

# **Software Engineering CSC648/848 Spring 2016**

## **ImageGrove - Milestone 5**

Team 7:

Samuel Pope Gluss - sgluss@mail.sfsu.edu

Meghana Dayananda

Marisa Noelani Abril

Edilson J Araujo

Harry Cao

05/18/2016

Revision History:

Revision #	Date	Description
1	5/18/2016	M5 Initial

## **Table of Contents:**

Milestone 1	Page 3
Milestone 2	Page 12
Milestone 3	Page 43
Milestone 4	Page 47
Final Screen Shots	Page 59
Team Member Contributions	Page 74
Main Challenge	Page 77

# **Software Engineering CSC648/848 Spring 2016**

## **ImageGrove Milestone 1**

Team 7:

Samuel Pope Gluss sgluss@mail.sfsu.edu

Meghana Dayananda

Marisa Noelani Abril

Edilson J Araujo

Harry Cao

03/04/2016

Revision History:

Revision #	Date	Description
1	03/04/16	Changes from class meeting

## **1. Executive Summary:**

ImageGrove is an innovative new system to allow the modern artists to reach their customers, and enable those customers to find exactly the digital art piece they are looking for. Our design is built from the ground up to provide a seamless, quick responding interface that gets the user what they want. Using modern technologies and techniques gives us the power to provide a product so intuitive, any consumer will enjoy using it.

We will give artists the capability to upload and manage licensing for anything they create, be it movies or pictures. By adding tags to these products, customers can easily find new pieces to enjoy, and new artists to love. Our topoftheline authorization system will enable artists to manage their own accounts, customers to browse and buy, and administrators address and issues, streamlining the experience for everybody.

ImageGrove will allow artists to upload art in image or movie form, and easily maintain their portfolio. Social media integration means artists can easily keep in touch with their sales base. With easy control of pricing and licensing, they'll be able to spend less time chasing sales and more time following their passion! The users will also appreciate a well designed system that allows them to quickly find just the piece they were looking for with simple keyword searches. These features are crucial to retention of clientele, whose preference above all is for sleek Performance.

Our crack team of software developers are highly talented, and committed to creating industry leading user experiences. Their mastery of the backend means your customers will get to spend more time looking at the finest in modern art, and giving their hardearned money to your clients!

## **2. Use Cases:**

Pam is a 23 year old graduate of RIT, and is trying to start her career after completing her undergraduate degree in animation. She is a webdesign savvy artist, who produces work widely utilised by tech companies around the world. She wants an online portfolio where she can host her graphic design artwork, and her animation projects. She does work both in animation and graphic design, so she will need the capability to easily upload these types of work to her portfolio, and make changes as necessary. She is an avid user of social media, so she will want to update her followers on her favorite platforms.

Charles is a 35 year old system administrator who has no hair. He is wheelchair bound, so accessibility is a big concern for him. Like all IT professionals, he is interested in being able to do his job as quickly and easily as possible. He may need to return access to an account when an artist loses their password, or ban any account that is being misused. Furthermore, he will need to be able to create accounts for users or artists, depending on the limitations of their technical Abilities.

Donnie is a 80 year old powerball winner who has no hair too. He has bought a new castle and planned to decorate the new castle with modern printable art. He has a elementary school diploma and he does not know what kind of masterpieces people like. Donnie first types in some keywords about things he wants and then sorts the result by views, sales, prices, etc. Using the related module of the application, Donnie finds many pieces similar to the one he has purchased, and saves these pictures to the wishlist for future purchasing. Based on his personal preferences, he can check out different licenses which the artists provide and also he can pay them anonymously if he chooses.

Eric is a 9 year old student at elementary school. He needs some extraordinary pictures in order to finish his school project. His classmate Donald Trump tells him that our website may help him to get a good grade. Eric goes through the recommended tutorial to learn how to get what he wants from our application. However, he does not know how to describe what he really wants, therefore, he uploads an image and uses the find similar image feature to find pictures which are similar to his image. After he pays for the license, he is able to download the picture through the order history module. He is so pleased with his find, he also shares the picture to his friends through other social websites such as facebook and twitter.

### **3. Glossary:**

Artists - People who own the media content and are allowed to set a price for it.

Verified User - Person who has created an account of any kind

Customer - Person who creates an account to purchase art licenses

Administrator - Person with special access to filter and monitor the content of all accounts.

Media - includes images and videos.

Content policy - only appropriate content will be allowed on the system.

Web License - meaning one can use the content only on the web.

Print License - meaning one can use the content only to print.

Unlimited License - meaning one can use the content in an unlimited number of projects (both on web and to print)

#### **4. Initial list of functional specs:**

1. Each Artist shall be allowed to have a portfolio on the system.
2. Artists shall be allowed to upload images, videos on to their respective portfolios.
3. Artists shall be allowed to remove images, videos from their respective portfolios.
4. Artists shall be allowed to set price for their media under three category of licenses namely unlimited, web, print
5. Each Customer shall be allowed to have an account on the system.
6. Customers shall be able to search through the images in the system.
7. Customers shall be able to select the media of their choice.
8. Customers shall be able to download the media after making a payment corresponding to the media they wish to download.
9. System shall have an interface for an administrator.
10. Administrator shall have access to all the accounts on the system.
11. System shall allow the administrator to block the account of a customer who violates the content policy.
12. System shall allow the administrator to block the account of an artist who violates the content policy.

