

# SW Engineering CSC648/848 Spring 2022

## *PurpleMarket*

### TEAM 8

Name	Role	Email
Mark Kim	Team Lead	mkim22@mail.sfsu.edu
Khushboo Gandhi	Front End Lead	
Cody Huang	Back End Lead	
Ernesto Diaz	GitHub Master	
Jesus Cervantes Fajardo		
Vivian Kuang		
Jiasheng Li		

### Milestone 2

March 14, 2022

Milestone/Version	Date
M2V1	03/11/2022

# Contents

<b>I</b>	<b>Executive Summary</b>	<b>2</b>
<b>II</b>	<b>Main Data Items and Entities</b>	<b>2</b>
<b>III</b>	<b>List of Functional Requirements</b>	<b>4</b>
<b>IV</b>	<b>List of Non-Functional Requirements</b>	<b>5</b>
<b>V</b>	<b>Competitive Analysis</b>	<b>6</b>
<b>VI</b>	<b>System Architecture</b>	<b>7</b>
<b>VII</b>	<b>Team &amp; Roles</b>	<b>8</b>

## Section I: Executive Summary

PurpleMarket is a web application that will connect students, staff, and faculty from San Francisco State together via a virtual marketplace. In this marketplace, they will be able to communicate with each other, as well as buy and sell products on a user to user basis. There will be no intermediate and no surprises. What one user posts will be verified and will be exactly what is sold to another user. With this in mind we hope to create a safer, open community at San Francisco State, while also providing our users with a more convenient way to shop. The value of PurpleMarket is in its ability to provide an experience custom tailored to people associated with San Francisco State. Our vision is to provide a marketplace that facilitate the ability to sell and purchase the types of items and/or services that are most important to our users.

PurpleMarket plans to be easy to navigate, and easy on the eyes. There isn't any clutter which makes the site much more presentable to our users. Our users are presented with a clean design that will make it much easier for users to navigate our site to find exactly what they are looking for. If they have any trouble at all finding their specific product, they are able to narrow their search in three ways. The first method to finding their desired product is by browsing different categories. By providing categories that our users are most interested in, our users will be able to more easily sell or find the products that they want. The second method will be a simple and easy to use search that would allow the user to type specific titles, names, or descriptions where the user will be presented with the results of the search. Specific to the academic environment, the final method for finding a product is by finding products that are associated with a particular course. Once a user finds what they are looking for they can sign up and create a PurpleMarket account that will allow them to communicate with the user that listed the product directly on our site and the rest is history. So what about listing products? Any verified user who opens an account will be able to list a product. Finally, PurpleMarket will be able to facilitate on-campus meetings between buyer and seller, which simplifies the transaction experience.

Our team of passionate San Francisco State students know the frustrations of navigating online classifieds and the unique needs of the people associated with the school. It is because of this that we are exceptionally equipped to understand the San Francisco State community and provide them with the experience that they desire.

## Section II: Main Data Items and Entities

**user:** The users are the people expected to use the web application. This entity will contain all the pertinent information that describes our user like their name, email address, and other identifying information.

1. user\_id: Primary key user identification
2. user\_first\_name: First name of user
3. user\_last\_name: Last name of user
4. user\_email: User's email

**item:** This entity are the items that are being bought, sold, or provided. This entity contains all the information that helps users find what they want. For example, this entity will have a name, a description, price, a category that it belongs to, and photos so that they can see what the item looks like.

1. item\_id: Primary key item identification
2. item\_name: Name of item
3. item\_desc: Description of item
4. item\_price: Asking price for item
5. item\_pic: Picture of item
6. item\_category: Category of item (e.g. furniture, books, etc)
7. item\_seller\_id: Id of the seller of the item

**category:** This is the category a particular item will belong to. For example, items may be furniture, books, electronics, etc. We want this data item to be easily changed or expanded if necessary in case our users find categories that they want.

1. cat\_id: Primary key for category
2. cat\_name: Name of the category (e.g. furniture, books, etc)

**account:** This will contain all the account information of our users as well as describe their permissions. This entity will contain the user's account password, the account type (e.g. Admin, Registered User, etc), and the permissions that are associated with the account type.

1. act\_id: Primary key account identification
2. act\_user\_id: User that this account is associated with
3. act\_username: Username for account
4. act\_pass: Password for this account
5. act\_role: Role associated with this account (e.g. admin, registered user, etc)

**location:** This will contain different locations that can be used on-campus where transactions can be completed. This is meant to provide an easy way for buyers and sellers to coordinate a meeting to complete a transaction.

1. loc\_id: Primary key location identifier
2. loc\_name: Name of location

3. loc\_coord: Coordinates of location

**message:** This entity exists to contain all the messages that are sent from one user to another. This will contain not only the message, but also the sender, recipient, and a time stamp of when the message was sent and received/read.

