

SW Engineering CSC648/848 Fall 2019

Gator-Aid

Section 02

Team 09

(local)

TEAM LEAD

Nayan Pandey

npandey1@mail.sfsu.edu

Nikhil Mohan – Front-end Lead

George Pernov – Back-end Lead

Stephanie Sechrist – GitHub Master

Ali Nasralla

Jackie Huang

Milestone 2

10/18/2019

Revision	Date	Description
-	10/18/2019	Initial Submission
1	10/25/2019	Revised per Dr. Petkovic's comments

Table of Contents

1. Functional Requirements – Prioritized.....	2
2. UI Mockups and Storyboards.....	3
3. High Level Architecture & Database Organization	12
3.1 Database Organization.....	12
3.2 Media Storage.....	14
3.3 Search Architecture and Implementation	15
4. High Level UML Diagrams	16
4.1 High-Level UML Class Diagram.....	16
4.2 UML Component and Deployment Diagram	17
5. Key Risks	17
5.1 Skill Risks.....	17
5.2 Schedule Risks.....	17
5.3 Technical Risks	17
5.4 Teamwork Risks	18
5.5 Legal/Content Risks.....	18
6. Project Management	18

1. Functional Requirements – Prioritized

Priority 1

Unregistered Users

1. Shall be able to browse items by category
2. Shall be able to view listings
3. Shall be able to register an account with an SFSU email account
4. Shall be able to search items
5. Shall be able to sort listings by price

Registered Users (includes above)

6. Shall be able to log in
7. Shall be able to post items for sale as listings
8. Shall be able to contact sellers

Administrator (includes above)

9. Shall be required to approve or reject listings
10. Shall be able to delete listings
11. Shall be able to delete users

Priority 2

Unregistered Users

1. Shall be able to sort listings by post date

Registered Users (includes above)

2. Shall be able to remove their own listings
3. Shall be able to respond to messages from buyers

Priority 3

Unregistered Users

1. Shall be able to search for books by department and/or course number
2. Shall be able to edit their own listings

Registered Users (includes above)

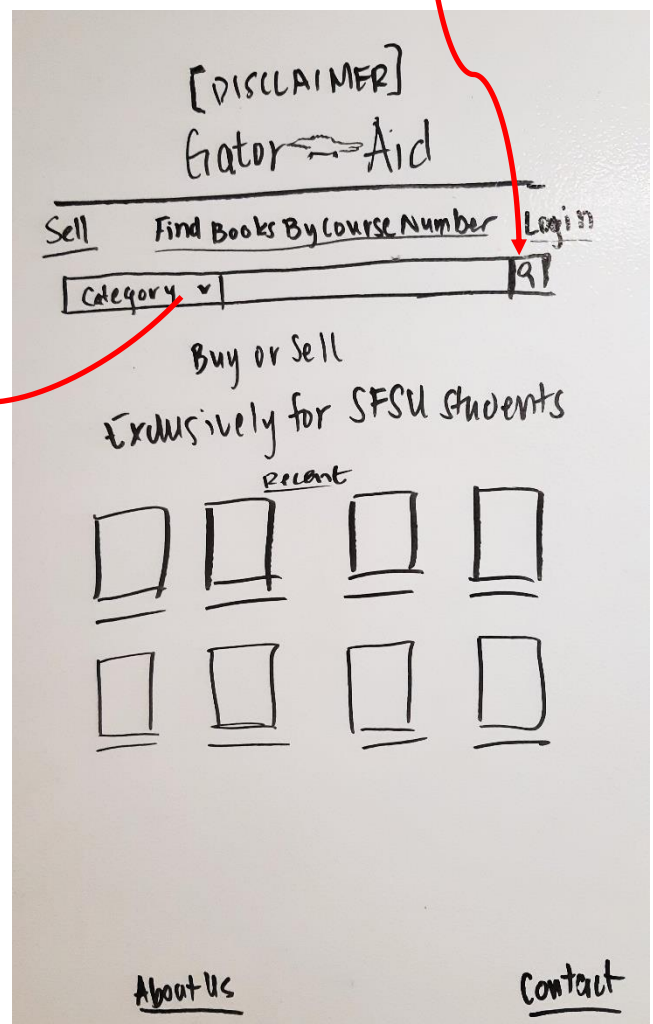
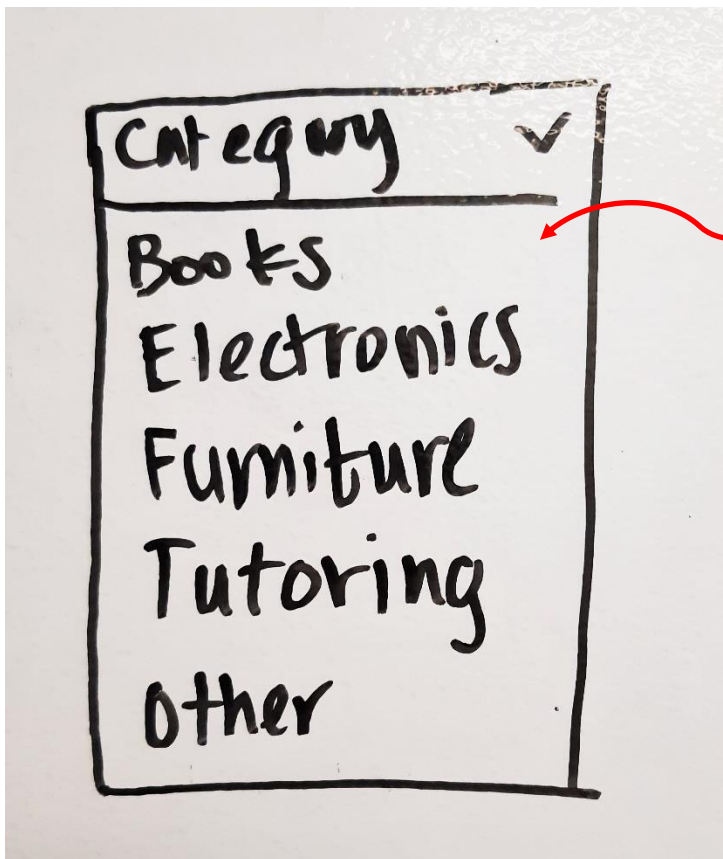
3. Shall be able to flag listings as inappropriate

