



# GATOR JOE'S

Team 13

## Milestone 2

Software Engineering CSC 648-848  
Fall 2019 - Section 02

Piero Calenzani (Team Lead): [pcalenzani@gmail.com](mailto:pcalenzani@gmail.com)

Grant Kennedy (Front-end Lead)

Mark Hammond (Back-end Lead)

Chuhao 'William' Yang (Front-end, Github Master)

Varun Sura (Front-end)

Zhuoxin Tan (Back-end)

Date Submitted	Date Revised	Description
10/18/2019		First draft

# I. Functional Requirements

## Priority 1 – Must Have

- 3. Website shall have a category bar for browsing.
- 4. Website shall show the prices of listings.
  - 4.1 Website shall show the price on listing page.
  - 4.2 Website shall show the price on listing thumbnails and previews.
- 8. Users shall be able to register.
  - 8.1 Users will only be prompted to registered when attempting to buy and sell.
- 11. Registered users shall be able to buy and sell.
- 12. Registered users shall be able to log in and log out.
- 19. Administrators shall be able to deny and approve listings.

## Priority 2 – Desired

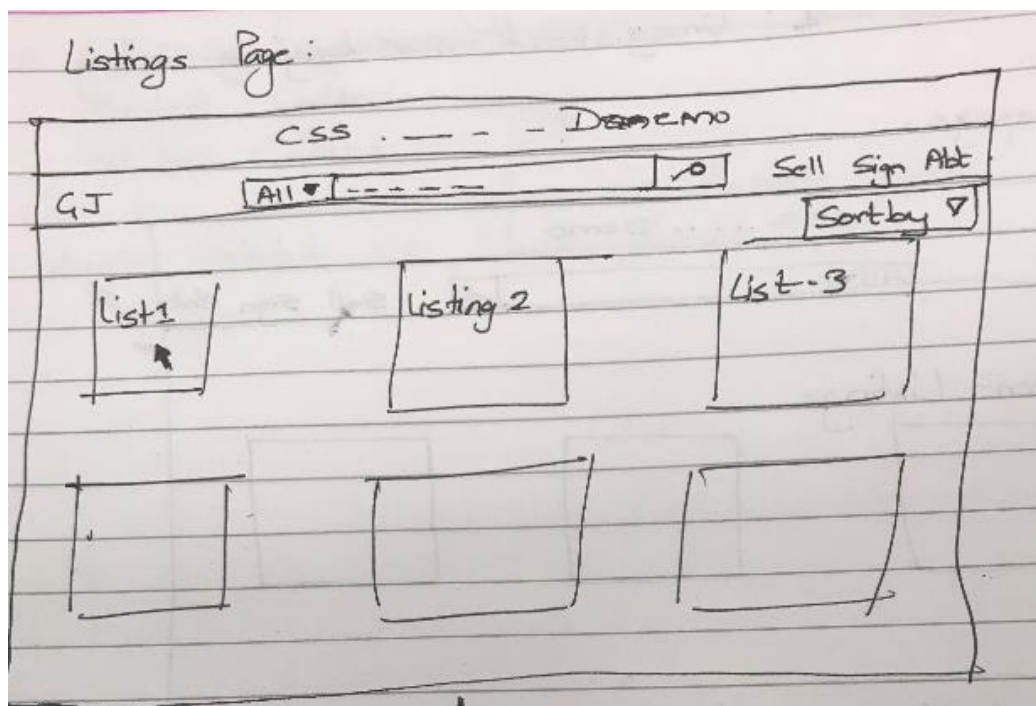
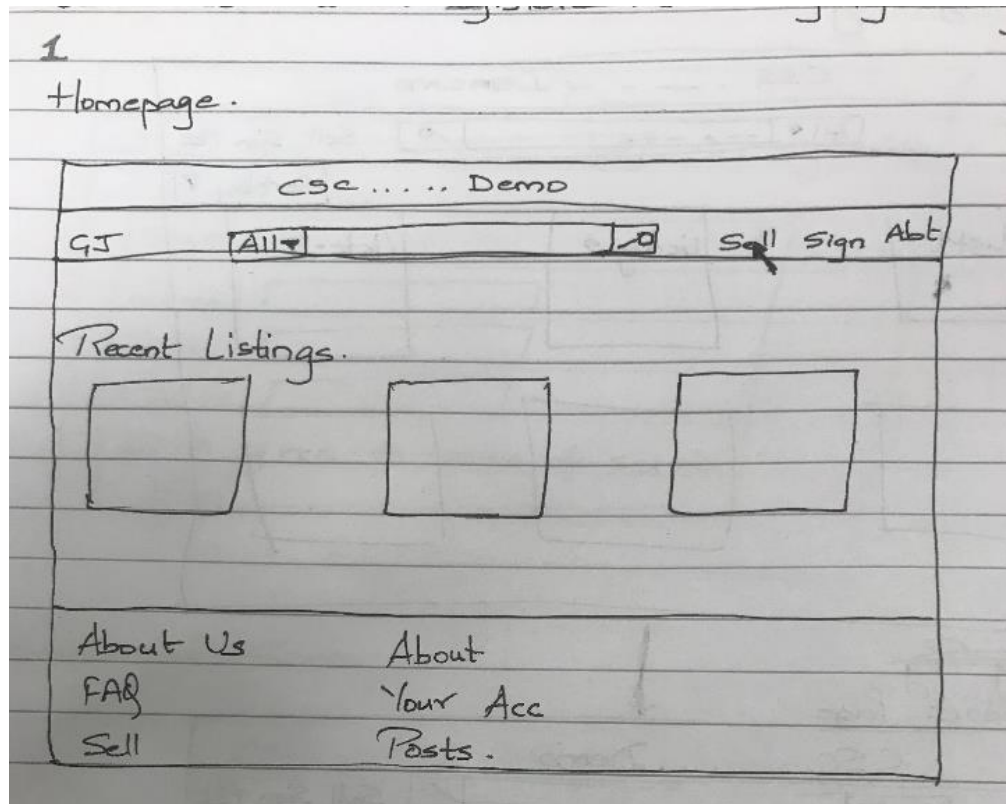
- 1. Website shall display listings on home page.
  - 1.1 Website shall display recent listings if logged out.
  - 1.2 Website shall display recent listings and user listings if logged in.
- 2. Website shall show the images of listings.
  - 2.1 Website shall show a preview image of listings when browsing.
- 5. Website shall display any description for listings.
- 6. Users shall be able to search.
  - 6.1 Users shall be able to search within a category.
- 16. Registered users shall be able to enter the information of their listings.
  - 16.1 Registered users shall be prompted for the proper information of items.

