwan

Daniel Mossaband

Gabriel Alfaro

Aditya Sheoran

Milestone 2

Date Created: March 19th, 2019
Date Submitted: March 21st, 2019
Date Revised: April 2nd, 2019

1. Data Definitions V2:

Unregistered user: The user can browse listings. User can sort, search listings by keywords. User can register on the website. User cannot contact the landlord, user cannot post the new listings.

Registered user: The user has all privileges of unregistered user. User can login. User can contact the landlord. User can post the new listings. User can also edit the previously posted listings.

Administrator: Administrator are the registered users. The admin can review the listings and can approve/disapprove the listing before it goes live. Administrator can also block/unblock other user.

Register: User can register on the website by giving information such as, First name, Last name, Username(valid email id) and Password(6 to 20 characters). User must accept terms and conditions in order to be able to register successfully.

Login: Registered user can login with registered username and password.

Logout: Logged in registered user can logout from his/her account.

Listings: The listing available to rent can be posted by registered users by filling listing type, address, number of bedrooms and bathrooms, rent, images and description.

Listing Type : Listing can be Apartment, house or room

Address : Address of the listing to be rented.

Bedrooms : Number of bedrooms in the listing

Bathrooms : Number of bathrooms in the listing.

Rent : Monthly rent of the listing

Description : Landlord can provide details of the listing such as, area per sq ft, amenities included if any, if the apartment is pet friendly and neighborhood features.

Images : Landlord can post up to 5 images of the listing. Image shall be in jpg format and shall be maximum of size 5MB. Images are saved as pointers in DB.

Date : Date on which listing is posted on website.

Distance from campus : distance from campus to the listing in miles.

Commute time to campus : commute time to campus in hour/min by walk, bus and car.

User Dashboard: The registered user has a dashboard with following.

Account details : Details such as first name, last name, username, password and date of registering on website.

Pending Listing : User can view the listings he has added which are pending for Admin's approval and cannot be seen on website listings

Add Listing : User can add new listing by clicking on Add Listing

Edit listing : User can edit the existing listing details which will go for Admin's approval before populated on website.

Messages : User can check the conversions with other users and respond.

Admin Dashboard: The administrator has the all the options in registered user dashboard and has an additional button of Admin.

Admin : Upon clicking on admin button, user can view the listings which are pending approval/disapproval. Upon clicking on the listing, admin can view all the listing details and has buttons for approve and disapprove. Upon approval, the new listing populates on website. Admin can also view users to delete them.

2. Functional Requirements:

Priority 1:

Unregistered Users:

1.1 Unregistered users shall be able to browse through available listing information through search or filter.

1.2 Unregistered users shall be able to register.

1.3 Unregistered users shall be prompted to register or sign-in upon confirmation or contact landlord.

1.4 Unregistered users shall be required to accept terms & conditions during registration.

1.5 Unregistered users shall be able to sort by posted date, price and distance.

1.6 Unregistered users shall be able to see listing information related to their search.

1.7 Unregistered users shall be able to browse by filter.

1.8 Unregistered users shall be able to search by keyword defined by location, room# available, type of apartment/house, etc..

Registered Users:

2.1 Registered users shall have all the function of unregistered user has.

2.2 Registered users shall be able to login.

2.3 Registered users shall be able to post new listing.

2.4 Registered users shall have options to confirm or cancel before posting the new listing.

2.5 Registered users should accept terms and conditions during registration.

2.6 Registered users shall be able to message the landlord.

2.7 Registered users shall be able to view their listings and messages.

2.8 Registered users shall be able to view their own dashboard.

2.9 Registered users shall be able to check the status of the newly posted listing in user dashboard.

Admin:

3.1 Admin shall have an admin panel.

3.2 Admin shall be able to approve/disapprove all listing through admin panel before they go live.

3.3 Admin shall have all the function of unregistered/registered user has.

3.4 Admin shall be able to review all listing through the admin panel.

3.5 Admin shall be able to block/unblock users through the admin panel.

Priority 2:

Unregistered Users:

-

Registered Users:

2.1 Registered users shall be able to edit/delete their listing.

Admin:

-

Priority 3:

Unregistered Users:

-

Registered Users:

2.1 Registered users shall be able to edit their personal information through user dashboard.

2.2 User dashboards shall show all viewed and tagged(favorite) listing information.

2.3 Registered users shall be prompted a status of the newly posted listing whether being approved or disapproved by an admin.

