SW Engineering CSC648/848 Section 01
Summer 2017

Jizhou Yang- Team Lead- jyang15@mail.sfsu.edu
Rafsan Saadi- CTO
Raghav Parti
Takahiro Odaka
Brandon Yu
Zach Melamed

Team # 3

Stock Photo Marketplace
Milestone 2
7/13/2017

Document Revisions

Submission Date	Feedback given	Feedback Implemented
6/30/17	7/1/17	7/6/17
7/13/17		

Table of Contents:

I. Executive Summary

We are proud to release our website, “stock marketplace”, for online media management. The website offers a platform for image store, search, and sale. We have provide beautiful user interface, secure media storage, powerful search engine, and friendly user environment. Users can browse, search and purchase the media. Contributors (usually artists) can upload, manage and price their media.

The potential market for the website is very promising. More and more people are seeking online media storage, management, and sale. Several companies have already provided online media service. Compared to our competitors, we provide easier downloading, faster search, popular product trending, and download product review. These features will bring good market share to our product.

The application is deployed on Amazon Web Service (AWS), provided LAMP stack environment that consists of the following components: Linux, Apache, MySQL, PHP. The Dev framework and APIs include: CakePHP, Bootstrap, jQuery, Google analytics. Supported browser matrix includes: Google Chrome and Mozila Firefox.

The website is developed by a group of passionate students in computer science at San Francisco State University. The team has 6 members: Brandon (back end developer), Jizhou (team lead), Rafsan (CTO), Raghav (front end developer), Takahiro (back end developer), and Zach (front/back end developer).

II Use Cases v2:

Unregistered user – Johnny visits the website by going on his laptop. He's new to the website, so he can browse the website and search for media content, but *won't have the same privileges* as registered users. He can search for media using *categories, type of media, as well as any matching free text* on the description. In addition, Johnny will be able to view a *low-resolution thumbnail* of the image or video he is interested in buying. Johnny finds a picture that he would like to use as a desktop wallpaper, but is *prompted to register* on the website before being able to purchase media of his choice.

Registered user (Buyer) – Alice finds a video she would like to purchase on the website. Alice is *prompted to register*, via an email address, user ID, password, and full name to the website in order to purchase media content. Alice likes spending time on the website, and is more active so she decides to register to the website through the *registration page*. In addition, Alice is given *more privileges* than unregistered users. Alice will be able to *message contributors* to set up buying of media. Anytime Alice wants to purchase media, she will have to log in through the login page. If needed, Alice is able to *flag* inappropriate or undesirable content on the website for an admin to see.

Registered user (Contributor) – Bob is a freelance photographer and video editor. He wishes to *sell media content on the website*. After *registering* to the website, Bob is able to add an appropriate *description/caption, set price point, and include various details about his media*. Bob is able to interact with website UI easily to promote more uploading of media and have an overall larger stock of media content available on the website. Bob is given *full recognition of the media* he uploads, and allowed to manage his own media content freely. Payment will be left for Bob and the *buyer to decide through website messaging*, where he will be notified about the buyer info, and which of his product is being bought.

Administrator – Jay is an admin who *monitors the website* and manages the media database to ensure a better user experience for customers. Jay has the privilege *to remove media content, edit descriptions, manipulate media categories, and remove users*. He will be in charge of maintaining the content on the website using *MySQL workbench*. Jay should keep a log of any server or database issues for ease of website maintenance.

III <u>Data Definition v2:</u>

• **Unregistered User (Guest)**: Users who do not have an account, and can only browse the site without being able to post items for sale or buy items.

• **Registered User:** Users who have an account in the website, and can log into it the site using their user credentials. Registered Users can be both buyers and contributors (sellers). The following data is needed in order to register in the website:

- First Name
- Last Name
- User Email
- User Id
- Password

User Account/Profile dentals should contain the following:

- Messages:
  - Inbox
    - From (Sortable)
    - Item name (Sortable)
    - Message content
  - Compose
    - To (Auto Populated)
    - From (Auto Populated)
    - Subject: Item name (Auto Populated)
    - Message Body
    - Send and Cancel Option
  - Item List (Owner)
    - Item listing (Name, Media Type, Category, and Thumbnail)
    - Data Operation (CRUD)
  - Purchase History
    - Item listing (Name, Media Type, Category, and Thumbnail)
    - Data Operation (RD*)

• **Buyer (Registered User):** User who has an account in the system, and is buying an item. Buyer can contact the Contributor / Seller to request addition information about specific item.

• **Contributor/Seller (Registered User):** User who has an account in the system, and is selling an item.

• **Administrator:** Users that have special privileges, and have the ability to remove posts from the site, remove items, issue warnings and bans from the site, and generally enforce the Code of Conduct for the site. Administrators also are responsible for helping users when needed.

• **Items/Services:** Any object which is for sale, or for purchase on the site. Our website will sell only Photos and videos and items will have the following Metadata:

- Title
- Description
- Category
- Media Type
- Price
- Thumbnail
- Owner

IV. Functional Requirements V2:

Type of User		Priority
Non-registered User	1. Non-registered Users shall be able to search for photos and videos by using a keyword or category. 2. Non-registered Users shall be able to browse photos and videos. 3. Non-registered Users of the website shall be eligible to become Registered Users of the website. 4. Non-registered Users shall be allowed to view previews of videos on the website. 5. Non-registered Users shall be allowed to view previews of photos of the website. 6. Non-registered Users shall be able to view thumbnails of the images	1 1 2 1 1 1
Registered User	1. Registered Users of the website shall be allowed the functions of Non-registered Users, along with the following: a. Registered Users shall be able to purchase photos and videos.	2
Contributor	1. Contributors of the website shall be allowed the functions of Non-Registered Users, along with the following: a. Contributors shall be able to upload photos and videos to the website.	2

	b. Contributors of the website shall be able to set a “price tag” for their uploaded media. If a Contributor chooses to, they are allowed to put up media for free.	2
Admin	1. Admins of the website shall be allowed the functions of Non-registered Users, along with the following: a. Admins shall be able to view the list of Registered Users and Contributors on the website. b. Admins shall be able to maintain the list of Registered Users and Contributors. c. Admins shall be able to remove content that is flagged as “inappropriate”.	3 3 3

V Non-Functional Requirements V2

1. Application shall be developed using class provided LAMP stack

2. Application shall be developed using pre-approved set of SW development and collaborative tools provided in the class. Any other tools or frameworks must be explicitly approved by Anthony Souza on a case by case basis.