## Priority 3 – Opportunistic

- 13. Registered users shall be able to view order status.
- 14. Registered users shall be able to view order history.
- 15. Registered users shall be able to add and remove items from their shopping cart.
- 21. Registered users shall be able to view their listings.
  - 21.1 Registered users shall be able to cancel their listings.
- 18. Administrators shall be able to monitor all user listings.
- 20. Administrators shall be able to cancel orders and transactions.

## 2. UI Mockups and Storyboards

Use Case #1: Lily opens the application and searches for backpacks. She sorts by price and quickly browses through the picture of each search result until finding one she wants. After checking the listing and moving to purchase, she is prompted to register. The registration is quick and asks for a small amount of info before prompting her payment. After paying, she is given the seller's email and contacts them to agree on an exchange spot.



5 ~~Lazy Reg~~  
Product Page

CSC . . . . . Demo

GJ [All ▼] [0] Sell Sign Abt

Img	Descr, title	(\$ Buy)
Description		
- - -		
Similar Prod		

Lazy Registration

CSC . . . . . Demo

GJ [All ▼] [0] Sell Sign Abt

Sign Up X

Name:

Email:

Mobile:

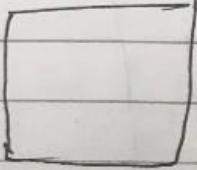
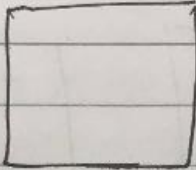
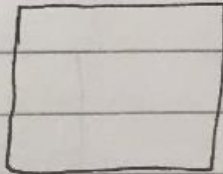
☐ I agree to terms of service