## **5. Initial list of nonfunctional specs:**

1. The system shall visually respond to the user within 6 seconds.
2. The system shall be user intuitive and easy to use.
3. The system shall abide by the content policy.
4. The system shall support commonly used image formats such as (jpeg, png, jpg)
5. The system shall protect the privacy of the users.
6. The data collected from the customers shall be used for identification purposes alone.
7. The system shall be compatible with widely used browsers such as firefox,chrome and IE

## 6. Competitive analysis:

Features	ImageGrove	Snapwire	GettyImages	Fotolia
Easy to use UI/Mobile-Friendly	+	+	+	-
Keyword Tags/Search by Text	+	+	+	+
Photographer Info	+	+		
Pricing of Images	+			-
Social Integration	+	+	+	-
Video Upload	+	+	+	+

We plan to use simple user interface that is mobile friendly, through the use of Bootstrap. All three competitors have eyecatching, but simple landing pages with featured image categories as well as a search bar clearly visible as soon as the front page loads. We would like to apply a similar, engaging style to attract users to our website. To allow customers to find the image(s) they are looking for, there will be an image search by use of keyword image tags. Some websites offer artist information, while others only provide a keyword search based on the artist's name. We want promote a more personal relationship between the customers and the artists and therefore want to show a short profile for each artist with information such as background, portfolio, contact information, etc. ImageGrove will allow artists to specify the price of each image, while most other websites have preset pricings for artist contributions. Fotolia in particular has a credit system where an artist earns credits for each purchase of his or her photos, which complicates the system and could be inconvenient for artists. In terms of social integration, all the websites we visited had some form of social media sharing. We would also like to incorporate social media sharing as it seems standard. All of the three websites have the option of uploading video, which as before with the social media we would like to be able to implement to keep up with the competitors.

## **7. Highlevel System Architecture:**

This following section will describe the tools, frameworks, and APIs being used in assisting our development of this project and how they will tie into the different components that brings this application together. LAMP

Our application will be developed following the traditional LAMP model environment that consists of the following:

- Linux
- Apache
- MySQL
- PHP Frameworks and APIs

As for the frontend of our application, we'll be utilizing the following frameworks to assist in our development:

- Twitter Bootstrap
  - A web development framework that allows us to get started quickly by providing a series of useful and tasteful templates and themes.
- jQuery
  - A JavaScript library that eases crossbrowser development using its own API. Supported Browsers

Our application shall be able to run smoothly in the following browsers:

- Google Chrome 49.0.2623.75, 48.0.2564
- Mozilla Firefox 44.02, 44.01

## **8. Team Configuration:**

Team Member Position Role

Sam Gluss Team Lead FrontEnd

Marisa Abril Tech Lead Backend

Harry Cao Backend

Meghana Dayananda FrontEnd

Edilson (Eddie) Araujo Full Stack

# **Software Engineering CSC648/848 Spring 2016**

## **ImageGrove - Milestone 2**

Team 7:

Samuel Pope Gluss - sgluss@mail.sfsu.edu

Meghana Dayananda

Marisa Noelani Abril

Edilson J Araujo

Harry Cao

03/11/2016

Revision History:

Revision #	Date	Description
5	4/22/16	M3 tech demo
4	3/18/16	M2 Final
3	3/11/16	Updated with M2 proto
2	03/10/16	Adjusted as per instructor feedback
1	03/04/16	Changes from class meeting

## **1. Executive Summary:**

ImageGrove is an innovative new system which allows artists to reach their customers. This enables art buyers to find the exact digital masterpiece they are looking for. Our design is built from the ground up to provide a seamless and responsive interface that gets the user what they want. Using the latest techniques in dynamic web application programming, we can provide a product so intuitive that any consumer will enjoy using it.

Artists want the capability to upload and manage licensing for anything they create. ImageGrove also allows artists to manage their portfolios of movies and pictures. By adding tags to these products, customers can easily find new pieces to enjoy, and new artists to love. Our comprehensive authorization system will enable artists to manage their own accounts, customers to browse and buy, and administrators to address any issues. This streamlines the experience for everybody, making sure their account privileges offer just the features they need.

ImageGrove's social media integration means artists can easily keep in touch with their sales base. Artists are given easy control of pricing and licensing, so they'll be able to spend less time chasing sales and more time following their passion! The users will also appreciate a well designed system which quickly allows them to find just the piece they were looking for using simple but robust keyword searches. These features are crucial to retention of clientele, who expect sleek performance from the apps they use.

Our expert team of software developers are highly talented and committed to creating industry leading user experiences. Their mastery of application development means your customers will get to spend more time looking at the finest art products and giving your clients their hard-earned money!

## **2. Use Cases:**

### **Artist uploading image**

Pam is a 23 year old graduate of RIT, and is trying to start her career after completing her undergraduate degree in animation. She does work both in animation and graphic design, so she will need the capability to easily upload these types of work to her portfolio, and make changes as necessary. She already has an artist account, so she finds the login field in the top right of the page. She logs in to the website by entering her username and password in the dialog and pressing ‘enter’. She then clicks the ‘my portfolio button’ on the left side menu. She selects ‘upload’ and picks some work from her computer that she wants to add to her online portfolio. Once these files are uploaded, dialogs are available beneath in which she can add titles, descriptions, and any tags she feels like the media should have. She also assigns the media to specific albums if she likes. She is an avid user of social media, so she will want to update her followers on her favorite platforms, by pressing the appropriate social media button on her profile.