3. Application shall be hosted and deployed on Amazon Web Services as specified in the class

4. Application shall be optimized for standard desktop/laptop browsers, and must render correctly on the two latest versions of all major browsers: Mozilla (Latest), Safari (Latest), and Chrome (Latest).

5. Application shall have responsive UI code so it can be adequately rendered on mobile devices but no mobile native app is to be developed

6. Data shall be stored in the MySQL database on the class server in the team's account

7. Application shall be deployed from the team's account on AWS

8. No more than 50 concurrent users shall be accessing the application at any time

9. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.

10. The language used shall be English.

11. Application shall be very easy to use and intuitive. No prior training shall be required to use the website.

12. Google analytics shall be added

13. Messaging between users shall be done only by class approved methods and not via e-mail clients in order to avoid issues of security with e-mail services.

14. Pay functionality (how to pay for goods and services) shall not be implemented.

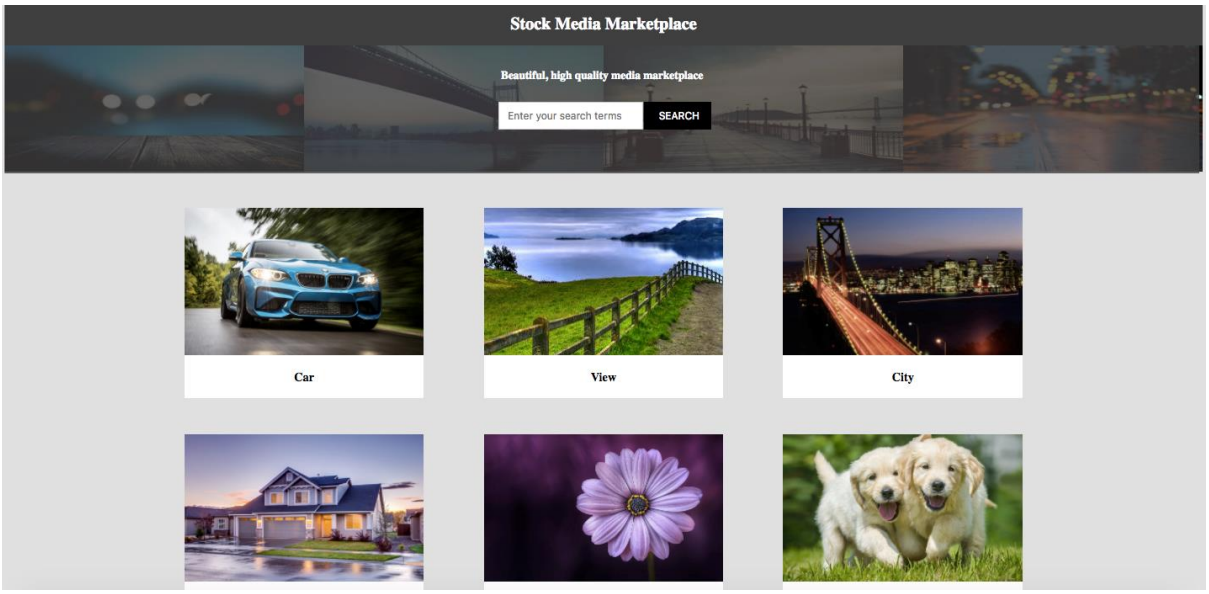
15. Site security: basic best practices shall be applied (as covered in the class)

16. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development

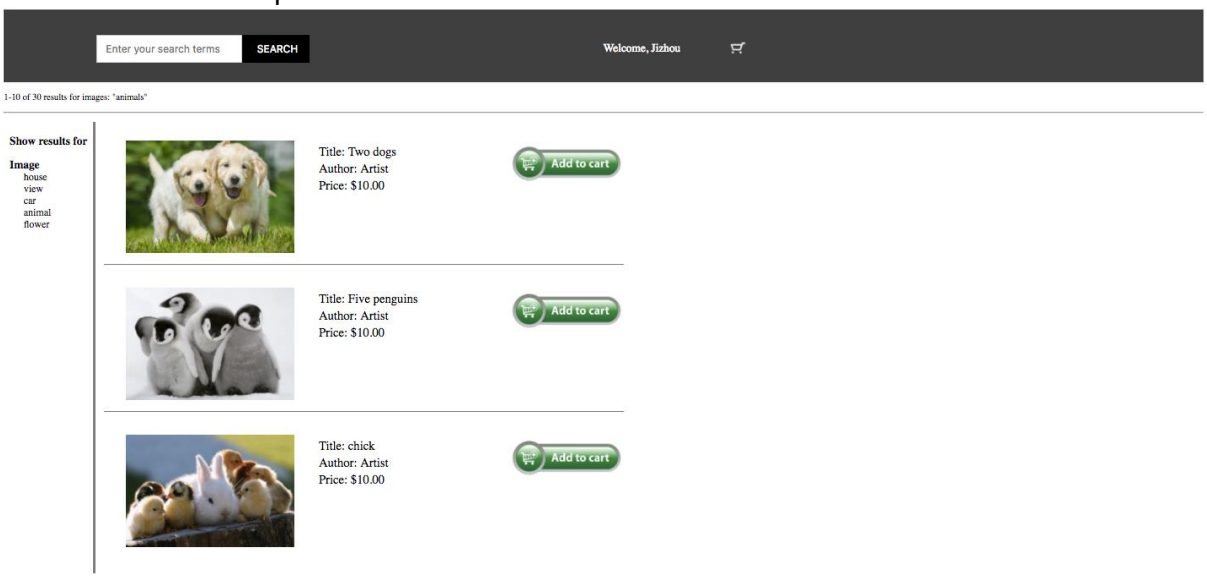
17. The website shall prominently display the following text on all pages "SFSU Software Engineering Project, Spring 2017. For Demonstration Only". (Important so as to not confuse this with a real application).