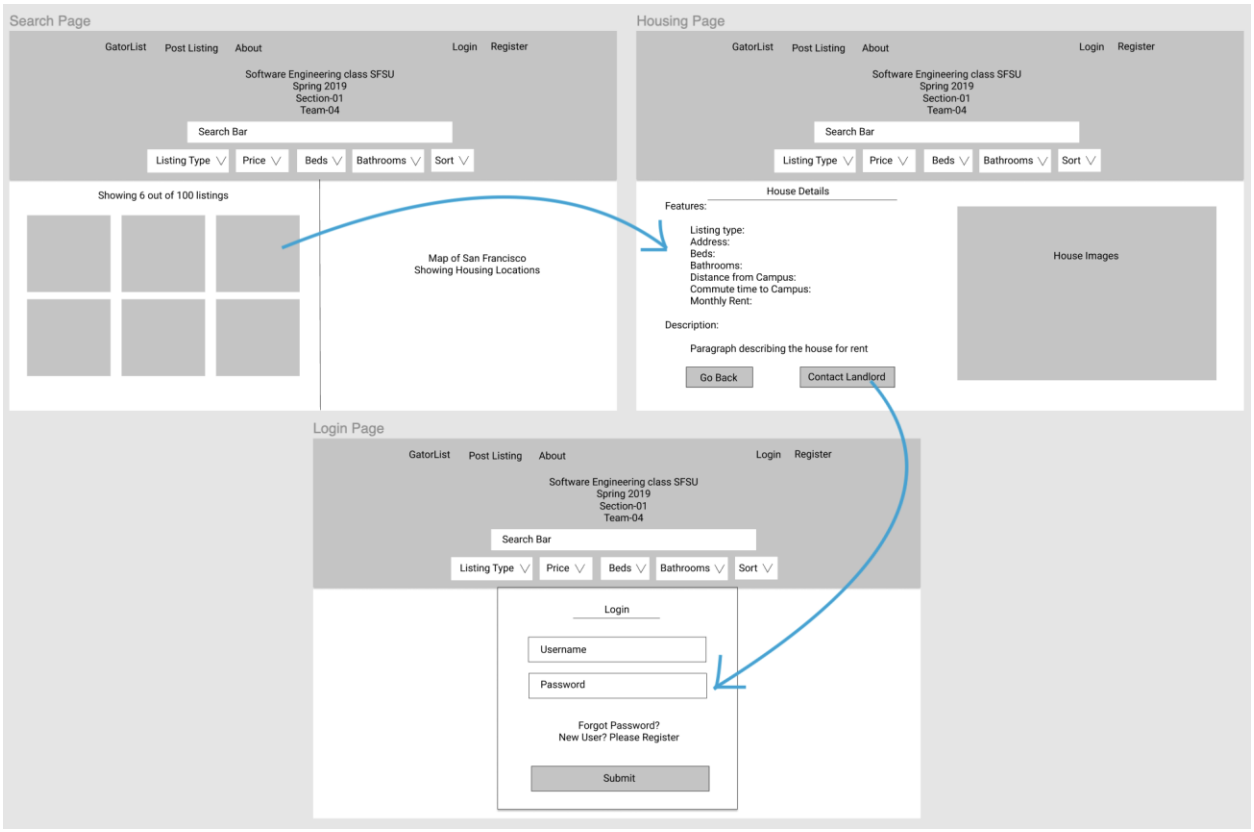
Admin:

-

3. UI Mockups and Storyboards:

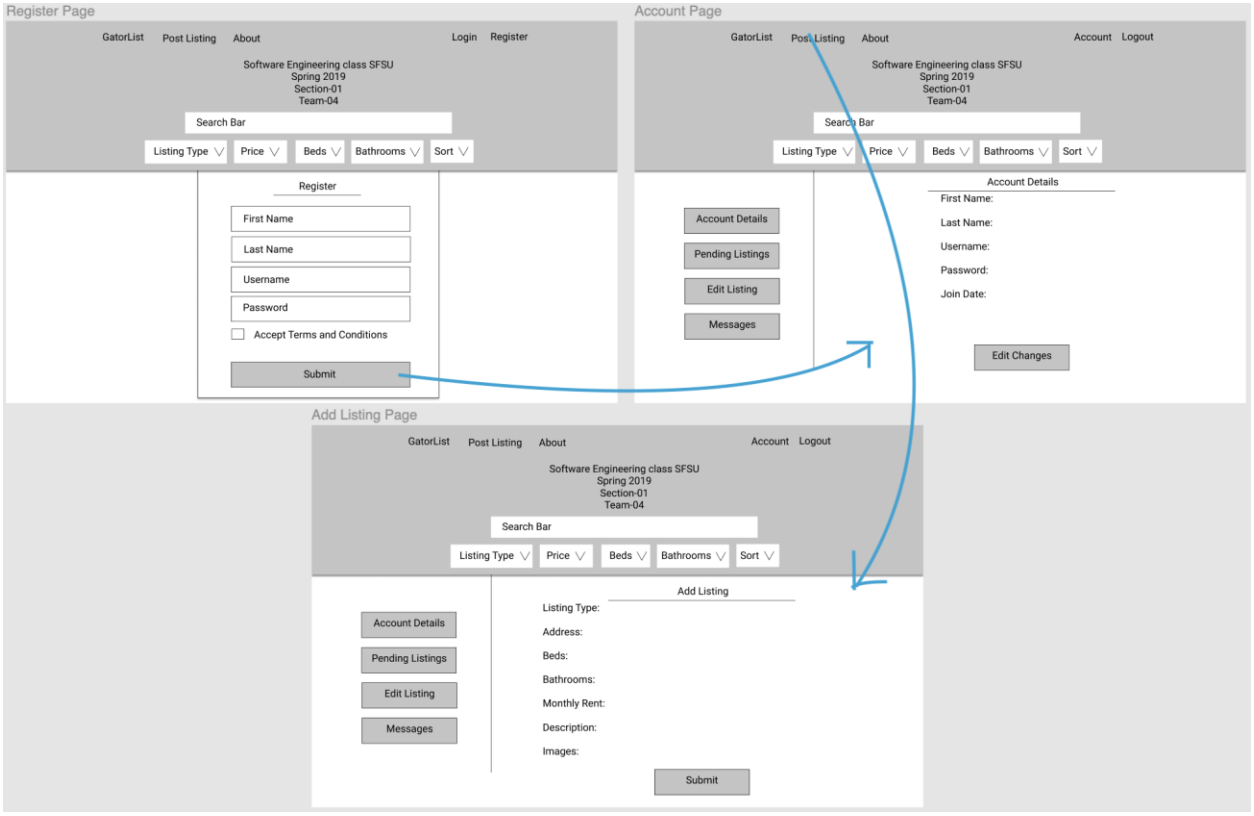
- *Use Case 1 (Unregistered User-tenant):*

Unregistered user can search and browse but cannot contact landlord.