### **User creating account**

Ana is trying to find an image to use for a sticker on her new truck. While browsing the web for a suitable stock image, she finds ImageGrove in a top-5-of-2016 image licensing websites lineup. Upon visiting ImageGrove, she is able to easily locate the search box in the top of the screen and inputs “rainbow unicorn”. She is redirected to a results page, where all the appropriate matches are presented. She finds one she likes, and selects a license from those available, which is placed in her cart. She clicks on the prominent checkout button, and is prompted to create a user account. She is presented a form where she must fill out her name, select a username, and enter her email. Upon completion, a confirmation email is sent to her email account, and she is able to complete checkout of her selection. Once checkout is complete, she receives a receipt email as well.

### **Admin making change to artist album**

Charles is a 35 year old system administrator. Like all IT professionals, he is interested in being able to do his job as quickly and easily as possible. He may need to return access to an account when an artist loses their password, or delete any picture or video that is not in conformance with the ImageGrove terms of service. After he logs in to the website, the management tool tab will show in the top right corner of the page. He sees a message indicator on the right side of the tool tab. When he clicks on it, he is informed in a new window that an artist has miscategorized a group of their images. Charles navigates to their page by clicking on the artist’s name, and selects the offending images by clicking on a selector in the corner of each image. He is then able to modify the metadata of the images as needed, by modifying values in a dialog box. When he clicks ‘save’, these modifications are saved. He then clicks on the ‘send

message' link next to the artist's name, and sends them a message notifying them that the necessary changes have been made to their album.

### **Admin blocking artist for abuse**

Kathy is a 25 year old IT specialist, who is working support for ImageGrove. She has received numerous email complaints about an artist, Daniel, who is posting material which breaches the ImageGrove terms of service. On the list of her administrative options, she clicks on the "List of Artists" link. She is taken to the list of artists page, which displays all registered artists on ImageGrove and she finds the offending artist on the list. Kathy selects the "ban with message" button located next to Daniel's name. In the popup, she is able to determine the ban period, the restrictions on the account, as well as attach a reprimanding message to the artist, with a link to the application's terms of service.

### **Customer searching and purchasing**

Donnie is a 80 year old powerball winner. He has bought a new estate, and plans to decorate the new house with modern printable art. He is not very tech savvy, so an interface with easy flow and obvious functionality will be important for him. He goes to the ImageGrove website and views the landing page. He clicks on the search bar at the top of the screen and enters his keyword search for the art that he wants. After he clicks the search button, he is directed to the search results page, displaying the images that match his keyword search. Donnie clicks on the "Add to Cart" button located under the thumbnail of the image that he wants to purchase. When Donnie finds and adds all the images that he wishes to purchase into his cart, he clicks the "Checkout" button, which takes him to the payment information page where he completes his purchase.

### **Customer trying to figure out what he wants**

Eric is a 9 year old student at elementary school. He needs some extraordinary pictures in order to finish his school project. His classmate tells him that ImageGrove may help him to find some unique media. Eric goes through the recommended tutorial to learn how to get what he wants from ImageGrove. However, he does not know how to describe what type of image he is really looking for. As a result, he takes advantage of the ImageGrove advanced search, which allows him to match synonyms of search terms in image data, as well as tags. He selects advanced search from below the search window, and selects the type of media metadata he is interested in searching for. His results are presented in grid form. After he pays for the license, he is able to download the picture through the order history module. He is so pleased with his find, he also shares the picture with his followers on twitter, by clicking the twitter icon next to his purchase.

### **3. Glossary:**

Term	Description
Verified User	A person who has created an account of any kind.
Artist	A person who owns the media content and is allowed to set a price for the content.
Customer	A person who creates an account to purchase art licenses.
Administrator	A person with special access to filter and monitor the content of all accounts.
Media	Files and data uploaded to the website by an artist, including images and videos.
Content policy	Only appropriate and legal content will be allowed on the system.
Web License	A license with which a customer can use the content only on the web.
Print License	A license with which a customer can use the content only to print.
Unlimited License	A license with which a customer can use the content in an unlimited number of projects (both on web and in print)

## **4. Functional specs:**

### **Priority 1 :**

1. Each Artist shall be able to have a portfolio on the system.
  - a. Each Artist shall be able to create an unique account associated with their email-id.
  - b. Each Artist shall be identified with his/her account for further transactions.
2. Artists shall be able to upload images on to their respective portfolios.
  - a. Each Artist shall be able to upload images of their choice on their portfolio using the UI.
  - b. Image upload shall be blocked if it exceeds the maximum size(refer: non-functional spec 4 for supported size) .
  - c. Image upload shall be blocked if it is an unsupported format.(refer: non-functional spec 3 for supported formats)
4. Artists shall be able to remove images, videos from their respective portfolios.
  - a. Each artist shall be able to delete the images that they do not wish to keep. This shall be accomplished using the UI .
5. Artists shall be able to remove videos from their respective portfolios
  - a. Each artist shall be able to delete the images that they do not wish to keep. This shall be accomplished using the UI .
6. Artists shall be able to set a price for their media under three category of licenses namely unlimited, web, print.
  - a. Each Artist shall be able to set a price for their media under unlimited category, which means that the media can be used in unlimited ways by a customer upon making the purchase.
  - b. Each Artist shall be able to set a price for their media under web category ,which means that the media can be used on the web by a customer upon making the purchase.
  - c. Each Artist shall be able to set a price for their media under print category, which means that the media can be used to print by a customer upon making the purchase.
7. Each Customer shall be able to have an account on the system.
  - a. Each Customer shall be able to create an unique account associated with their email-id.
8. Customers shall be able to search through the images in the system.
  - a. Customers shall be able to search through the images using the UI to select images.
9. Customers shall be able to select the media of their choice to purchase.

