SW Engineering CSC 648/868

Section 01 Summer 2017

Milestone 2

Picturesque

Team 5

Tiffany Ku, Team Lead | tyku@mail.sfsu.edu

Di Mao (Andy), Tech Lead | dmao1@mail.sfsu.edu

Haotian Zhang (Ivan)

Teng Yu

Andrew Cheng

Cody Jackson, Data Engineer | cjackso5@mail.sfsu.edu

Calvin Ip | ip@mail.sfsu.edu

Last Modified
July 1, 2017
July 14, 2017

1. Use Cases V2

- Registered/Unregistered User

A general visitor, John, who plans to purchase some media for their project is able to view low quality videos and pictures. In order to purchase the products, he would have to provide certain information to become a **registered user**, which allows him the rights to make purchases. John can **search** for specific media such as videos or pictures. John needs to enter **keywords** such as the category he wishes to look for or a description of the media. Once John finds the product he likes, when he wishes to make a purchase, he will be prompted to login or register. He will be able to buy from the **seller** once he registers on the site and provides his information, such as his full name and billing info, to be able to complete his purchase.

- Admin/Author

John has finished his project and would like to upload his own pictures so others can enjoy his work and possibly purchase them. The **admin** sees that John has uploaded an inappropriate photo and prevents the photo from being uploaded. John then receives a message stating why his photo was not uploaded and is prompted to change his photo. **Authors** that wish to publish inappropriate content will be advised by the admin to change their content, otherwise their account will terminate and they will revert back to a general visitor.

- Buying

All users can view media that is available to be sold. After viewing the media, the user may either buy the product or save it in a checkout cart for later. Users that wish to purchase the products right away will be required to login or register for an account. The now registered user can now purchase the products. An email will then be sent to the author of the product stating a user wishes to purchase their product.Users that add products to their cart will be able to continue browsing. Once the user wishes to checkout everything in their cart, they shall login or register before continuing. Afterwards, the registered user can purchase the products and an email will be sent to the authors of the products.

- Seller

Sellers may upload their products for sale. They must first have an account or register for an account to be able to sell their products. The sellers upload their products which must have a title, genre, and a price that the registered user wishes to sell the product for. Once the seller has confirmed their information is correct the new product will be submitted and available for users to view and purchase.

2. Data Definition V2

➢ **General Visitor**: can browse public photos with low resolution, cannot download them and cannot add any comments. Does not need to register/login.

➢ **Registered User**: can access all photos (excluding private photos) with high resolution, can add comments and can add into own list. Need to register/login.

➢ **Approved User**: can access all photos explicitly approved by the author, , can add comments and can add into own list. Need to register/login and apply for author's approvement.

➢ **Seller**: person who is responsible for the selling of an author's content. It's need copyright of the selling photos.

➢ **Author**: person who uploads photos and owns copyrights on the site. Need set item public or private and deal user's approvement request. Need to register/login.

➢ **Admin**: can access all data and modify the database. Need to register/login.

➢ **Public photos**: can be browsed by anybody with low resolution and can be access by registered user with high resolution.

➢ **Item**: price, title, description, tags(location, event...etc), keywords(for search), comments and photo.

➢ **Author credit**: can achieve by post photos and can be used to purchase photos or get discount.

➢ **User credit**: can achieve by purchasing items and can be used to purchase photos or get discount.

➢ **Watch list**: user can add favourite author into list in order to easily watch their posts ASAP.

➢ **Browse record**: record those images which user browse, used to recommend user's preferred contents.

➢ **Purchase history**: record all information(price, date, author...etc) of item which user purchased.

➢ **Thumbnails**: for users to quickly browse images with reduced-size version. Used to help in recognizing and organizing items, serving the same role for images as a normal text index does for words. In the age of digital images, visual search engines and image-organizing programs normally use thumbnails, as do most modern operating systems or desktop environments, such as Microsoft Windows, Mac OS X, KDE (Linux) and GNOME (Linux).