## Purchased Screen

CSC - Demo		
QT	All ▾ - - - -	10 Sell Sign Abt
<div>Img</div>	Title	Purchased
Description		Pickup date & Location

Use Case #2: Noah opens the application and navigates to the seller page where he fills out the all the information for a nightstand he is selling. He is able to upload multiple pictures showcasing his item and also lists the furniture dimensions. After finalizing his listing, he is prompted to register and promptly fills it out. He is met with a confirmation that tells him his listing is waiting for approval.

Homepage.

CSC ... Demo		
GJ	All ▾	Sell Sign Abt
Recent Listings.		
		
About Us	About	
FAQ	Your Acc	
Sell	Posts.	

Product Sale form

CSC - ... Demo	
GJ	All ▾
Select Category: ▾	
Item Name:	<input type="text"/>
Description:	<input type="text"/>
Item Price: <input type="text"/>	Add Image: <input type="button" value="Upload"/>
Item Dimensions: <input type="text"/>	
Course name if Book: <input type="text"/>	<input type="button" value="Submit"/>



## Lazy Registration

CSC - Demo		
GJ	[All ▼]	[0] Sell Sign Abt
<div>Sign Up <span>X</span></div> <div>Name: <input type="text"/></div> <div>Email: <input type="text"/></div> <div>Mobile: <input type="text"/></div> <div><input type="checkbox"/> I agree to terms of service</div> <div><input type="button" value="Sign"/></div>		

4  
~~Punch~~ Post uploaded.



CSC - Demo		
GJ	[All ▼]	[0]
<div>Img</div>	Titte	<div>Pending Appro</div>
Product description		

Use Case #3: Martha opens the application and browses under the textbook category. She fills in her class' id and section into the sorting options and quickly finds the textbook needs. She opens the listing and is prompted to register or log. She fills in her login credentials and is taken to the payment page. After paying, she is given the seller's email to decide on a meeting spot.

### CSC 648 .... Demo

GT

All

V

Q

Buy Books Sell Sign Up About

Recent Listings

About Us

FAQ

Sell

Account

### CSC 648 ... Demo

GT

All

Q

Buy Books Sell Sign Up About

Buy Books

Dept	Course	Section	Q
Account	ACCT 100	01	
Accounting	ACCT 101	02	
Accounting	ACCT 201	03	
Art			
Art History			
Astronomy			

About Us

FAQ

Sell

Martha visits Home Page,

Selects "Buy Books" from Navigation

Martha Selects her course from the Dept, Course, and Section Dropdown Menus, and enters the search



## CSC 648 ... Demo

GJ

ALL v

Q

Buy Books Sell SignUp...

Buy Books

Accounting ACCT 101 03 Q

Results

Managerial Accounting

Author: Braun

Publisher: ---

---

L

About Us

Sell

---

FAQ

Account

Martha Selects  
a listing  
from the  
search results

## CSC 648 ... Demo

GJ



Managerial Accounting

Author: Braun

Publisher: Pearson Published: 2015

Buy

Martha can  
buy this  
book from  
the descripti  
page

**CSC 648... Demo**

GJ [ALL] [Q] Login

Login

Username

Password

Steven Clicks on Log in from Navbar, and enters credentials

**CSC 648... Demo**

Administrator Portal Steven

Listings Pending

Listing Photo	Description

Approve Deny

Steven is brought to his Admin. Portal

Sees the top Pending Listing in the Queue.

Can read info, check each photo, and Approve.


Or Deny

Steven is brought to his Admin. Portal

Can read info,  
check each  
photo, and  
Approve.

Or Deny

Administrator Portal

Steven 

Listing Rejected


Leave a Comment

Submit

If Deny,  
Steven is prompted  
to leave a  
feedback comment  
explaining why  
the post was  
rejected.

After Submitted,  
Steven is directed  
back to the other  
Pending Listings.

Administrator Portal

Steven 

Listings Pending

You're All Caught Up  
No Pending Listings.