10. Customers shall be able to download the media after making a payment corresponding to the media they wish to download.
  - a. Customers shall be able to download the media only after they have made a payment.
  - b. Download link should be provided after the payment has been made.
  - c. A visual response shall be displayed during the download process
12. Administrator shall have access to view all the accounts on the system.
  - a. Administrator is like a super user and he/she should be given access to history of customer's purchase.
  - b. Administrator should be given access to artist's profile.
14. System shall allow the administrator to block the account of a customer who violates the content policy(refer glossary).
15. System shall allow the administrator to block the account of an artist who violates the content policy(refer glossary).
18. Customers shall be able to search through media using keywords.
  - a. Customers shall be able to search through the media using keywords in search boxes.
  - b. Customers shall be able to search through the media using the dropdown provided.
35. Artists shall be able to add description of the image using text box on the UI.
36. Artists shall be able to add title of the image using text box on the UI.
38. Artists shall be able to add description of the video using text box on the UI.
39. Artists shall be able to add title of the video using text box on the UI.
40. Artists shall be able to add tags to video.

### **Priority 2:**

3. Artists shall be able to upload videos on to their respective portfolios.
  - a. Each Artist shall be able to upload videos of their choice on their portfolio using the UI.
  - b. Video upload shall be blocked if it exceeds the maximum size.( refer: non-functional spec 6 for supported size).
  - c. Video upload shall be blocked if it is an unsupported format(refer: non-functional spec 5 for supported formats).
11. System shall have an interface for an administrator to monitor activities of all the accounts.
  - a. Using this interface the administrator shall be able to perform all his/her administrator related activities.

13. Administrator shall be able to assist the users with activities related to their account.
  - a. This shall be done using the administrator interface.
  - b. Administrator shall be able to create accounts for the Customers.
  
19. Customers shall be able to search for media by size
  - a. Upon selecting the size of the media the result shall be filtered and only those which are within the limit of the entered size shall be displayed to the Customer.
  
22. Customers shall be able to sort the media search result based on price.
  - a. Upon selecting the price of the media the result shall be filtered and only those which are within the limit of the entered price shall be displayed to the Customer.
  
27. Customers shall be able to search for media by titles
  - a. Upon entering the title of the media the result shall be filtered and only those which match the entered title or those which contain entered title shall be displayed to the Customer.
  
28. Customers shall be able to search for media by descriptions
  - a. Upon entering the description of the media the result shall be filtered and only those which match the entered description or those which contain entered description shall be displayed to the Customer.
  