2. UI Mockups and Storyboards

Use Case 1: Unregistered User

Alexa wants to buy a certain electronic item. She searches Gator-Aid after selecting the Electronics category, finds the item she wants, and clicks on the listing. She clicks Buy and is prompted to log in or register. Once she does, she can message the seller.

Alexa selects
"Electronics" from drop-
down Category list (seen
left), types in her search
query, and clicks the
search icon.



[Disclaimer]

Gator Aid

Sell

Find Books By Course Number

Login

Category v | 19

Results for " "

Showing — out of — results

Sort by: Date v

img

\$ —

img

\$ —

img

\$ —

img

\$ —

About us

Contact

Clicks listing

[Disclaimer]

or — Aid

Find Books By Course Number

Login

Category v | 19

Title of Post

img

...

Price \$

Contact Seller

Description

Flag as Inappropriate

About us

Contact

Clicks contact
seller button

Log-in pop up

A hand-drawn sketch of a log-in pop-up window. It features a title bar with a close button (X) in the top right corner. The main content area contains the following elements from top to bottom: a label 'username' followed by a text input field, a label 'password' followed by a text input field, a rectangular button labeled 'Log In', the text 'Forgot Password?', and the text 'Register'. A red arrow originates from the right side of the pop-up and points towards the 'Register' text.

Registration pop up

A hand-drawn sketch of a registration pop-up window titled 'Create New Account' with a close button (X) in the top right corner. The form contains the following elements from top to bottom: a label 'SFSU email' followed by a text input field, a label 'username' followed by a text input field, a label 'password' followed by a text input field, a label 'confirm password' followed by a text input field, a checkbox followed by the text 'I agree to Terms & Conditions.', and a rectangular button labeled 'Register'. Two red arrows point from the left towards the checkbox and the 'Register' button.