3. Functional Requirements V2

1. (Priority 1) Media Search: All users shall be able to search for the products they are looking for
2. (Priority 1) User Registration: All users shall be able to register a new account or log in to existing account
3. (Priority 1) All users shall be able to purchase a product (as guest)
4. (Priority 1) All guests shall provide email address and password to create an account
5. (Priority 1) Messaging feature: Registered users shall be able to leave message to product's owner
6. (Priority 1) Registration form: Required for users to register. Contains name, e-mail, username, password. Stored in the database.
7. (Priority 1) Items for sale form: required for users to post an item for sale: Contains title, price, description, and preview image. Stored in the database.
8. (Priority 1) Registered users shall have access to post new product
9. (Priority 1) Registered users shall be able to delete/modify their item listings
10. (Priority 1) Administrators shall be able to visit, browse, and search the site.
11. (Priority 1) Administrators shall have access to all users posts
12. (Priority 1) Administrators shall be able to delete posts
13. (Priority 1) Administrators shall be able to delete accounts
14. (Priority 1) Sellers/Authors shall have access to the photos and videos they own and are selling

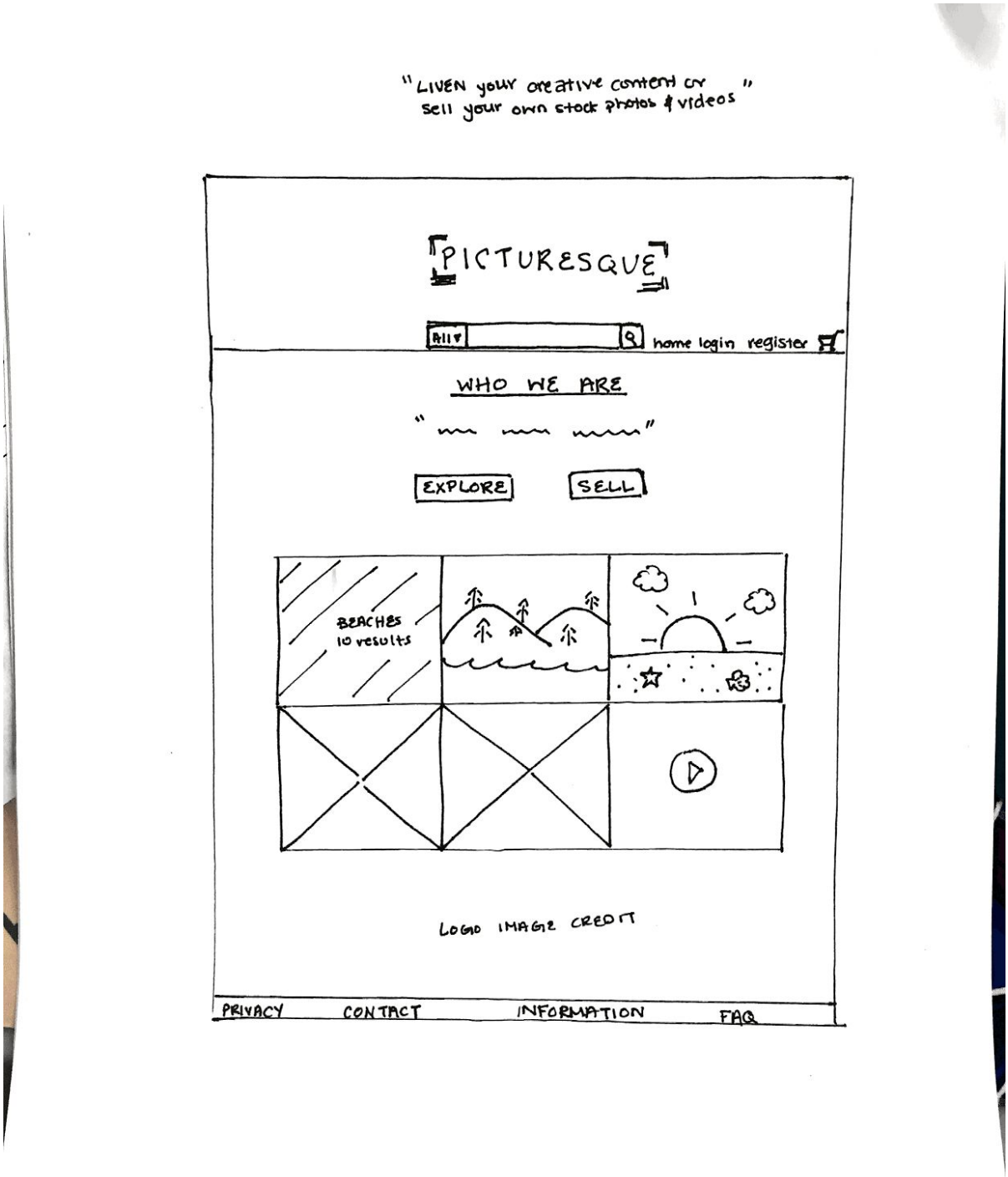
15. (Priority 2) Registered users shall be able to view their purchase history
16. (Priority 2) Sellers/Authors shall be able to delete/modify the price, description, and title of their items