31. Administrator shall be able to send an email notice to the Customer after blocking the account.
  
32. Administrator shall be able to send an email notice to the Artist after blocking the account.
  
37. Artists shall be able to add tags to image.

**Priority 3:**

16. Customers shall be able to integrate with social networking websites(facebook,twitter) .
  
17. Artists shall be able to integrate with social networking websites(facebook,twitter) .
  
20. Customers shall be able to add media to his/her wish-list.

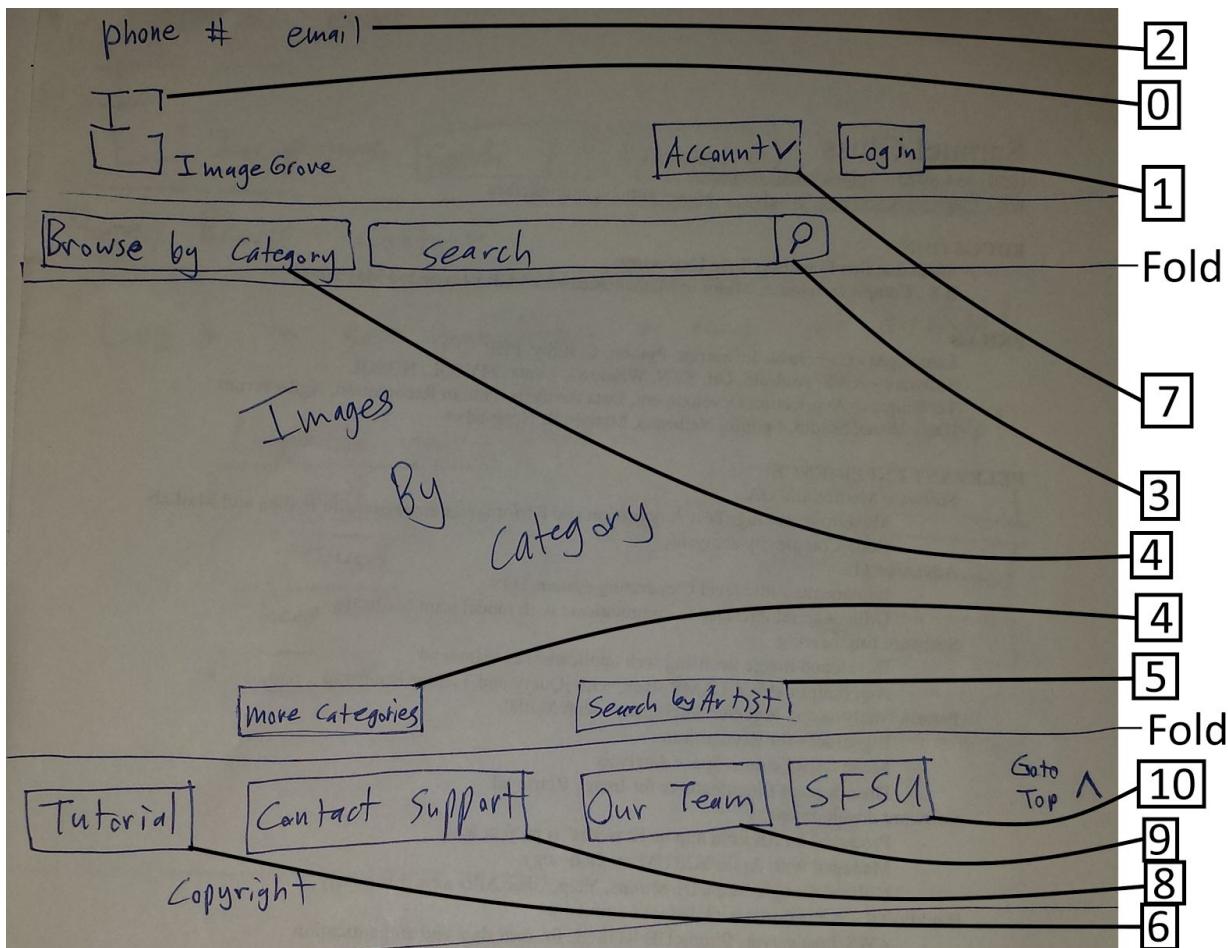
- a. Customers shall be able to add media to his/her wish –list to make a purchase anytime in future.
21. Customers shall be able to make an anonymous payment for the media.
- a. Those customers who wish to checkout as a guest shall be allowed to do so by making an anonymous payment.
23. Customers shall be able to sort the media search result based on sales.
- a. Upon selecting the sales parameter of the media the result shall be filtered and only those which are within the limit of the entered sales parameter shall be displayed to the Customer.
24. Customers shall be able to sort the media search result based on views.
- a. Upon selecting the views parameter of the media the result shall be filtered and only those which are within the limit of the entered views parameter shall be displayed to the Customer.
25. Customers shall be able to view their browse history.
- a. Customers shall be able to view their browse and purchase history only if the activity is associated with their account. In other words anonymous customer will not be able view history.
26. Customers shall be able to purchase media based on browse history.
- a. Only those Customers who are associated with an account and those whose history has been logged on to the system shall be allowed to make a purchase based on browse history.
29. System Shall be able to send an email to the Customer after an account has been created on the system.
30. System shall be able to send an email to the corresponding Customer after a purchase has been made by him/her.
33. Users shall be able to message each other.
34. Artists shall be able to subdivide their portfolios into albums.

## **5. Non-functional specs:**

1. The system shall visually respond to the user within 6 seconds.
2. The system shall be user intuitive and easy to use.
3. The system shall support commonly used image formats such as (jpeg, png, jpg)
4. The system shall support image upload of size upto 1MB only.
  - a. The system shall display an error message to the user who is trying to upload an image which exceeds 1MB.
5. The system shall support mp4 video format.
6. The system shall support video upload of size upto 5 MB only.
  - a. The system shall display an error message to the user who is trying to upload a video which exceeds 5 MB.
7. The system shall protect the privacy of the users. The data collected from the customers shall be used for identification purposes alone.
8. The system shall be compatible with widely used browsers such as firefox 44, chrome 49.02 and IE -10
9. The system shall be compatible with social network platforms such as facebook, twitter.
10. The system shall be hosted by AWS
11. Dynamic web content shall be generated by PHP5
12. Website layout shall be generated by Bootstrap
13. Dynamic website behaviour shall be produced with jQuery