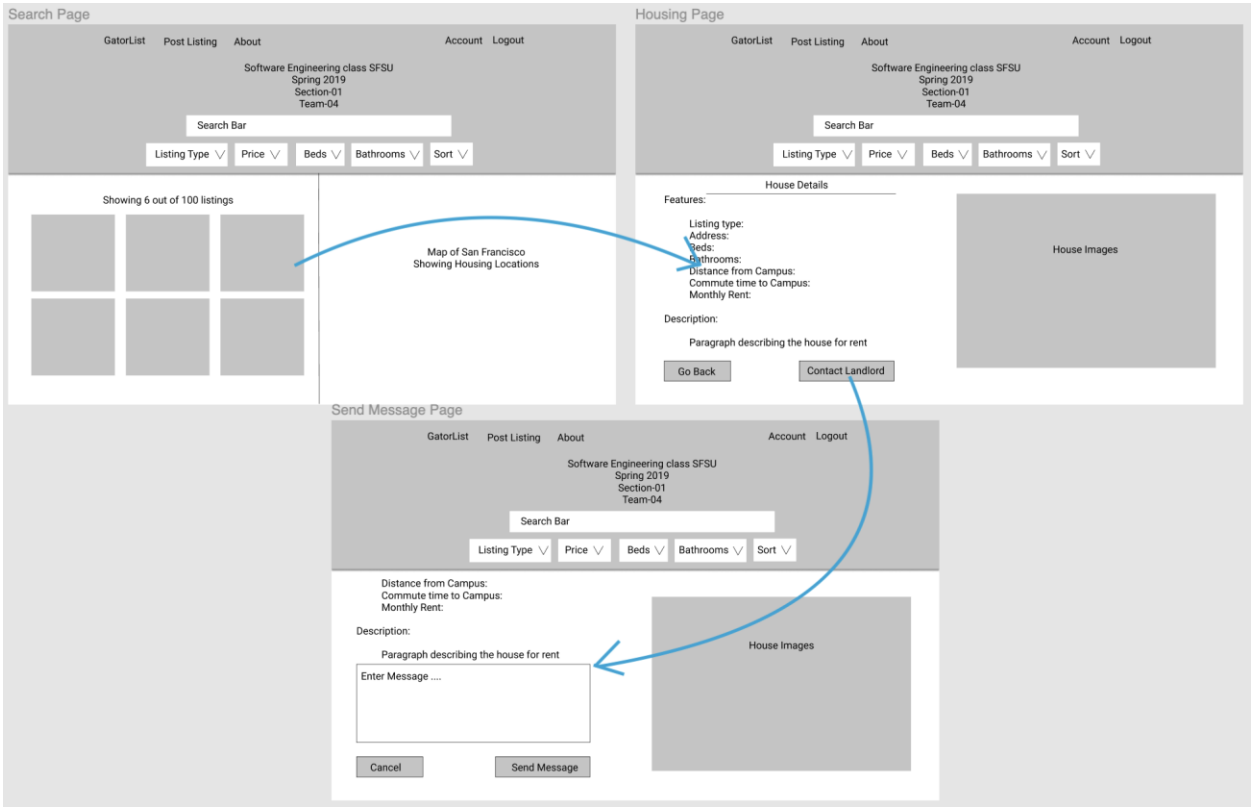
- *Use Case 2 (Unregistered User - landlord):*

Unregistered user can register and add listings once logged in.