Message Seller in new tab

[Disclaimer]

Gator Aid

Sell Find BooksBy Course Number username

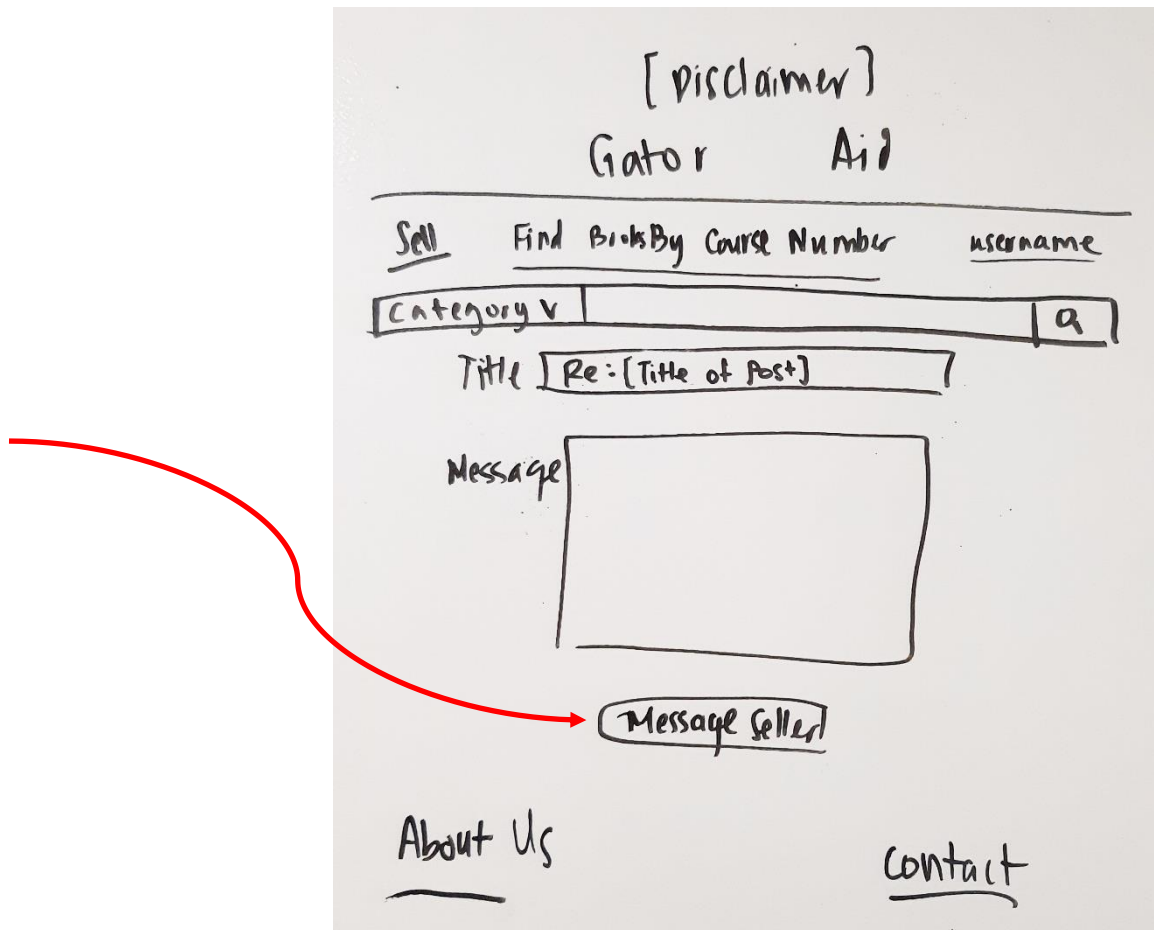
Category v [a]

Title [Re: [Title of Post]]

Message


Message Seller

About Us Contact



Use Case: User/Seller

Jennifer wants to sell her used textbooks. After clicking sell, she has to fill out the necessary fields and upload a photograph of the book. After clicking Post Listing, she must login or register. Her information is saved after doing so, and she clicks Post Listing again.









[Disclaimer]
Gator  Aid

Sell ~~Find Books By Course Number~~ Login

Category ▾

Buy or Sell
Exclusively for SFSU students

Recent

About Us Contact

[Disclaimer]
Gator - Aid

Sell Find Books By Course Number Login

Category ▾

Sell an Item

What are you selling?

Title

Description (300 character limit)

Upload Photos

Price

About Us Contact

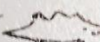
Log-in pop up

A hand-drawn sketch of a log-in pop-up window. It features a title bar with a close button (X) in the top right corner. The main content area contains the following elements from top to bottom: a label 'username' followed by a text input field, a label 'password' followed by a text input field, a rectangular button labeled 'Log In', the text 'Forgot Password?', and the text 'Register'. A red arrow points from the 'Register' text to the right, extending towards the registration pop-up sketch.