1. msg\_id: Primary key message identifier
2. msg\_item\_id: Item that the message is associated with
3. msg\_sender\_id: Account ID for sender
4. msg\_rec\_id: Account ID for recipient
5. msg\_body: Message body
6. msg\_meet\_loc\_id: Proposed location for meeting
7. msg\_meet\_date: Proposed meeting date
8. msg\_meet\_time: Proposed meeting time
9. msg\_time: Timestamp of message

**review:** This data item contains the reviews for all users. This entity will contain the review itself, the recipient of the review as well as the person who wrote the review, the time stamp of the review, and also the associated transaction/item that goes with the review.

1. rev\_id: Primary key review identifier
2. rev\_item\_id: Item that the review is associated with
3. rev\_reviewer\_id: Account ID for reviewer
4. rev\_reviewed\_id: Account ID for user being reviewed
5. rev\_time: Timestamp of review
6. rev\_rating: Rating given to user being reviewed (out of 5)
7. rev\_body: Review text body

## Section III: List of Functional Requirements

### *Unregistered User*

1. Unregistered users shall be able to search the site for certain posts.
2. Unregistered users shall be able to filter search results.
3. Unregistered users shall be able to register for an account.

### *Registered User*

6. Registered users shall be able to do everything an unregistered user can do.
7. Registered users shall be able to create posts.

8. Registered users shall be able to get in contact with the poster/seller.
9. Registered users shall be able to edit and correct and/or delete their own post after it was submitted.
10. Registered users shall be able to report violations of terms of service of another user.
11. Registered users shall be able to login to their account.
12. Registered users shall be able to logout of their account.
13. Registered users shall be able to request/reset their password if they forgot.
14. Registered users shall be able to leave reviews of other users regarding completed transactions.
15. Registered users shall be able to propose meeting locations or modify location proposals to complete transactions.
16. Registered users shall be able to inspect their transaction history.

#### ***Admin***

17. Admins shall be able to do everything a registered user can do.
18. Admins shall be able to delete and edit posts done by any user.
19. Admins shall be able to contact any user.
20. Admins shall be able to ban users.
21. Admins shall be required to approve posts before they can go live.
22. Admins shall be able to inspect any user's transaction history.

## **Section IV: List of Non-Functional Requirements**

1. Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0
2. Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers
3. All or selected application functions must render well on mobile devices
4. Data shall be stored in the database on the team's deployment server.
5. No more than 50 concurrent users shall be accessing the application at any time
6. Privacy of users shall be protected
7. The language used shall be English (no localization needed)
8. Application shall be very easy to use and intuitive
9. Application should follow established architecture patterns
10. Application code and its repository shall be easy to inspect and maintain

11. Google analytics shall be used
12. No e-mail clients shall be allowed. Interested users can only message to sellers via in-site messaging.  
One round of messaging (from user to seller) is enough for this application
13. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.
14. Site security: basic best practices shall be applied (as covered in the class) for main data items
15. Media formats shall be standard as used in the market today
16. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development
17. The application UI (WWW and mobile) shall prominently display the following exact text on all pages *"SFSU Software Engineering Project CSC 648-848, Spring 2022. For Demonstration Only"* at the top of the WWW page nav bar. (Important so as to not confuse this with a real application).

## Section V: Competitive Analysis

Specification	PurpleMarket	Amazon	Ebay	Craigslist
Short time to make post	-	+	++	+
Easy to contact seller	++	+	+	++
Seller associated with SFSU	++	-	-	-
Ability to meet seller in person	++	-	+	+
Propose on-campus meeting location	++	-	-	-

++ = Specification is Outstanding

+ = Specification is Good

- = Specification is Bad

Our web app has numerous planned advantages compared to our competitors. We make sure that all users of our app are SFSU students, staff, or faculty. We make sure that buyers can easily contact the seller. Another advantage is that our seller and buyer will be able to meet in person, in a safe and public environment, whether it be on campus or otherwise. We also have a site administrator that guarantees the validity of each item posted and swiftly approves each post or rejects inappropriate posts.

## Section VI: System Architecture

Server Host:	Amazon EC2 2vCPU 4GB Ram
Operating System:	Amazon Linux 2
Database:	MySQL v8.0.19
Web Server:	Nginx v1.19
Server-Side Language:	JavaScript/Node.js 10.19.0
Web Framework:	React.js 16.14.0
IDE:	VSCode
Web Analytics:	Google Analytics



## Section VII: Team & Roles

Name	Role
Ernesto Diaz	GitHub Master/Front End Dev
Jesus Cervantes Fajardo	Back End Dev
Khushboo Gandhi	Front End Lead
Cody Huang	Back End Lead
Mark Kim	Team Lead/Document Master
Vivian Kuang	Back End Dev
Jiasheng Li	Front End Dev