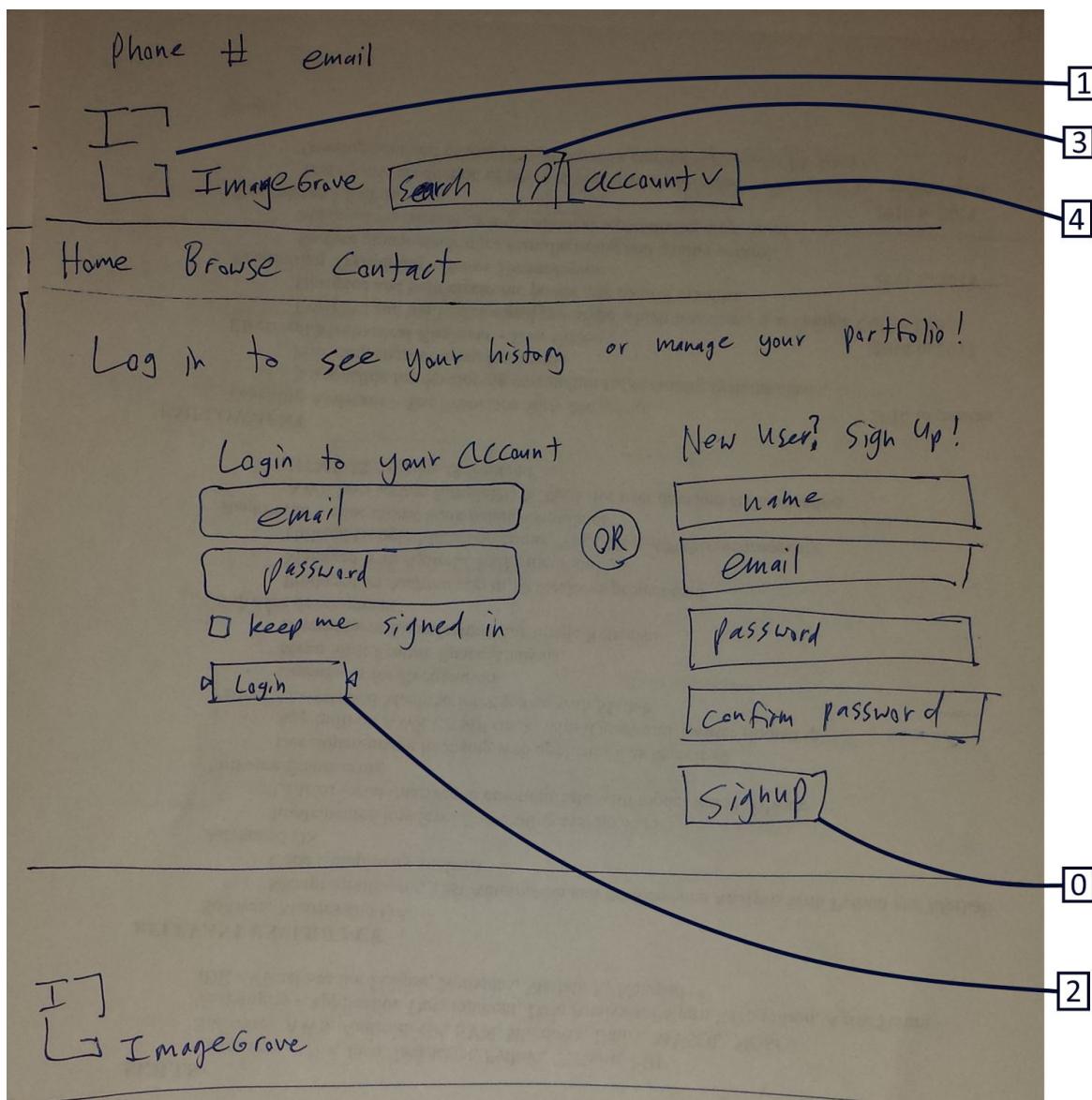
## 6. UI mockups and Storyboards

### 1. Main Screen



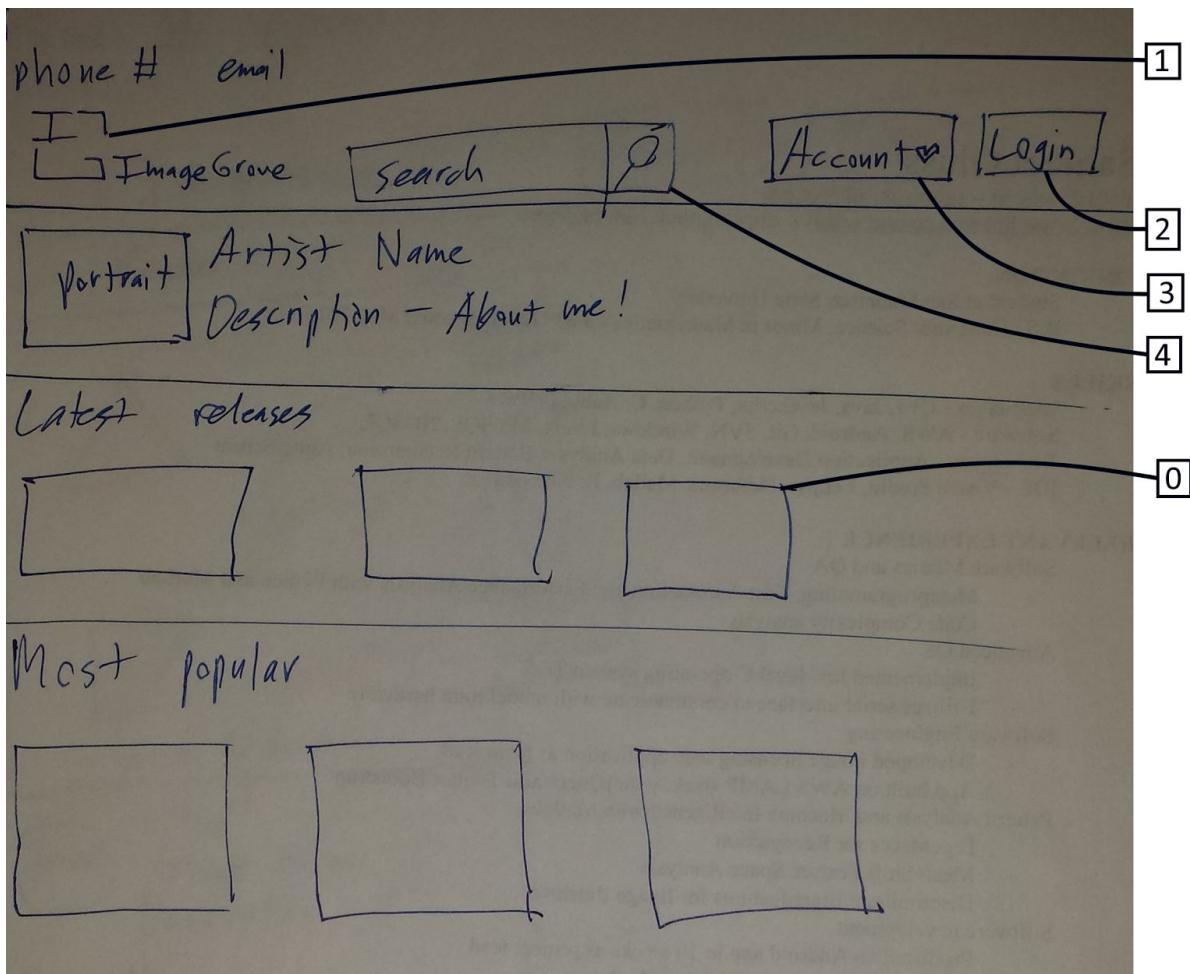
0. Clicking 'logo' returns user to main page
1. Clicking 'login' takes user to login page
2. Clicking on email allows user to send an email
3. Entering a search query takes user to search results page
4. Clicking on 'Browse by Categories' directs user to Image Browse page
5. Clicking on Search Artists take user to search page
6. Clicking 'tutorial' takes user to tutorial page
7. Clicking 'Account' tab allows access to any account-specific functionality
8. Clicking on 'Contact Support' allows users to contact site admin
9. Clicking 'our team' takes user to about page
10. Clicking 'SFSU' takes user to San Francisco State University homepage