When all Listings  
have been  
reviewed,  
Steven is  
dismissed by  
this page.

### 3. High Level Architecture, Database Organization

The organization of our database will be focused on two primary tables, with multiple secondary tables acting in support. The first primary table, RegisteredUsers, will contain a int field for the primary key ID and string fields for the login, password (which will be encrypted), name, and email. Administrators will be part of a separate Admin table. The primary key for the admin table will be a foreign key connected to the RegisteredUsers ID column. The second primary table will be the Listing table. The Listing table will contain int fields for the listing ID, number of items sold, and the foreign keys for the Category\_ID and Seller\_ID (from RegisteredUsers). There will be string fields for the title and description. A Boolean entry for whether or not an administrator has approved the listing will be included. Finally, a photo column for the listing will be included as a binary blob. Category will be its own table. It will only contain an ID and category name. There are two support tables containing addition information for listings. Both will have their primary IDs be a foreign key from the Listings table. The Book table will contain columns for author, class, section, and any addition prudent information for selling textbooks. The Furniture table will contain a dimensions column. Finally, the Order table will contain the have columns for three foreign keys (in addition to its own ID primary key). There will be a column for the Listing.ID, the buyer's RegisteredUser.ID, and the seller's RegisteredUser.ID.

Search will primarily be handled by SQL queries and %like functions. The terms from the database that will be searched come from the columns Listing.Title, Listing.Description, and all Book column outside of the ID column. We will need to create APIs for creating new users and creating new listings.

#### RegisteredUsers

- ID: int
- Password: string
- Name: string
- Email: string

#### Admin

- Admin\_ID: int (foreign key: RegisteredUsers.ID)

#### Category

- Category\_ID: int
- Category\_Name: string

#### Listing

- ID: int
- Title: string
- Description: string
- Category\_ID: int (foreign key: Category.Category\_ID)
- Sold: int
- Approved: boolean
- Seller\_ID: int (foreign key: RegisteredUsers.ID)
- Photo: binary blob

## Book

ID: int (foreign key: Listing.ID)

Author: string

Title: string

Class: string

Section: int

ISBN: int

## Furniture

ID: int (foreign key: Listing.ID)

Dimensions (string)

## Order

ID: int

Listing: int (foreign key: Listing.ID)