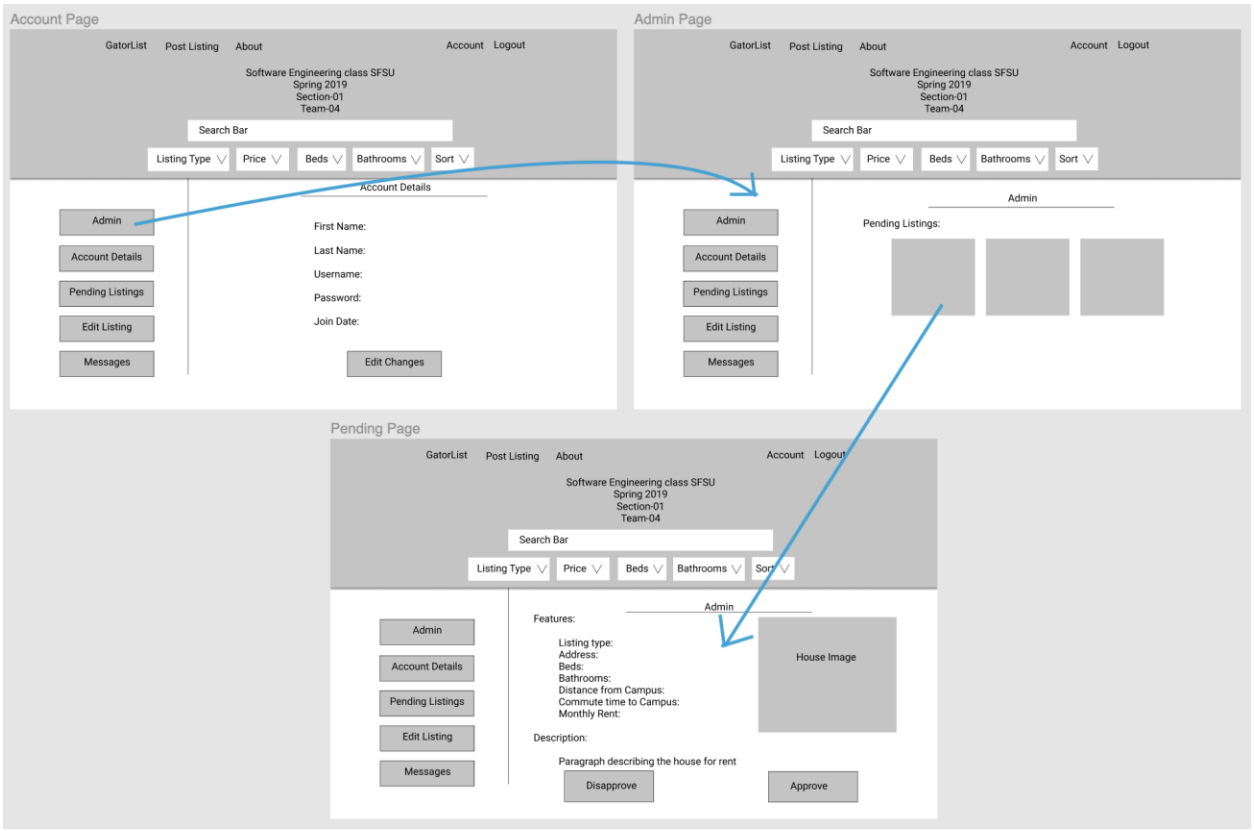
- *Use Case 3 (Registered User):*

Registered user can search through all available listings and may contact landlord.



- *<u>Use Case 4 (Admin):</u>*

  Admin can approve or disapprove pending listings.



4. High level Architecture, Database Organization:

DB organization:

We are going to have one main Database titled ‘team04db’. Inside this Database we will have the tables titled the following: `users`, `conversations`, and `apartments`.

Users:

INT(11) user_id	VARCHAR(70) first_name	VARCHAR(70) last_name	VARCHAR(255) email_address	CHAR(16) password	TEXT starred_apts	DATETIME timestamp

Messages:

INT(11) convo_id	INT(11) sender_id	INT(11) receiver_id	TEXT message	DATETIME timestamp

Listings:

INT(11) id	VARCHAR (9) type	TEXT address	TINYINT(3) bedrooms	TINYINT(3) bathrooms	FLOAT rent	TEXT description

DATE TIME date	FLOAT distance_from _campus	INT(11) commute_time _to_campus	INT(11) landlord_id	TEXT combined	FLOAT latitude	FLOAT longitude

Media storage:

We are going to be storing images on the server. We will be only be accepting images smaller than 5mb and then further compressing all images server side before permanent storage. We will protect these files from public access by storing them one folder above the domain directory.