## 2. Signup/Login page



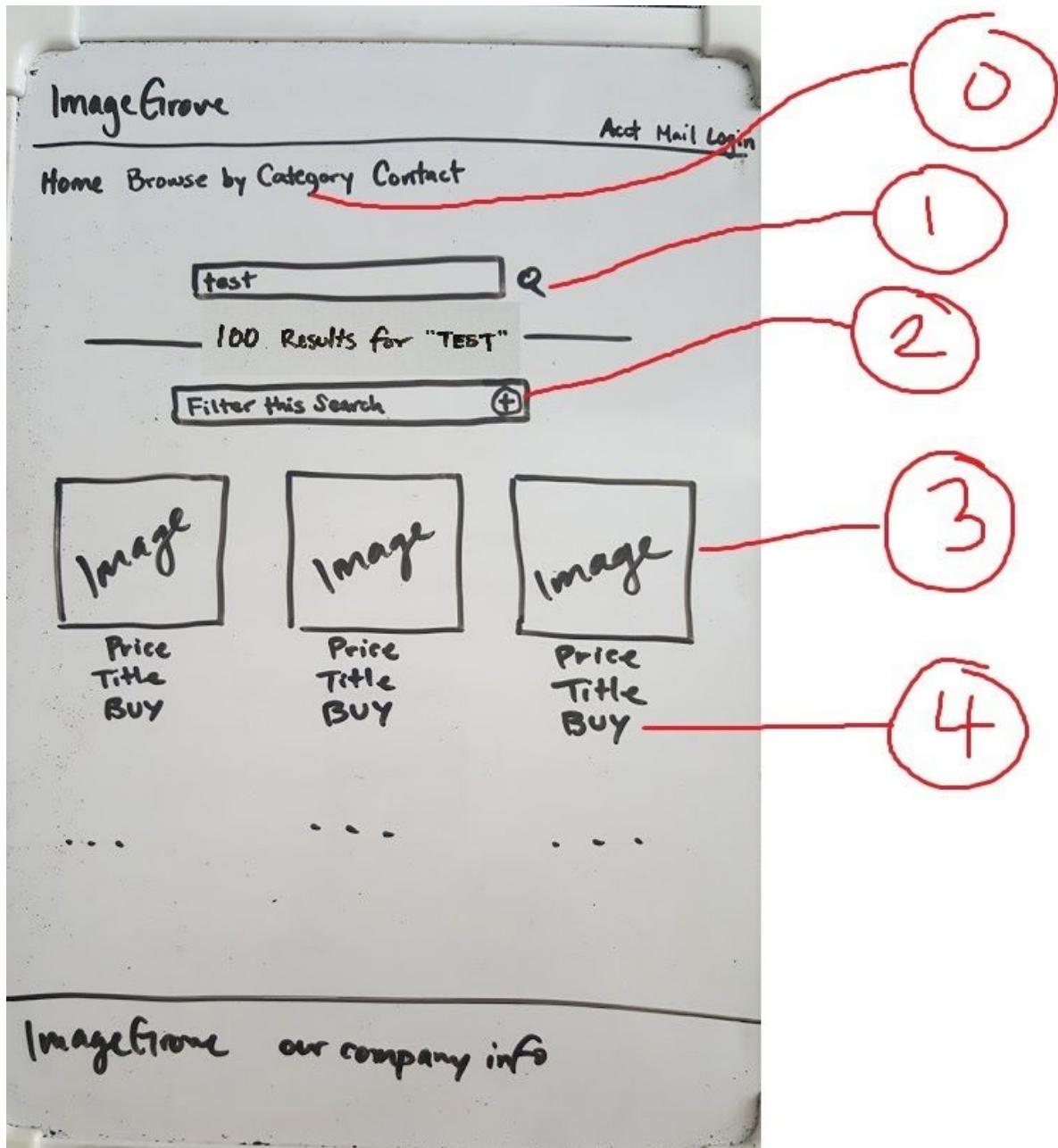
0. Clicking 'sign up' adds user registration information, if fields are correctly filled out.
1. Clicking 'logo' returns user to main page
2. Pressing 'enter' key logs user in, if username/password are correct
3. Search bar allows user to enter a query, pressing enter directs user to results page
4. Clicking 'Account' tab allows access to any account-specific functionality