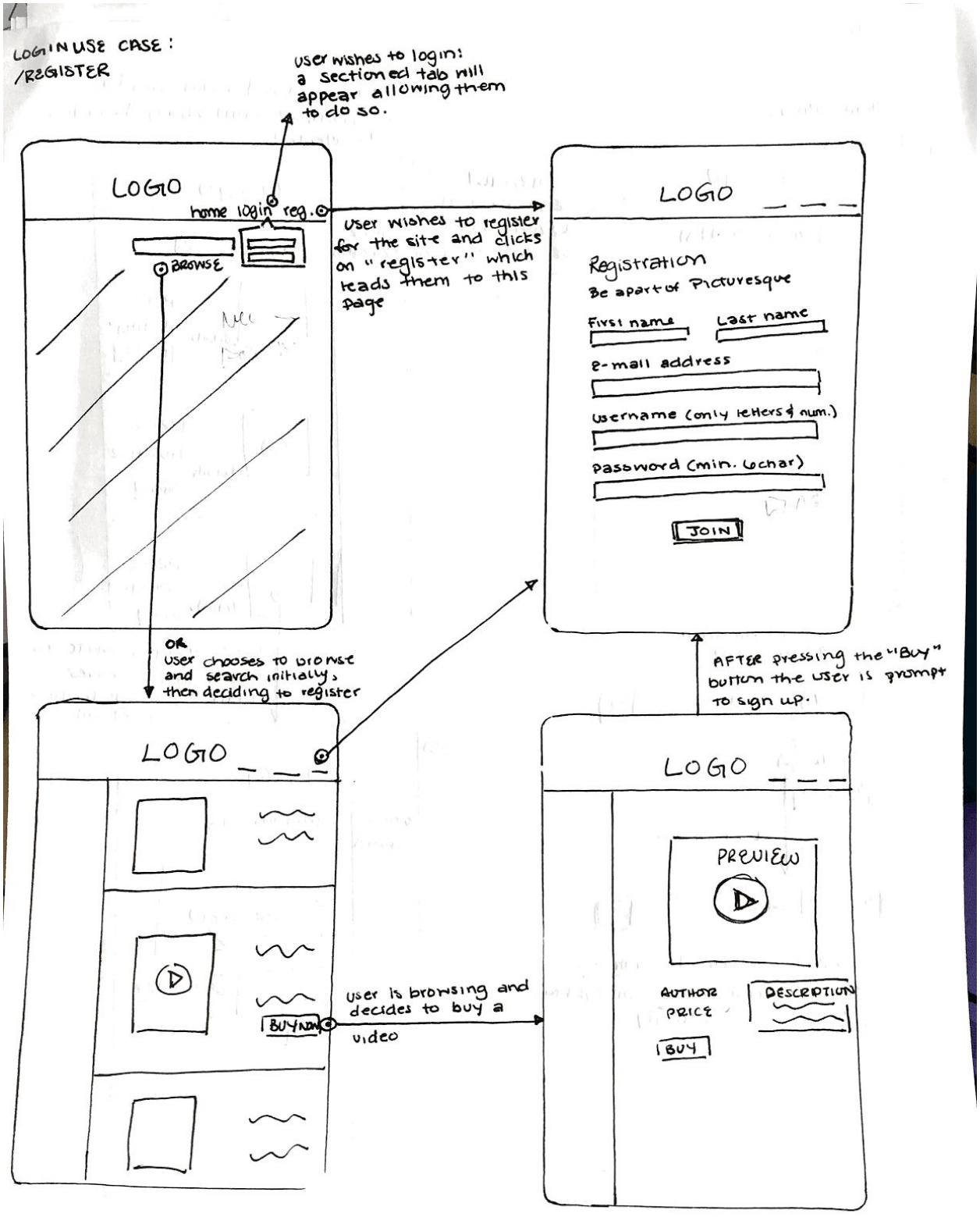
17. (Priority 3) Registered users shall be able to rate and comment on content