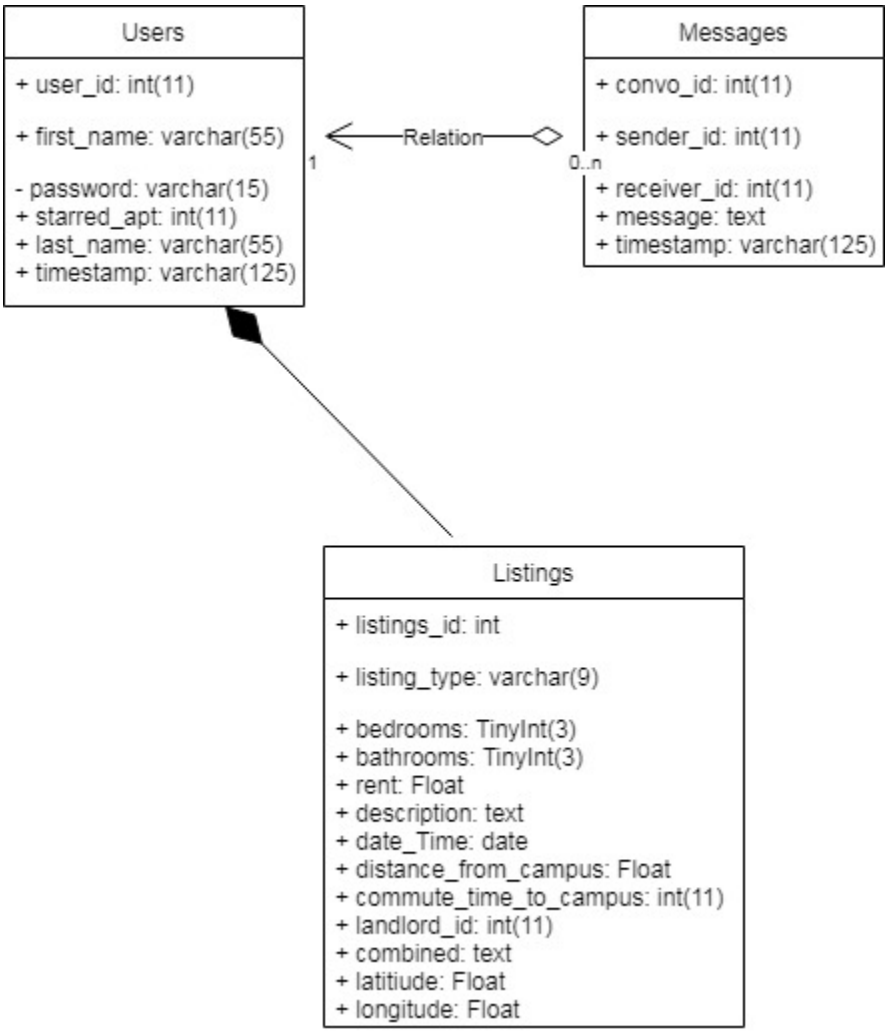
Search/filter architecture and implementation:

Registered User and Unregistered User will be able to filter results by premade filters. These filters will be: Housing Type, Number of Rooms, Number of Bathrooms, Proximity to Campus, Commute Time to Campus and Newest Apartments. For any of these filters, there will be a simple SQL query to retrieve rows from the ‘apartments’ table. There is also an SQL column titled ‘combined’ that will concat every other column.