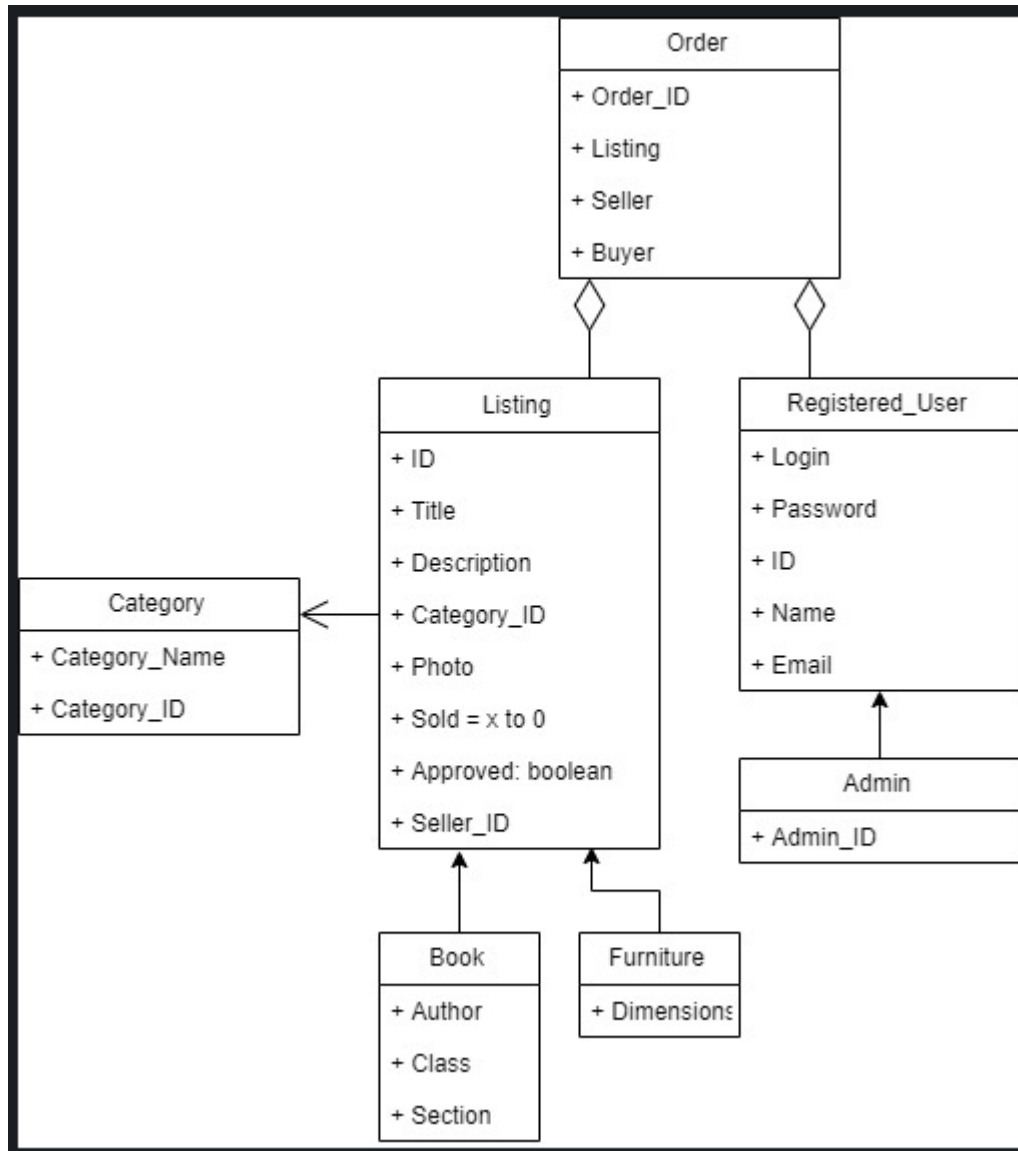
Seller: int (foreign key: RegisteredUser.ID)

Buyer: int (foreign key: RegisteredUser.ID)