IV. UI Mockups and Storyboards (high level only)

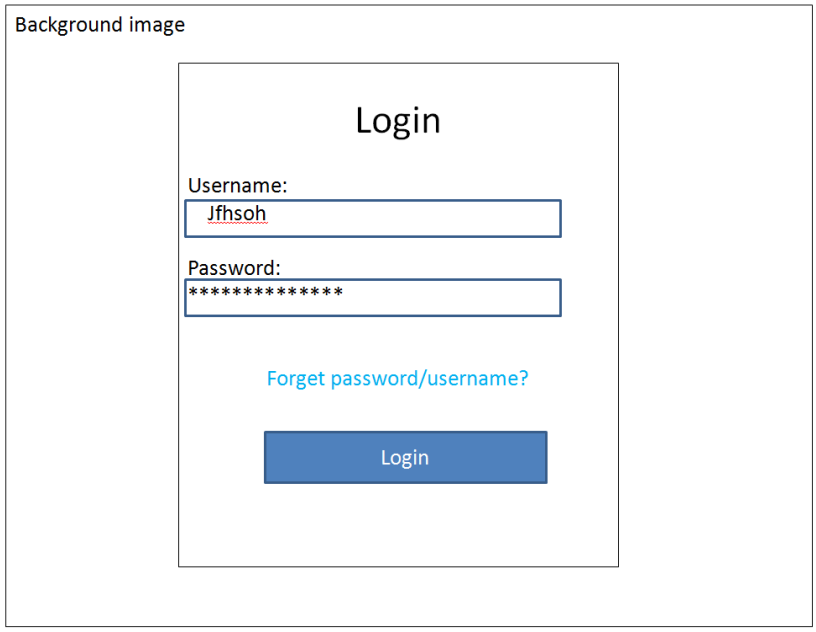
Homepage Version 1 mockup:



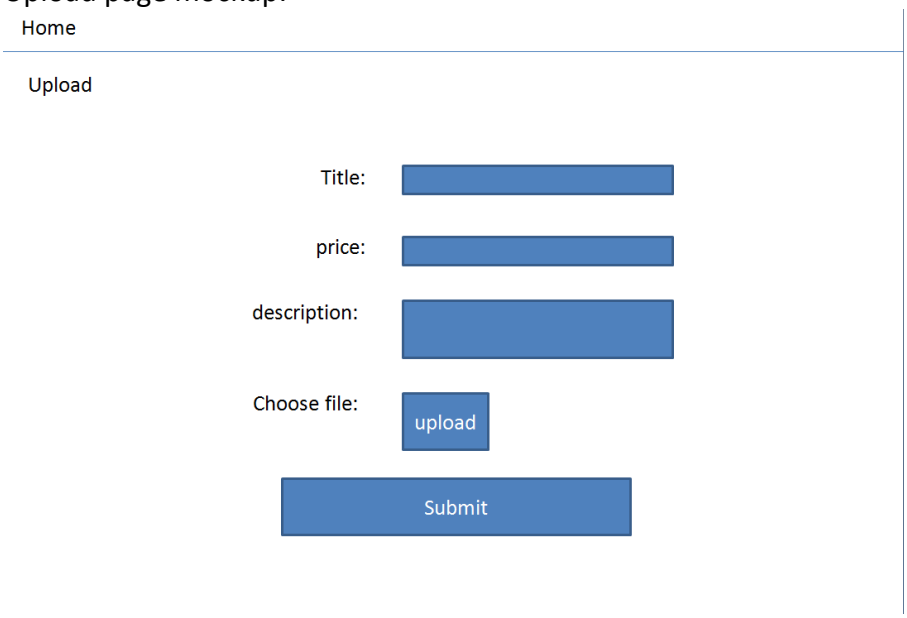
Search Result Mockup:



Login page mockup:



Upload page mockup:



Registration page mockup

Home

Registration

Email:

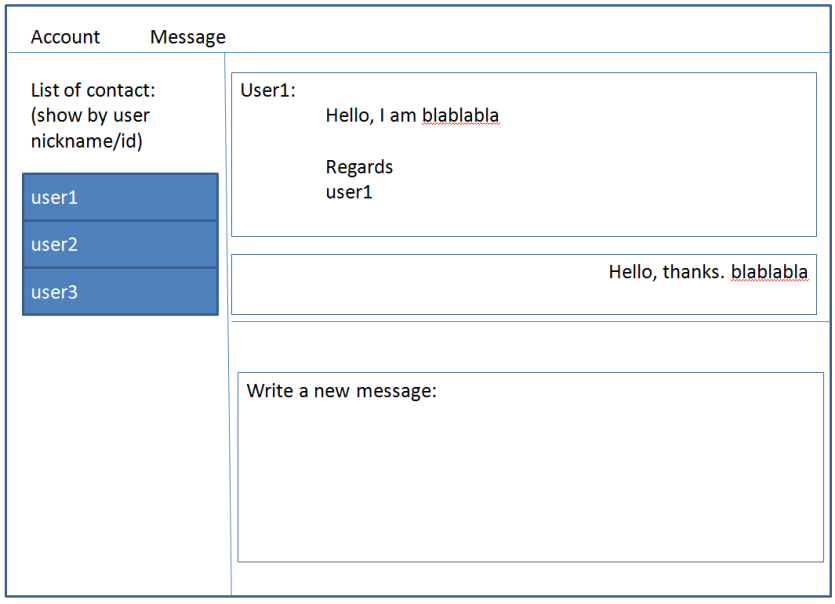
password:

username:

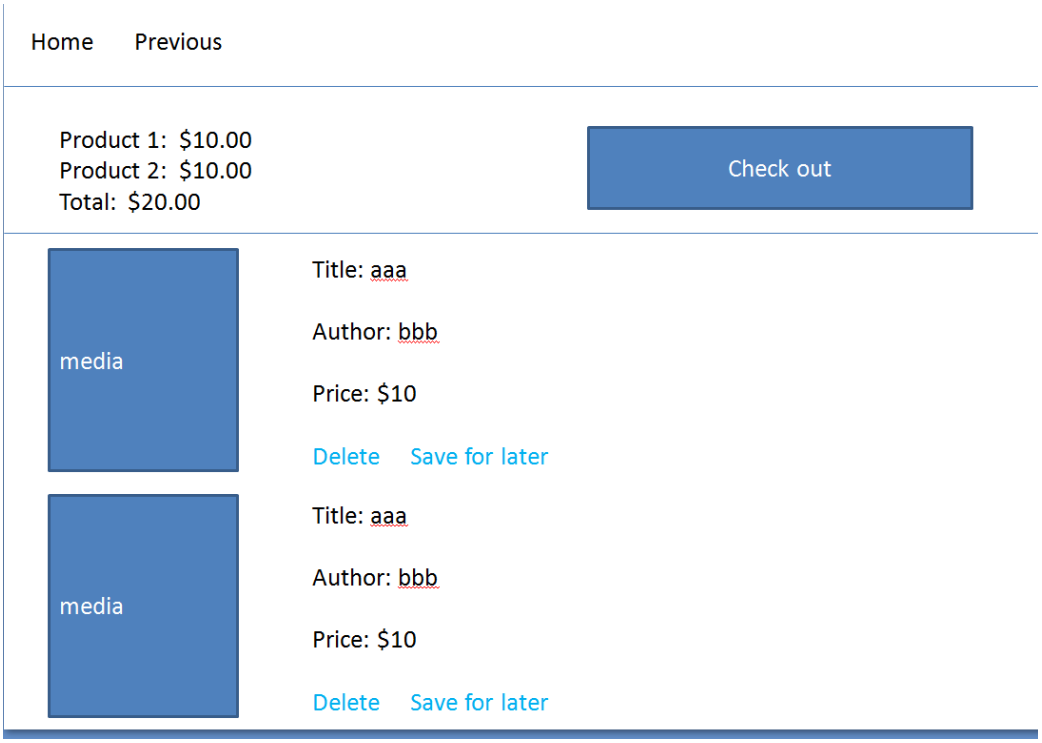
Security question:

Submit

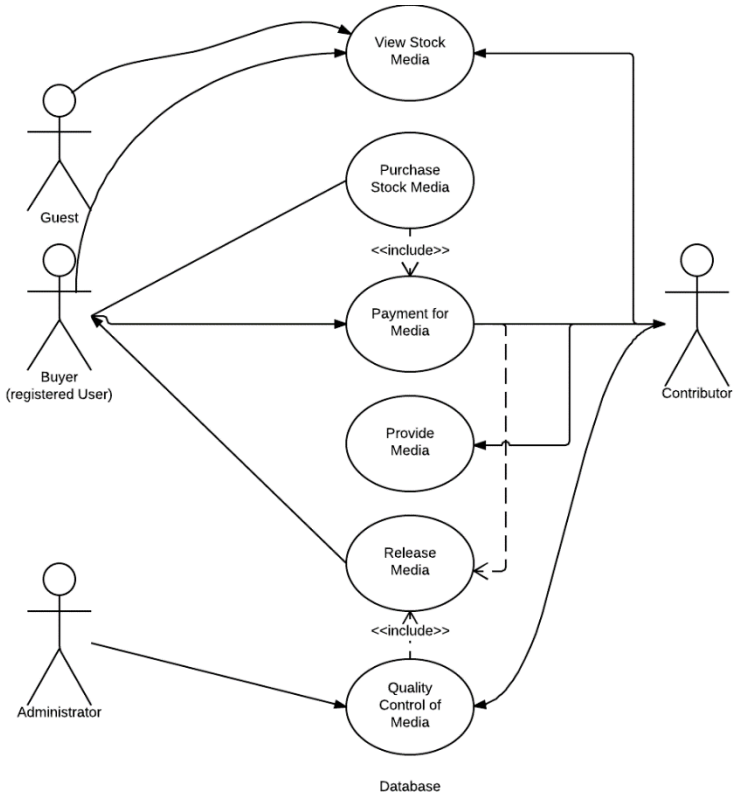
Messaging inbox mockup:

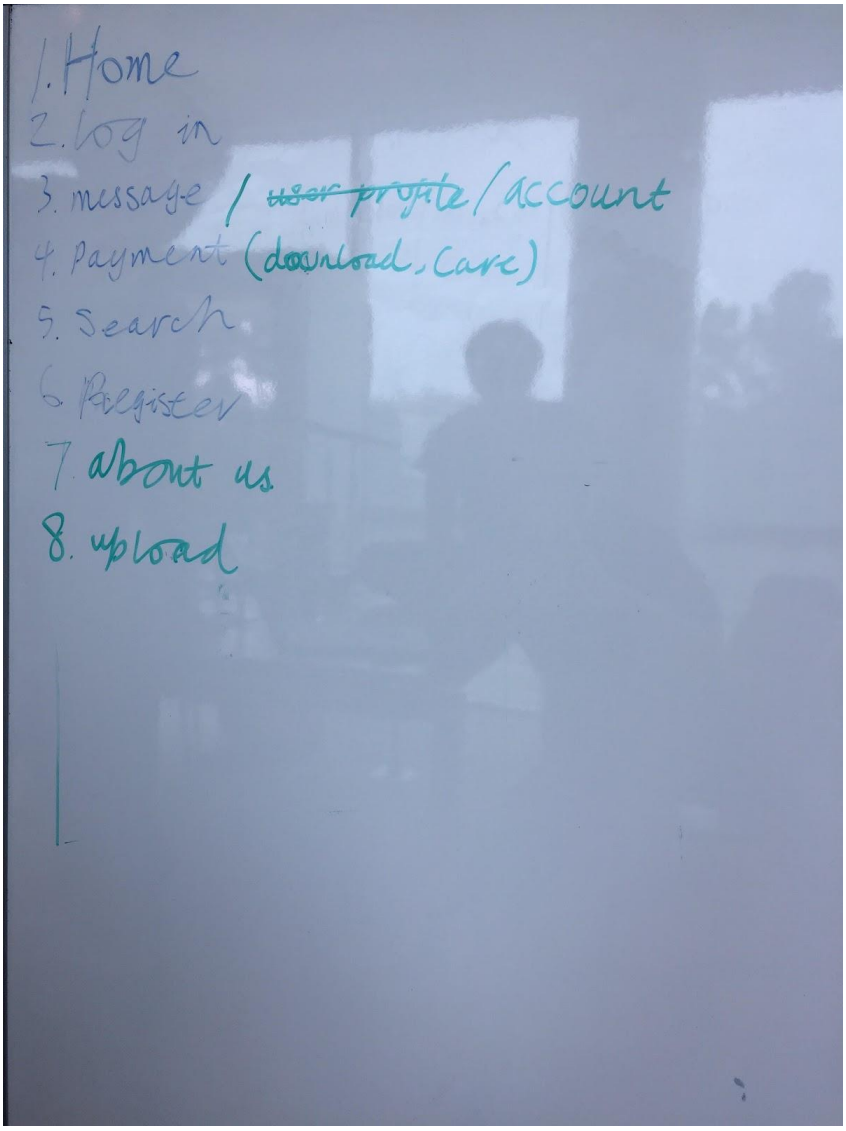


Checkout page mockup:

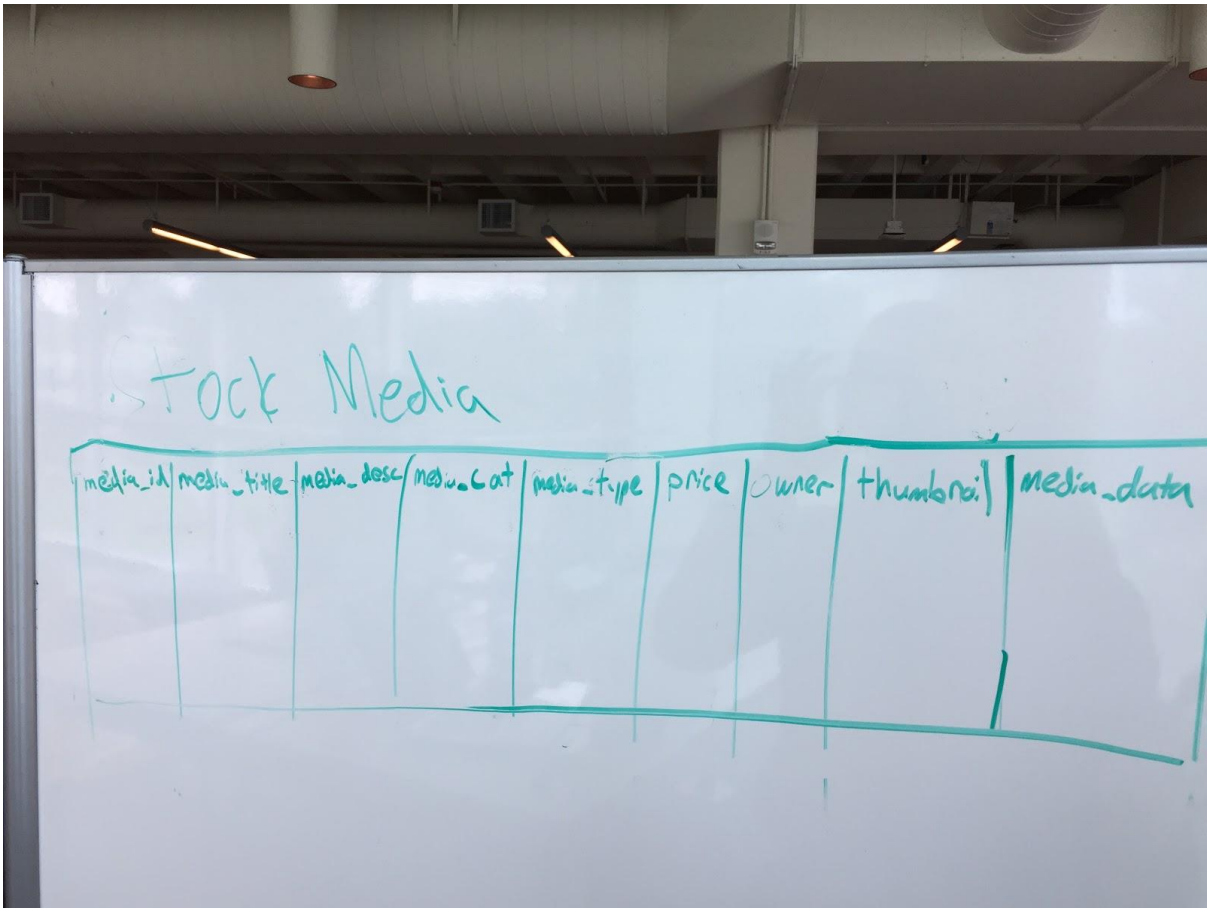


Use class mockup diagram



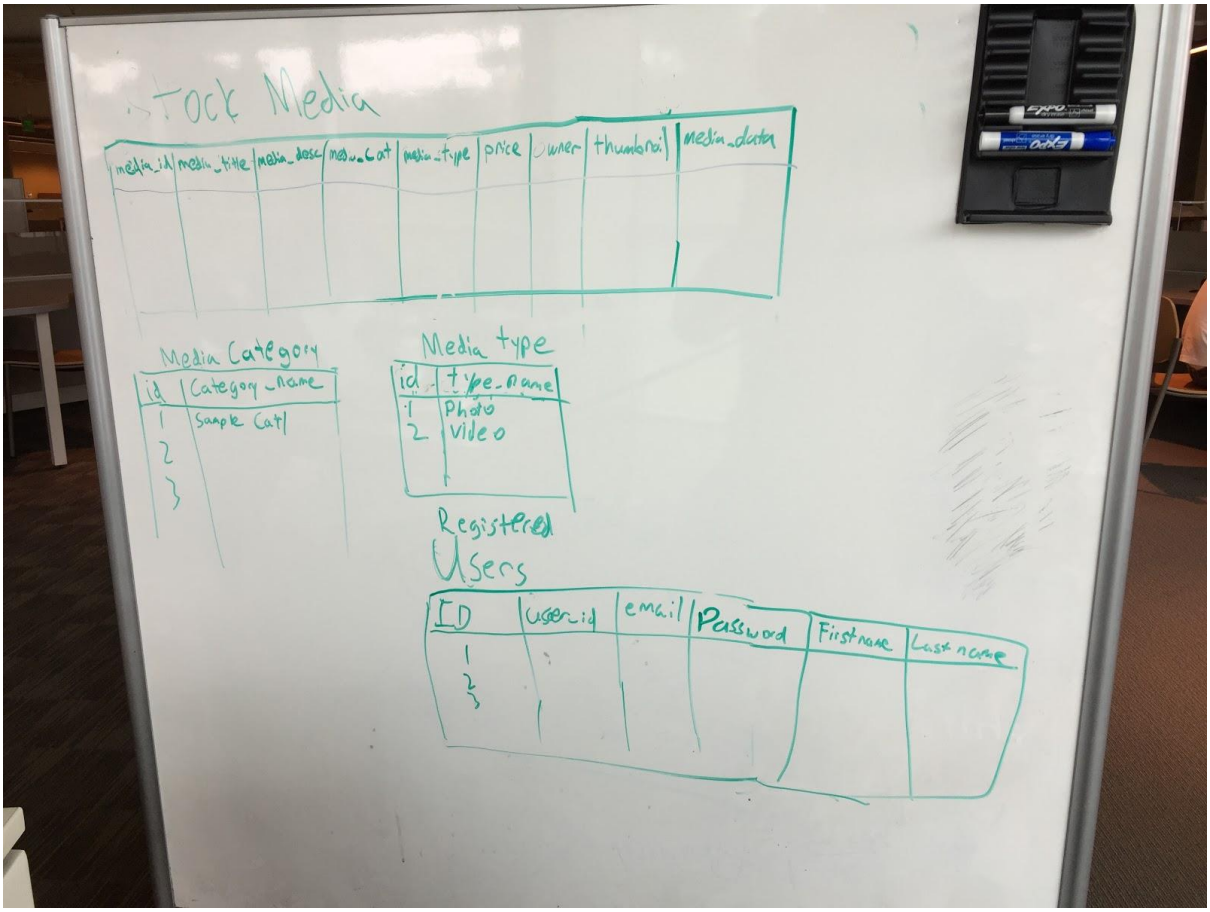


Storyboard mockup with our side navbar will allow users quick and easy functionality moving between different pages of the website. Certain features will be greyed out until logged in with a registered