4. Non Functional Requirements V2

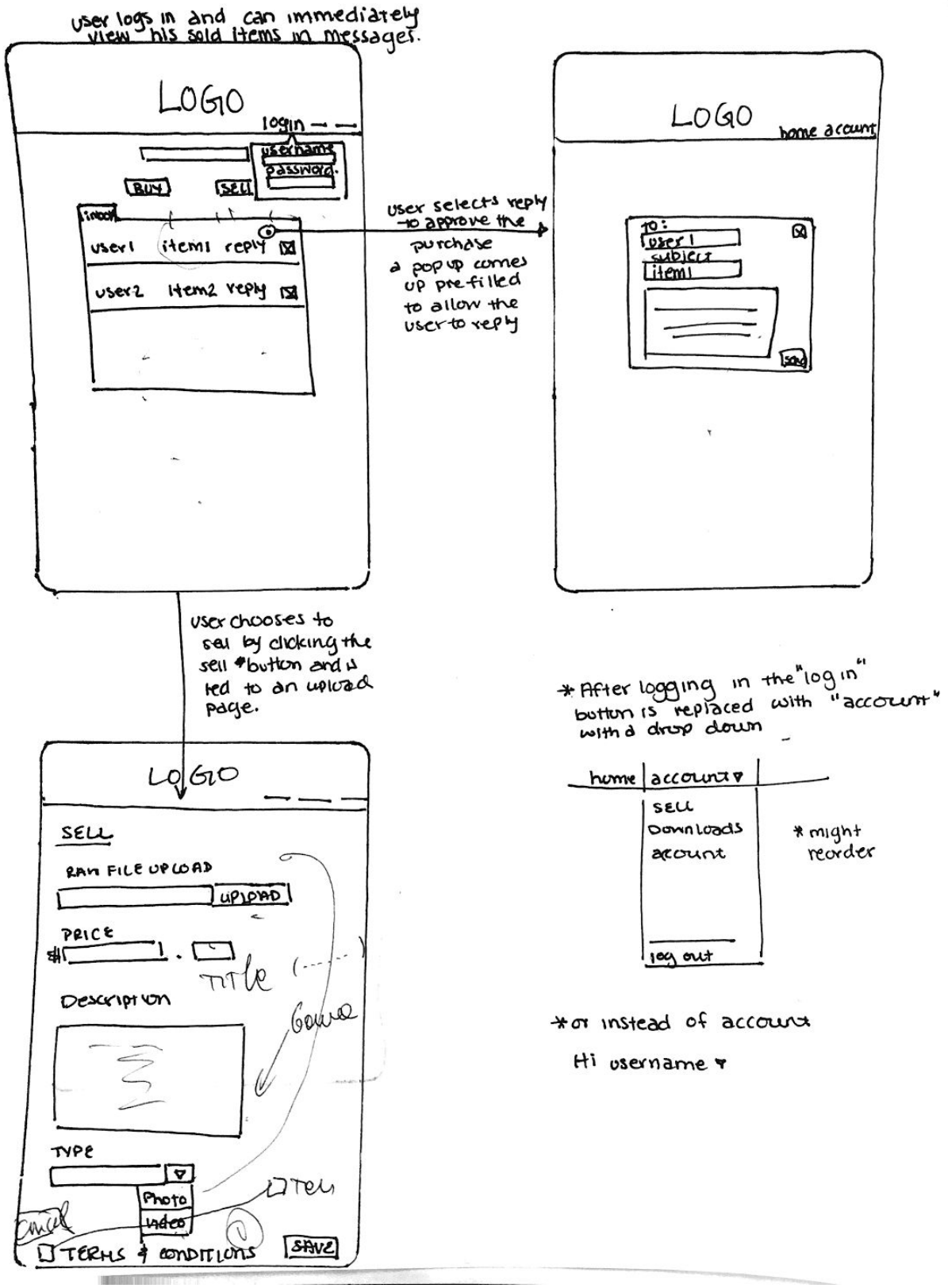
1. Application shall be developed using class provided LAMP stack
2. Application shall be developed using pre-approved set of SW development and collaborative tools provided in the class. Any other tools or frameworks must be explicitly approved by Anthony Souza on a case by case basis.
3. Application 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, Safari, Chrome.
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).
18. Browser Version Support: ● Internet Explorer (IE11 and IE10) ● Google Chrome (59.0.3071 and 58.0.3029) ● Mozilla Firefox (54 and 53) ● Safari (10.1 and 10.0.3) ● Opera (37 and 36)