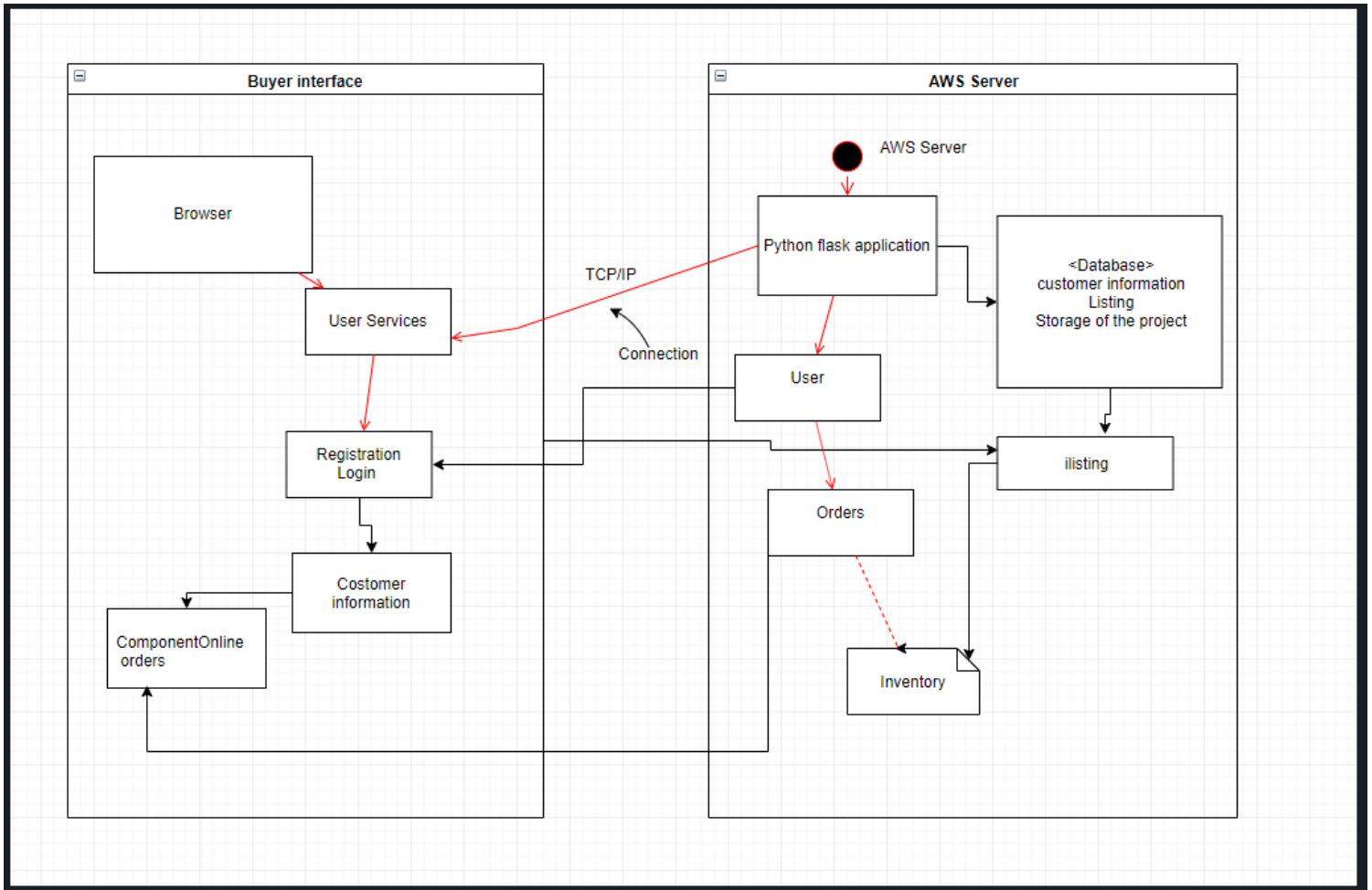


## 4. High Level UML Diagrams

UML Class Diagram:



## UML Component and Deployment Diagram:



## 5. Key Risks for Project

### Skill Risk

Currently, our most prominent risk is that most of the team is learning or re-learning the languages used for the project. Between Bootstrap, MySQL and Python Flask, our team members are not fully familiar with all three, which means a lot of learning is necessary along with project development. This issue can be resolved by taking careful looks through tutorial and documentation of each language. In addition, code reviews will help inexperienced members learn from the more experienced ones within a particular language.

### Schedule Risk

So far, we have committed to a few extra features in our project that extend the original scope of the application. Examples are functionalities like shopping cart, order history and furniture dimensions. As a safe way of managing expectations, those functionalities have been made as priority 2 or 3 to keep the focus on the primary functions of the application before extending ourselves to opportunistic features.

### Teamwork/Technical Risk

Some team members do not have a lot of experience working on projects as a group. A potential risk to our application is dealing with consistency and uniformity when combining everyone's parts together. Code reviews help with the understanding of each other's code but for merging it together, extensive testing and careful git merging will be required. In our current setup, front and back-end teams will have separate branches to commit too in order to avoid overwriting work.

## 6. Project Management

This team is managed by Piero, team lead, through Slack and Trello. Slack is used for announcements, checking progress and all general communication among the team. The slack workspace set up for the project has separate chat channels for front-end, back-end, git requests, and announcements. Regular check-ins are done through Slack to ensure that team is progressing and on track for what the current milestone may be. Trello is used as a clear way to display and keep track of all relevant tasks and their present priorities. Any work assigned to team members will be communicated in person, announced on Slack and reflected on our Trello dashboard.

Project work is assigned to either the back-end or front-end leads who are responsible to split up the work with the corresponding members of their sub-team. All major milestones and deadlines will have repository branches corresponding to the back and front teams to avoid any code discrepancies while they work on their assigned tasks. Team lead and back-end lead are responsible for any issues with the webserver itself.