account, but most will be available without the need for login.

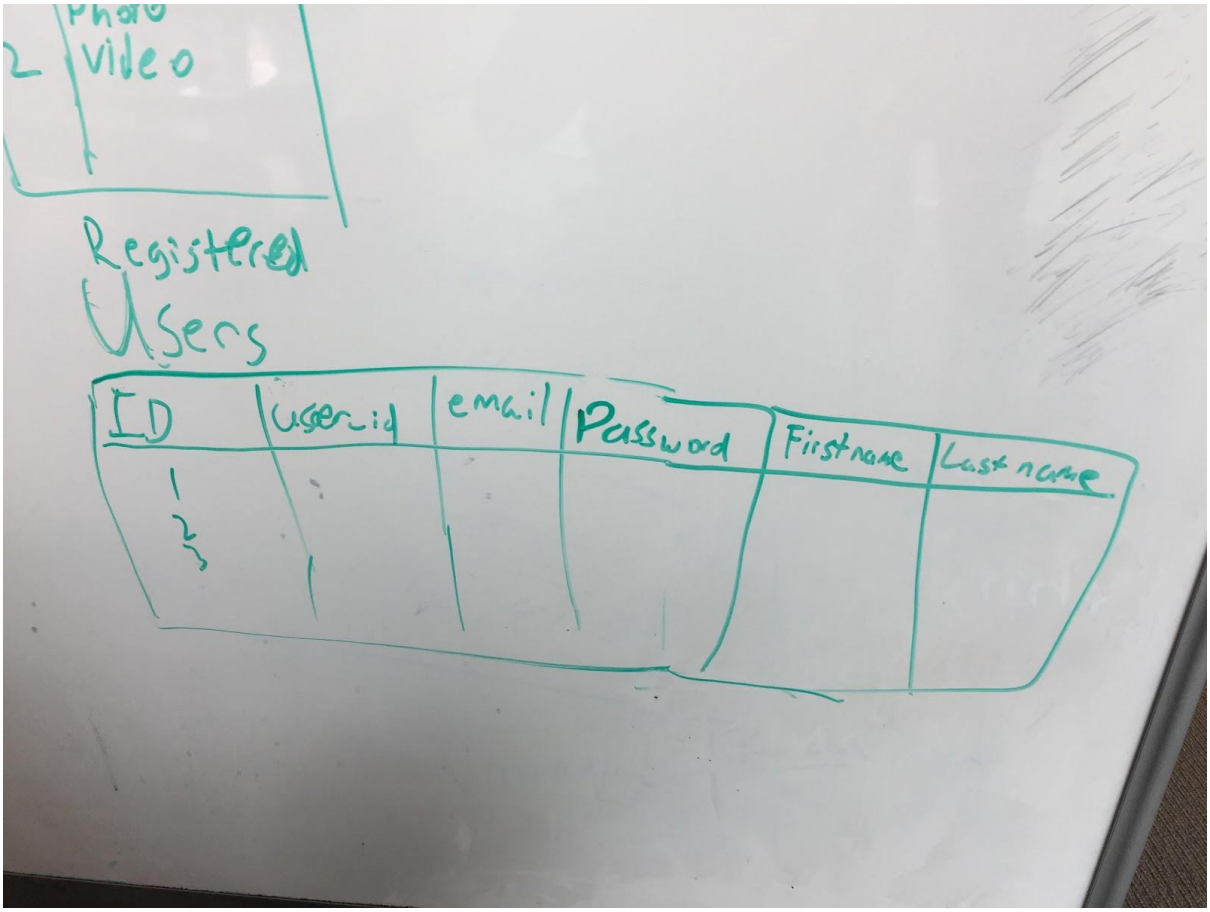


Storyboard Mockup showing how our stock media will be stored in our database with the following entries:

- media_id
- media_title
- media_desc
- media_cat
- media_type
- price
- owner
- thumbnail
- media_data