Registration pop up

A hand-drawn sketch of a registration pop-up window titled 'Create New Account' with a close button (X) in the top right corner. The form contains the following elements from top to bottom: a label 'SFSU email' followed by a text input field, a label 'username' followed by a text input field, a label 'password' followed by a text input field, and a label 'confirm password' followed by a text input field. Below the input fields is a checkbox followed by the text 'I agree to Terms & Conditions'. At the bottom is a rectangular button labeled 'Register'. Two red arrows point from the left towards the checkbox and the 'Register' button.

[Disclaimer]

Gator  Aid

Sell Find Books By Loose Number Log-In
Category VI 19

Would now show
username

What are you selling? Category VI

Title

Description (300 character limit)

Upload Photos

Browse

Price

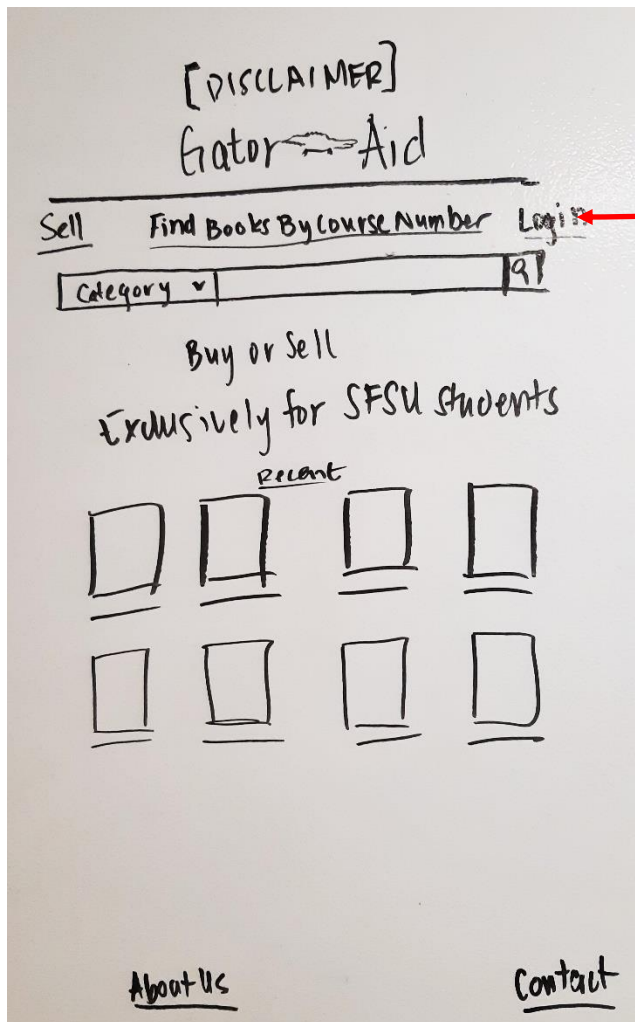
Post Listing

About Us

Contact

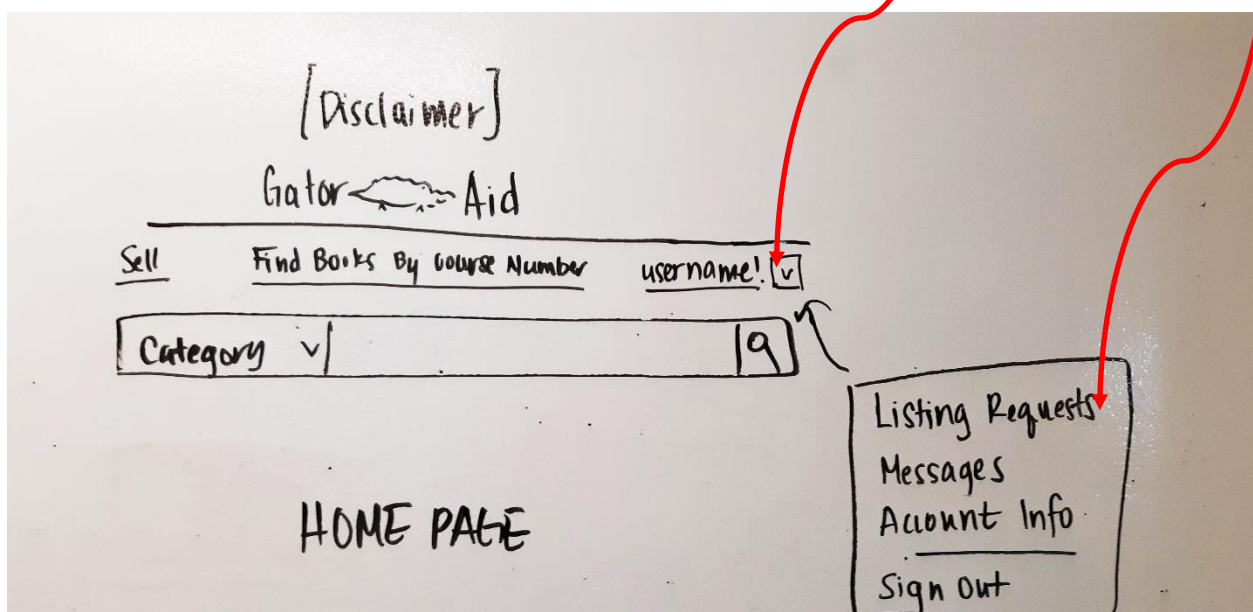
Use case: Administrator

John logs in and sees that he has a notification (!) about new listings. On the listing requests page, he sees listings waiting to be approved. He clicks the titles to view each listing and either approves or rejects the listing. After doing so, he is returned to the Listing Requests page.




Clicks pull-down
menu to access
Listing Requests

Clicks Listing Requests



[Disclaimer]

Gator  Aid

Sell Find Books By Course Number username! ☒

Category ☒

Sort By: ☒

user	listing	type	date

About Us Contact

Clicks title of listing

[Disclaimer]
or - Aid

Books By Course Number Login

Title of Post

< >

Price \$

Description

About Us

Contact

3. High Level Architecture & Database Organization

3.1 Database Organization

Users:

Name	Data Type	Default View	NULL
user_id	uuid	Auto Increment	No
first_name	varchar(30)		No
last_name	varchar(30)		No
email	varchar(30)		No
password	char(32)		No

Categories:

ID	Category
1	Books
2	Electronics
3	Furniture

4	Tutoring
5	Other

Listings:

Name	Data Type	Default View	NULL
listing_id	uuid	Auto increment	No
user_id	uuid		No
date_created	date	Current Timestamp	No
description	text		Yes
price	decimal(7,2)		Yes
image	Varchar(255)	NULL	Yes
confirm_listing	boolean	False	No
title	Text		No
category	int		No

is_textbook	Boolean	False	No
class_number	int		Yes
department	varchar(30)		Yes

Messaging:

Name	Data Type	Default Value	NULL
message_id	uuid	Auto Increment	No
sending_user	uuid		No
listing_id	uuid		No
message	varchar(255)		No

Class Number:

Name	Data Type	Default View	NULL
department	varchar(30)		No
class_number	int		No

3.2 Media Storage

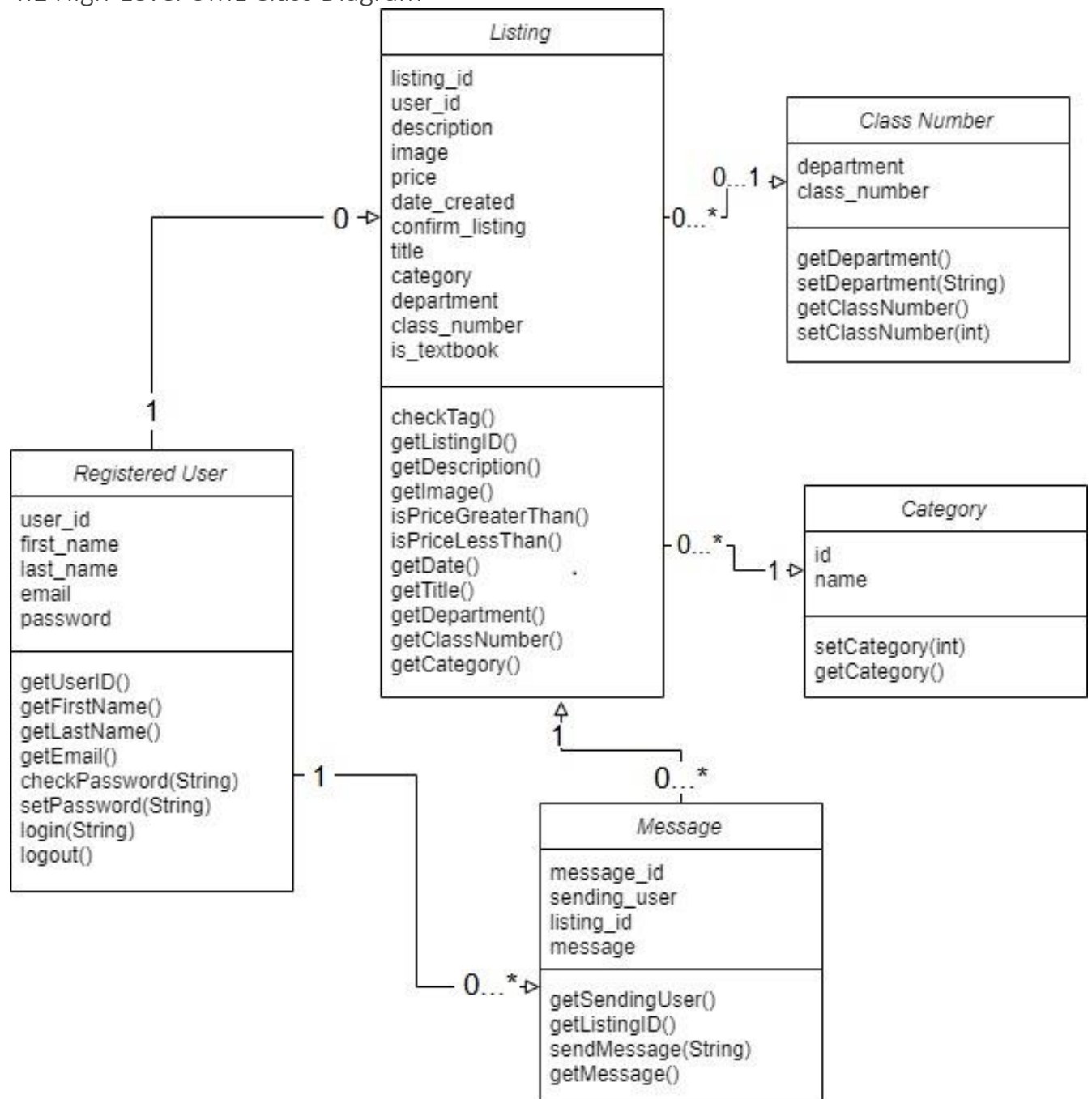
All media will be kept in a file system format with the database containing the relative file path names pointing to all the media being used.

3.3 Search Architecture and Implementation

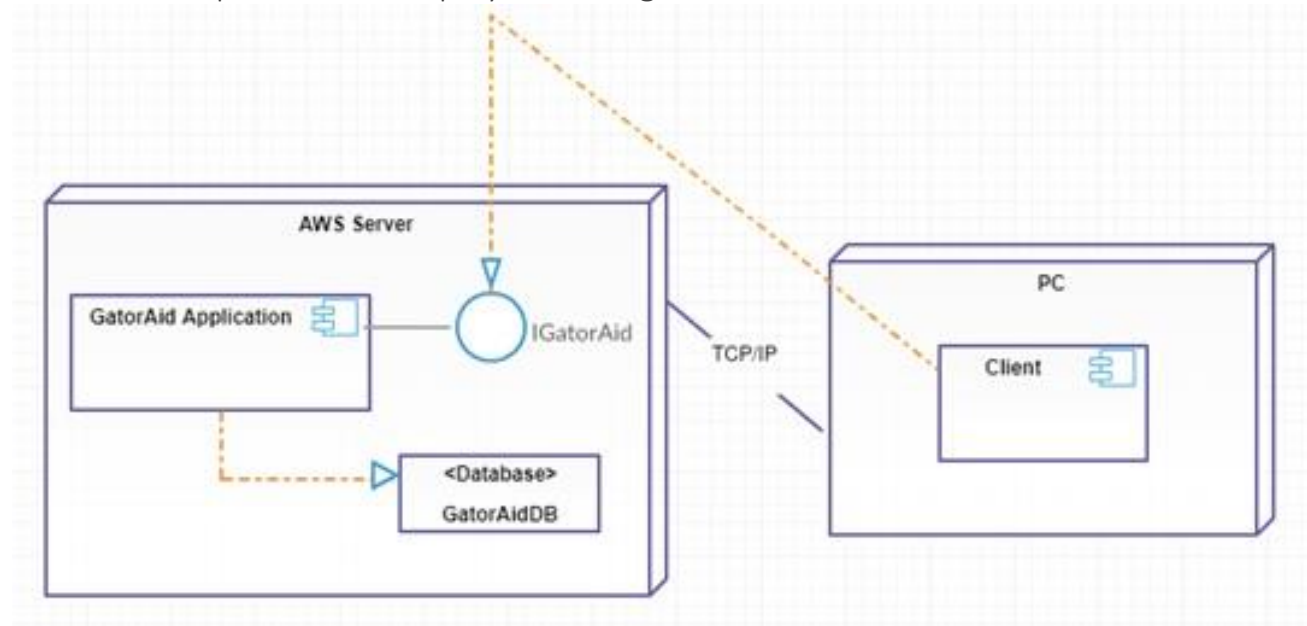
- If the user does not input anything in the search bar or choose a category, the display page will show a variety of listings from all categories, sorted by the date created.
- If the user chooses a category but does not input anything in the search bar, the display page will show a variety of listings from the specified category, sorted by the date created.
- If the user enters a keyword in the search bar, listings whose title or description contains the keyword will be listed, sorted by the date created. This is implemented using the %LIKE operator.
- APIs to be used: Postgres, %LIKE,