5. UI Mockups and Storyboards

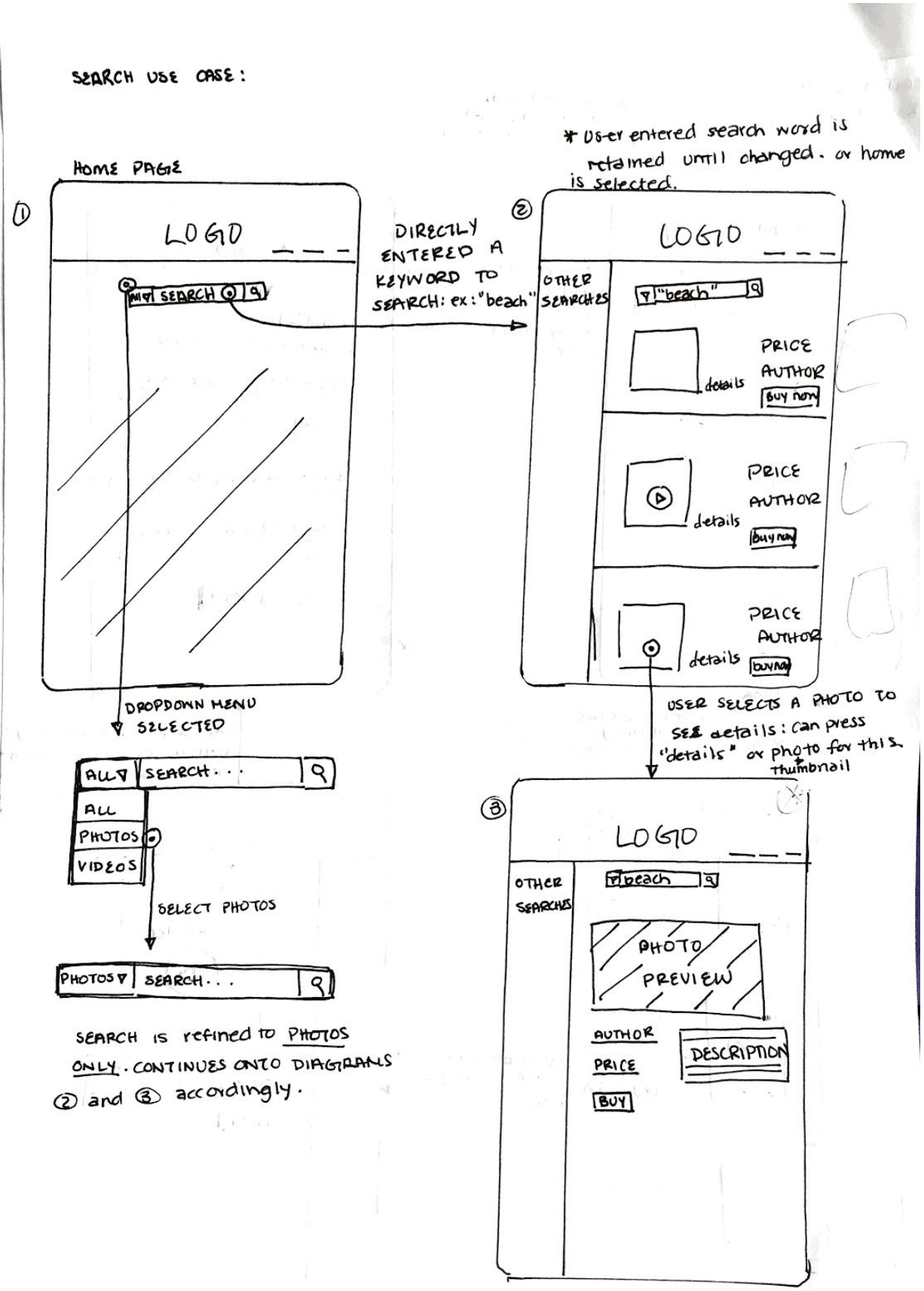



LOGIN USE CASE :
/REGISTER
User wishes to login: a sectioned tab will appear allowing them to do so.
LOGO
home login reg.
BROWSE
User wishes to register for the site and clicks on "register" which leads them to this page
LOGO
Registration
Be apart of Picturesque
First name
Last name
e-mail address
username (only letters & num.)
password (min. 6char)
JOIN
OR
User chooses to browse and search initially, then deciding to register
AFTER pressing the "BUY" button the user is prompt to sign up.
LOGO
BUY NOW
User is browsing and decides to buy a video
LOGO
PREVIEW
AUTHOR
PRICE
DESCRIPTION
BUY

SELLER/AUTHOR USE CASE:

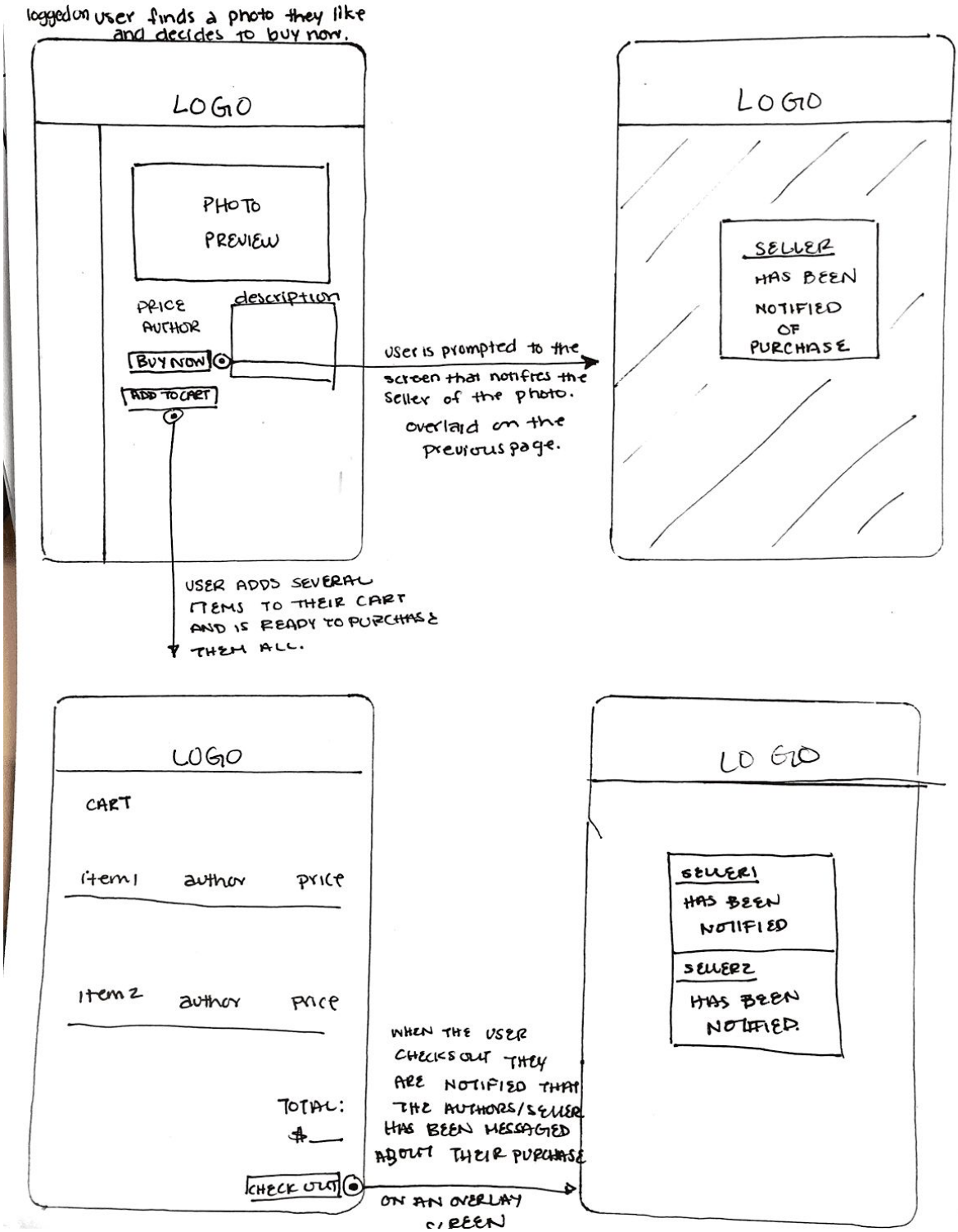


SEARCH USE CASE:



SEARCH is refined to PHOTOS ONLY. CONTINUES ONTO DIAGRAMS ② and ③ accordingly.