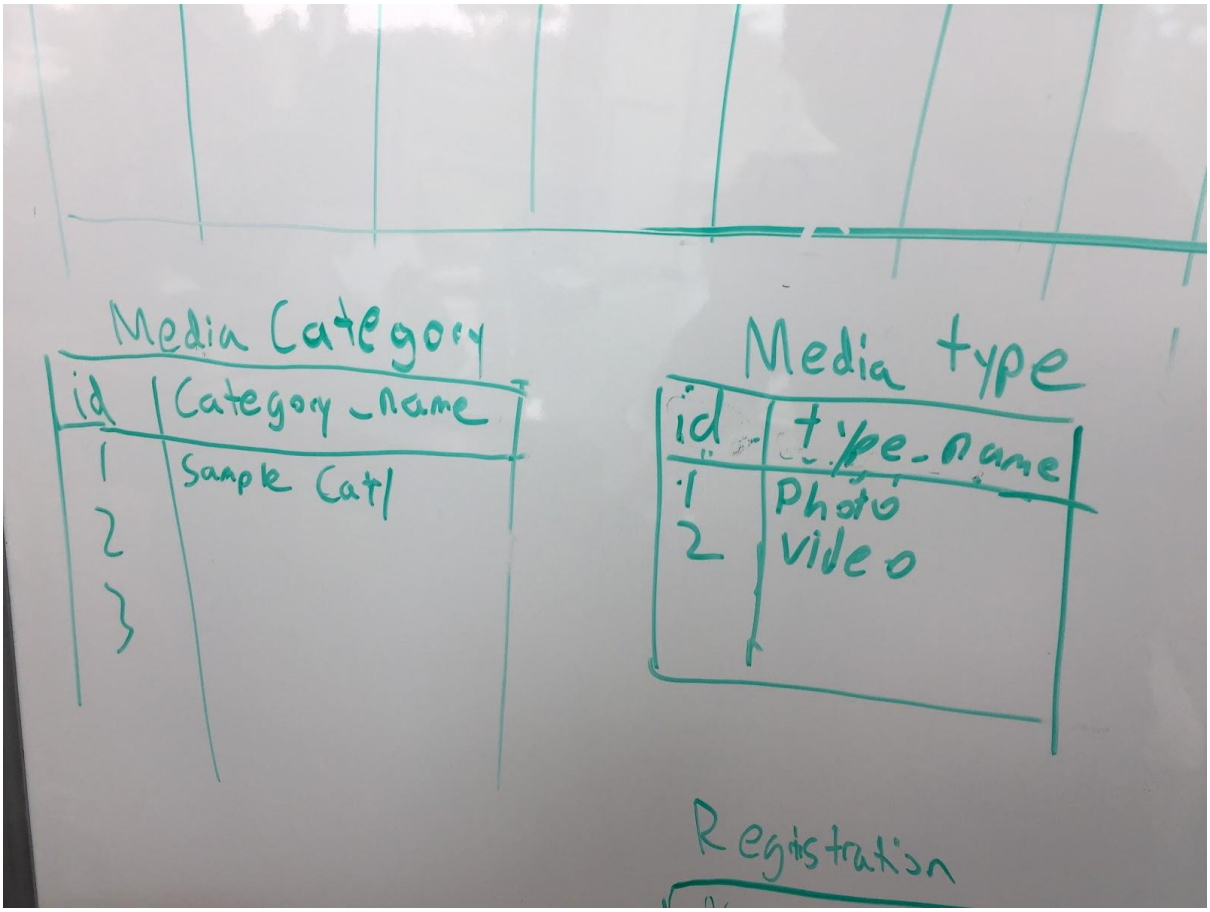


Overall database mockup, details show in other pictures.

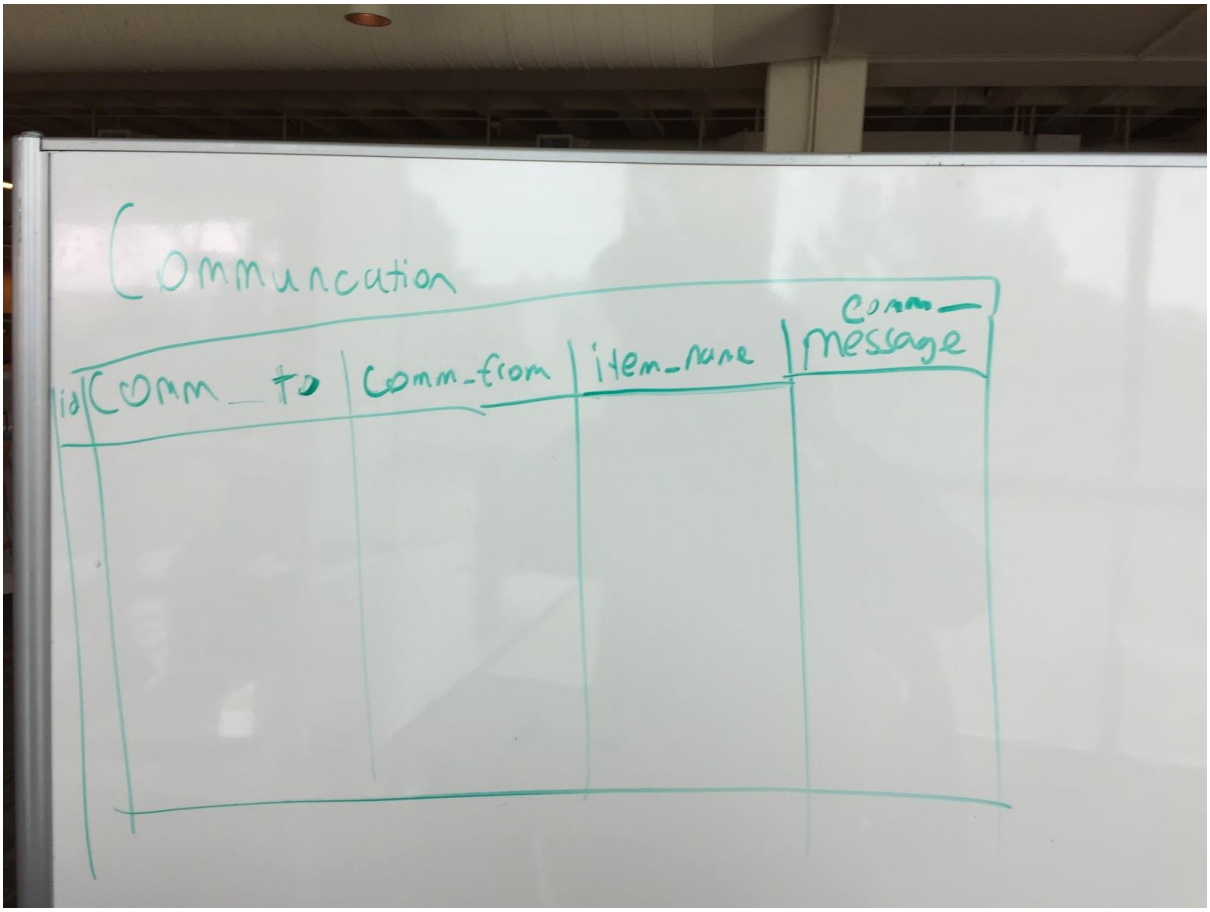


User id database mockup containing:

user_id, email, password(encrypted), firstname, and lastname



Database mockup for storying different types of media, both photos and video, along with categories for all media types.