4. High Level UML Diagrams

4.1 High-Level UML Class Diagram



4.2 UML Component and Deployment Diagram



5. Key Risks

5.1 Skill Risks

A major skill risk that we have is learning the new technologies that we have chosen for this project. We each have a variant degree of knowledge on the technologies that we have chosen. This can cause implications when coding is a big part of the project. However, since there are hundreds of helpful documentation and guides on these specs that we have chosen, it is very easy to get an understanding of how it works. Furthermore, since each group member has different knowledge levels, we are able to help each other when implementing the application.

5.2 Schedule Risks

All of us meeting at the same time is a struggle due to our differing school and work schedules. Often only 2 or 3 teammates at a time can meet at school. It is rare for all of the teammates to be there. Our solution to this risk is to come up with a weekly meeting where everyone can participate. If we absolutely cannot do this, we are able to have a group call using Discord to discuss the project as well. In addition, if we find that time is running out, we can reduce the scope of our application and scale back on functions that are not priority 1.

5.3 Technical Risks

A technical risk we face is conforming to coding standards and best practices. This is a challenge because we all have our unique coding style, and we are also learning new technologies, so we may not yet know the best practices for each language we are using. We can avoid this risk by setting standards on how we write our code.

Another technical risk we have encountered is that the website that we are creating is showing different results when running it on the server host versus running it on the localhost. This can be easily solved by asking the CTO why this is happening.

5.4 Teamwork Risks

One of the main teamwork risks that we have is keeping track of the progress for each specific group member task. There is also a possibility that two or more members might be doing the same task while no one is doing the other task. In order to solve this risk, we can start using Trello. Trello is a useful software to show explicitly which member has which task and they are able to add info on the task to show what their progress is.

5.5 Legal/Content Risks

As of this moment, we believe that we are not in any legal/content risk when developing this project. There could be risks of a client losing their credit information when buying items, but as of now, our site does not accept any kind of payment. Also, the site has disclaimers that it is a class project to avoid any liability.

6. Project Management

We are managing our team tasks by meeting up as much as we can either in person or communicating through the messaging application Discord.

When we meet up in person, we work on a majority of the tasks together and are able to get some of them done fully. This can range from proofreading to brainstorming to getting everybody up to speed on the current milestone of the project. We also ask for information on concepts we are not clear about. Meeting together as a team allows us to complete tasks together efficiently and well.

We communicated through Discord from the very beginning, as we feel Discord is really convenient due to improved general chat and has an effective voice/video chat capability. Also, we have assigned specific channels in Discord for front-end and back-end teams, where each team can work independently. Discord also allows us to create multiple “voice channels” on the same server, so we can communicate effectively with each other when working outside of class. We also use Discord to note really essential tasks or jobs that must get done ASAP. In the future when the project size expands, we will also use Trello to manage tasks. Discord and Trello together allow us to communicate quickly and effectively to handle issues and work on tasks as a team.

For the rest of the semester, we intend to meet up in person and trying to assign as much as we can there. We will also utilize GitHub from Milestone 3 and on. The GitHub master will be working on managing all the conflicts if there are any. For the rest of the project, we will follow our pattern of meeting up, communicating with Discord on our own time, managing tasks with Trello, and utilizing GitHub to work together as a team towards our goal of creating a great software product.