BUYING USE CASE:



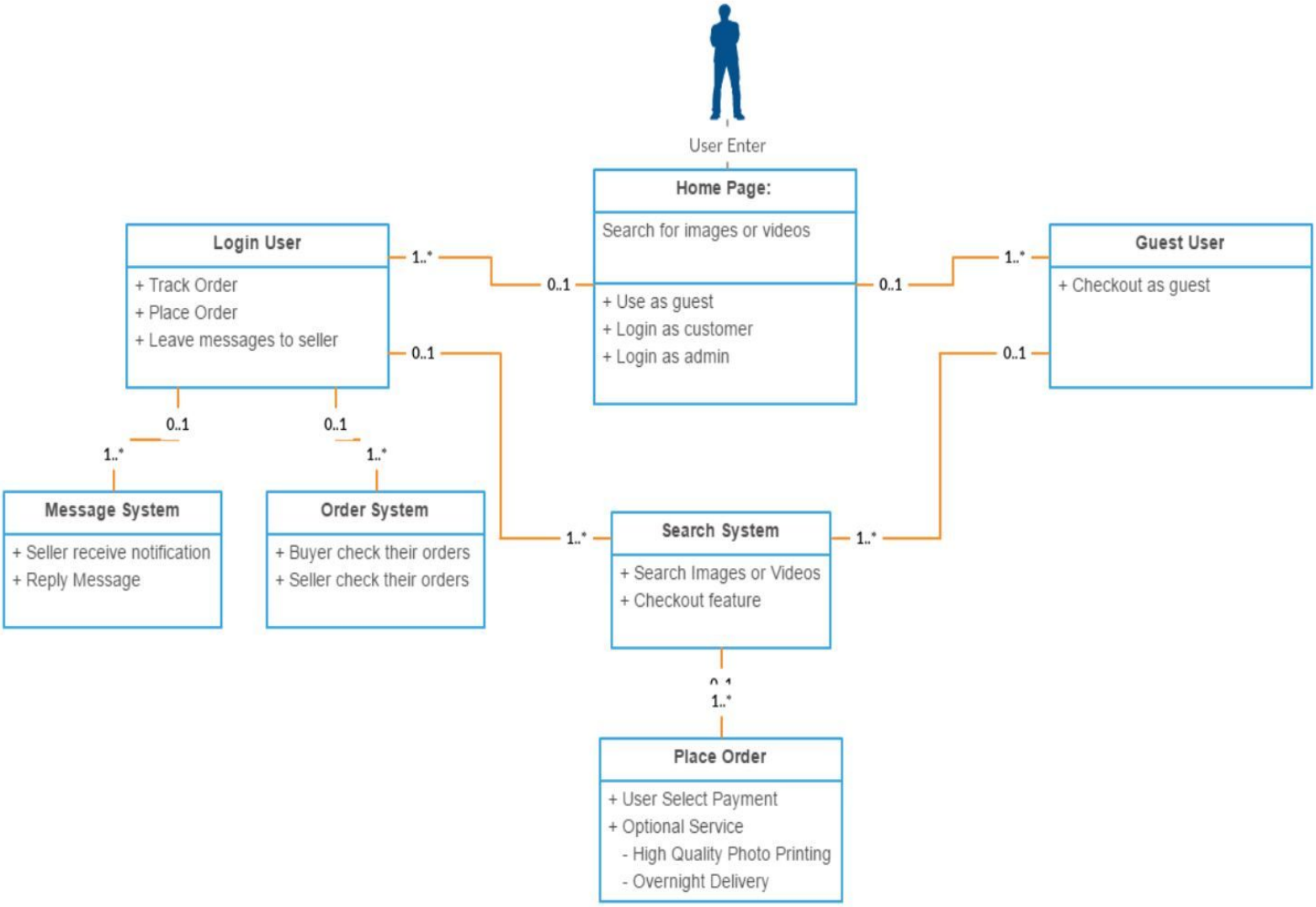
6. High level Architecture, Database Organization