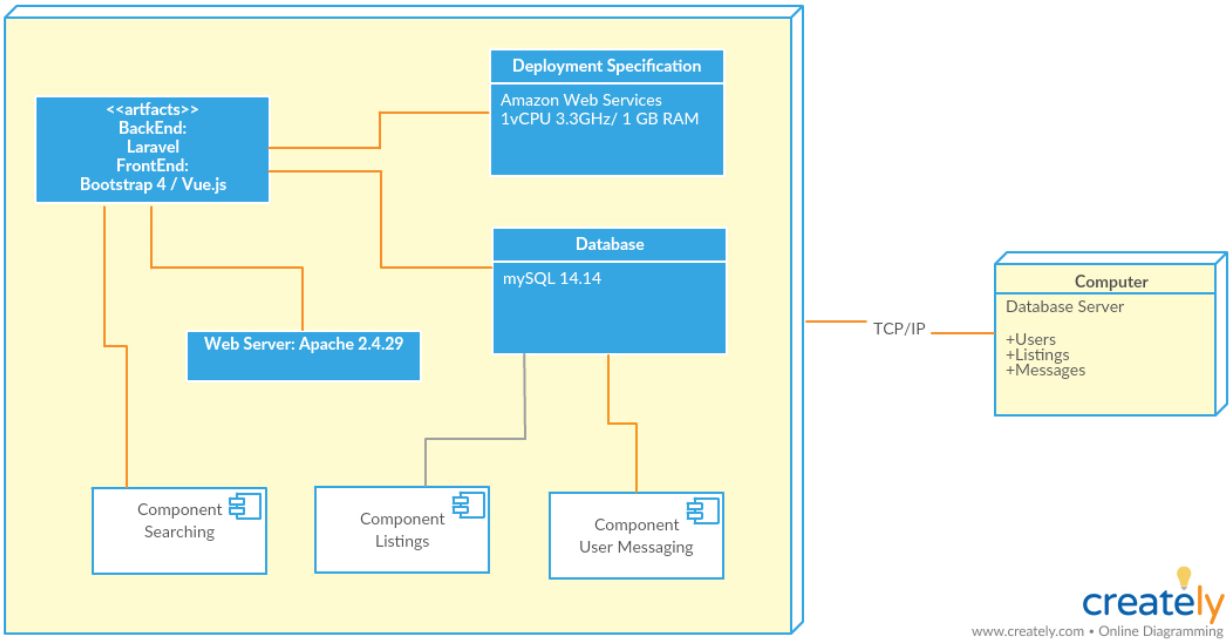
When a Registered or Unregistered user searches, we will perform a LIKE SQL query.

5. High Level UML Diagrams:

a) ***UML Class Diagram***



b) ***Deployment Diagram***



6. Identify actual Key risks for your project at this time:

1. Skills Risks

1. Even though our team has had some experience in programming, there would still be some risk in this project since none of us have ever stepped out of our comfort zone to build a website from scratch. However to solve this problem we will have to put more effort into learning as a team and use the information that were posted such as the tutorials.

2. Schedule risks

1. We should not be having any schedule risk because our team number one rule is to finish all our work a couple of days before the deadline. This way we can correct any miss information before the due date and check if all the requirements are met. In order for this to work, communication is the most important role to be on track.

3. Technical risks

1. Throughout the project, we might face some technical risk, since this is a team project. Potential technical issues we may face could be having a merge conflict or having the server go down. To solve any of these issues we have our team lead, front end lead and backend lead frequently check if everything is working properly.

4. **Teamwork risks**

    1. As long as we communicate properly and finish our individual task there should not be any teamwork risk. Since we are using Trello and slack, each member of the team knows what to work on.

5. **Legal/content risks**

    1. In order to avoid any legal/content risk, our team plans to keep everything private and original and use our own content. In any case, if there were any legal/content risks, we plan to put a disclaimer to ensure users that this is just a project for a class.

7. **Project Management:**

Up until starting Milestone 2, we have almost exclusively used Slack for all forms of communication apart from meeting in person. Even though this has worked aimlessly so far, we are aware we must use other services to stay on track as the workload starts to increase. For Milestone 2 and onwards, we are using Trello as our primary task management service. With the gradual shift towards frontend and backend teams working more independently, we established various tasks viewable on the Trello dashboard indicating what tasks are to be done by frontend, backend, or both groups as well as their current status. Furthermore, checkpoints will be established to maintain workflow and reduce idleness.