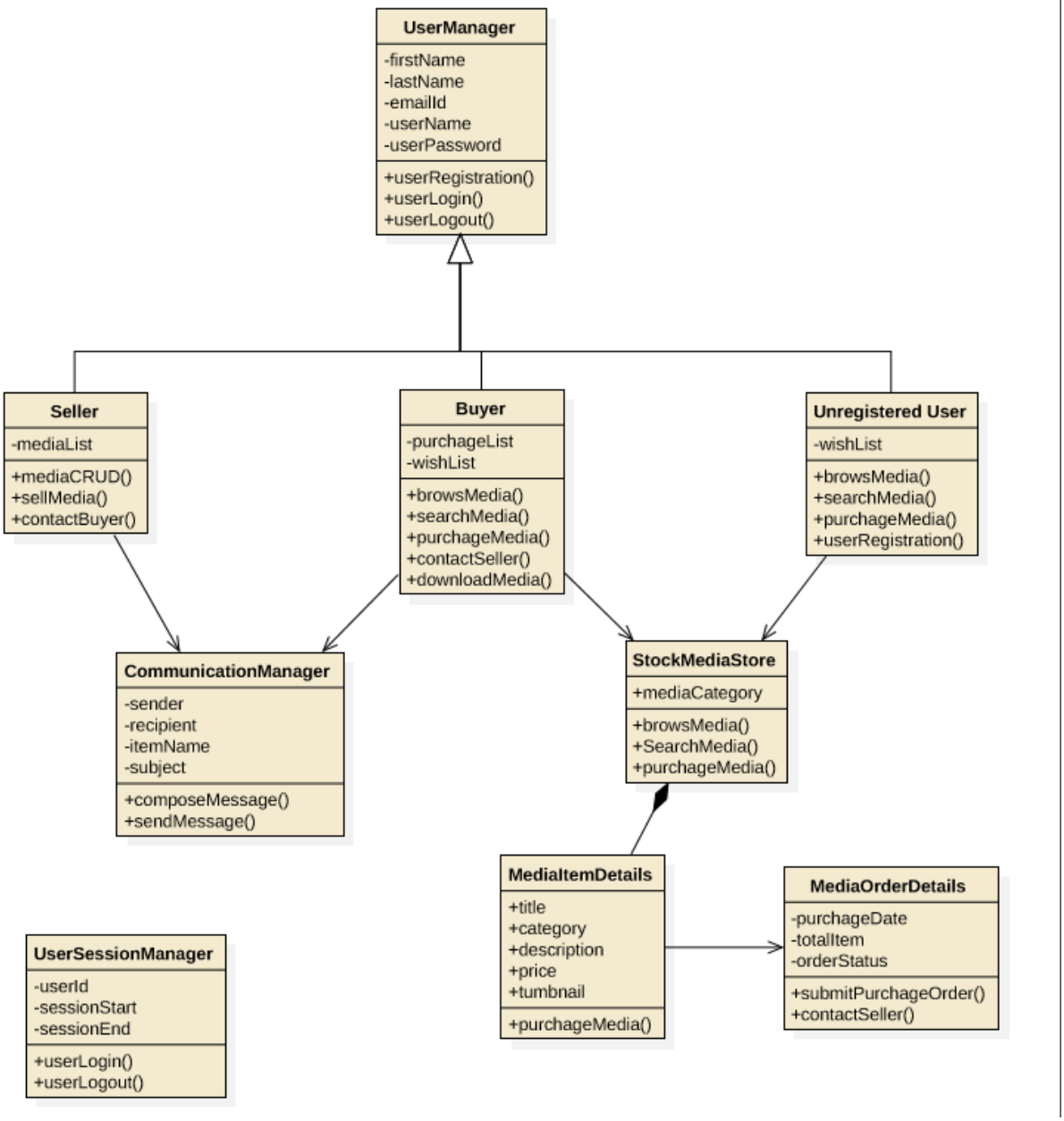


Database table mockup for messaging to and from buyers and contributors.

VI. High level Architecture, Database Organization

1. Primary Dev Component: Our application will be developed provided LAMP stack environment that consists of the following components:
    a. Linux
        i. We are using Ubuntu Linux as our primary platform for both application development and deployment
    b. Apache
        i. We are using Apache web server to allow users to access our application in the web
    c. MySQL
        a) User data will be stored and managed using MySQL database system.
        3. PHP
        a) PHP is the scripting language that allows creating web application with dynamic content.
        b) Our application backend will be developed using PHP.
2. Development Framework and APIs: We will be using the following framework and components which will allow rapid PHP application development along with front and backend solutions for our team application:
    a. CakePHP
        a) CakePHP is an open-source web, rapid development framework that makes building web applications simpler, faster, and require less code.
        b) Our application will be developed using CakePHP by utilizing their rapid development framework
    b. Bootstrap
        a) A web development framework that allow us to start on frontend development quickly by providing a series of templates and themes.
        b) Allows to design responsive web pages for different screen sizes easily.
    c. JQuery
        a) JavaScript library that eases cross-browser development.
        b) Helps to speed up frontend development by providing wrappers for many Java Script functionalities.
    d. Google analytics
        a) Our application will be integrated with Google Analytics Solutions which offers free and enterprise analytics tools to measure website, app, digital and offline data to gain customer insights.
3. Application Deployment:
    a. Application will be deployed on Amazon Web Services (AWS), the cloud-computing platform.
    b. Supported Browser Matrix:
        1. Following browsers must be used for faster and smoother use experience with our application:
            i. Google Chrome- Latest
            ii. Mozilla Firefox- Latest
    c. Other Tools:
        1. MySQL workbench will be used for database maintenance.

VII. High Level UML Diagrams



IX. Key Risks

- **skills risks (do you have the right skills):** We may have some difficulty with PHP, as we are not familiar with it.
    - team get more familiar with sql and php
- **schedule risks (can you make it given what you committed and the resources):**Dealing with unfamiliar language and framework, difficult to predict how much time it takes to implement the functions.
- **technical risks (any technical unknowns to solve):** We may have difficulties implementing the message function.
    - looking into discussion forums and other online tools
- **teamwork risks (any issues related to teamwork):** It is a challenge to work with people from different cultures
    - more open communication in person and via online tools,
- **legal/content risks (can you obtain content/SW you need legally with proper licensing, copyright):** No foreseeable legal or content risks.

X Team Roles

Outline key roles of team members

Team Member	Roles
Jizhou Yang	Team Lead
Rafsan Saadi	CTO
Raghav Parti	Front End Developer
Takahrio Odaka	Back End Developer
Brandon Yu	Back End Developer
Zach Melamed	Front/Back End Developer