- Software products used are LAMP, Workbench, Netbeans, Amazon cloud,

- 4 tables are used to store and organize our data.
    - media_genres
        - genre_id - is incremented each time a new entry is made.
        - genre_name - allows users to search for a specific category.
    - media
        - media_id - is auto incremented each time an entry is made.
        - media_title - is used to describe the currently searched media.
        - genre_id - is a foreign key that references column genre_id on the media_genres table.
        - author_id - is a foreign key that references column user_id on the users table.
        - type_id - is a foreign key that references column type_id on the media_types table.
        - price - allows users to see how much the media costs.
        - upload_date - displays the date in which media was uploaded.
        - permissions - allow what kind of user can view the media.
        - URL - points to where the media is located.
        - sold count - how many times the media has been sold.
    - users
        - user_id is auto incremented each time a user registers.
        - password is hashed so that it is encrypted.
        - email is unique so that a user cannot register an email twice.
        - registered_date - records date which the user registered.
        - last_login_date - records date of user's last login.
        - Username - user’s handle name.
        - role - indicates what type of user (admin or normal)
        - TOKEN -
        - salt allows users to register with existing information from another platform
    - media_types
        - type_id - is incremented each time a new entry is made.
        - type_name - media type (video or images).

- Images are in jpeg format and range in size from 1MB to 8MB. Videos are mp4 and are ~ 6.5 MB. Thumbnails will be 75x75. Every image will have administrative information along with the ability to contact the owner. Furthermore, technical information on what kind of medium was used to capture the media as well as descriptive data such as names or specific details pertaining to who/what was being photographed or filmed. All of our media are in color and will be stored in a File system.
- Users have the ability to use search terms relevant to beach, cars, cities, flowers, nature or sports and also have the ability to toggle whether or not they want videos or images. Our search algorithm finds all items in our image table that has a %Like relevant to the search. If search fails it displays all items.
- We will offer the option for users to use pre existing information that they have previously registered with another site e.g. facebook for a quick and effortless registration process.

7. High Level UML Diagram



8. Key Risks

Skills Risk:

Some skills may be lacking due to lack of experience for this type of project.

How to resolve:

Work together to figure out the necessary languages and common practices necessary to make this project successful. Pair programming can be used between someone less skilled and someone more skilled to help create an understanding with everyone on the development process and code.

Legal/Content Risks:

There is only a legal risk if the provided content was not in fact free as advertised and actually free of licensing. We must also ensure the logo being used (if made from a service) follows the guidelines for giving that service credit.

How to resolve:

Ensure that the site being used to collect the content is credible and is not falsely providing the media without permission. Also ensuring that the proper credit is given and appropriately displayed on the project site.

9. Team Organization

Tiffany Ku: Team Lead, will be working on front end development and in charge of organizing the team and ensuring communication.

Di Mao (Andy): Tech Lead, help with all issues regarding software and understanding its usage.

Haotian Zhang (Ivan): Making data definition and working on front end development, designing the layout and all view part.

Teng Yu: Working on backend and currently working on search.

Andrew Cheng: Working on backend and helping the others develop it further.

Cody Jackson: data engineer - develop and maintain database

Calvin Ip: Created use cases and working on front end UI/UX design