### 3. Browse Artist Page



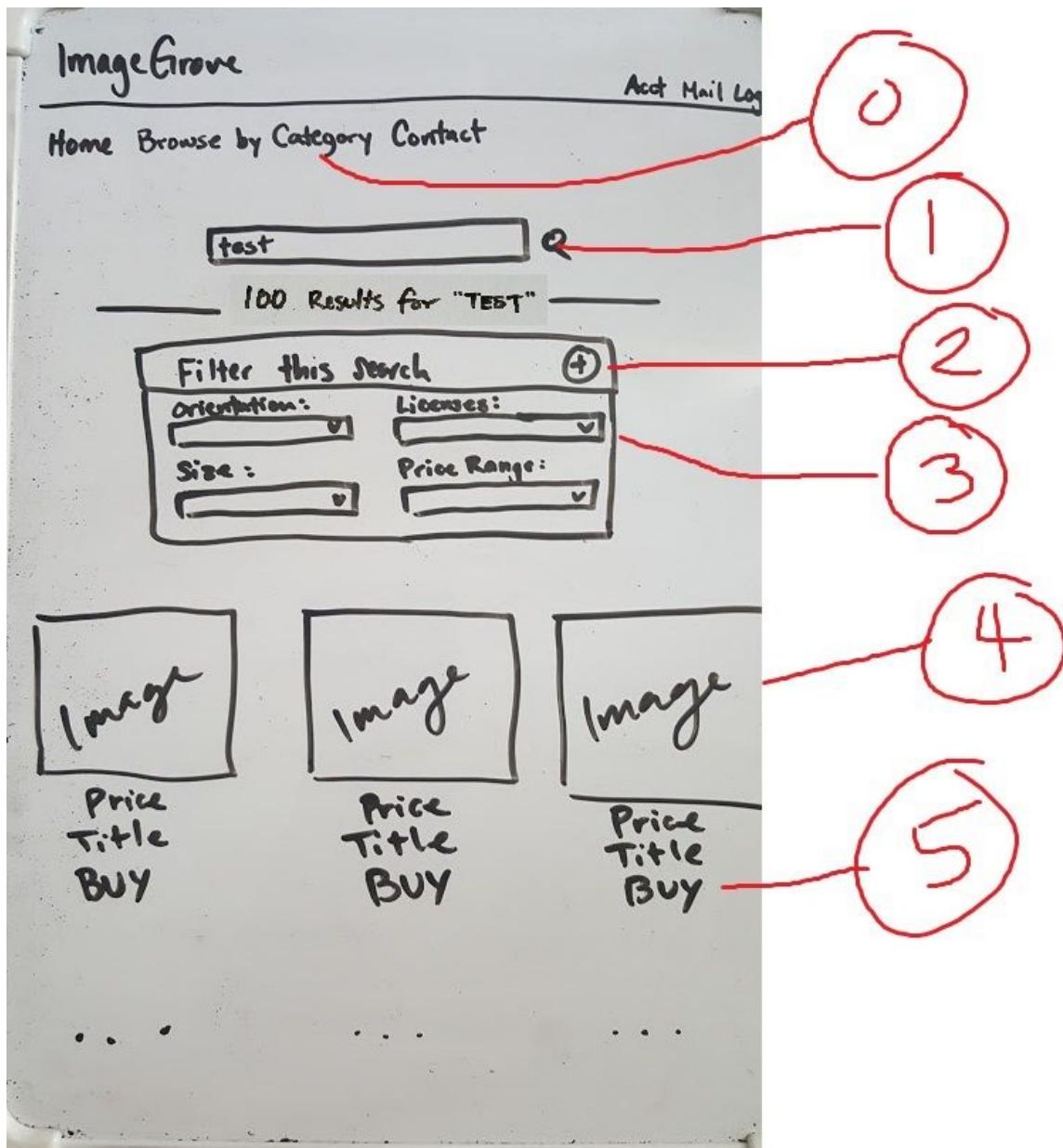
0. Clicking on any image takes user to view image page
1. Clicking 'logo' returns user to main page
2. Clicking 'login' directs user to login page
3. Clicking 'Account' tab allows access to any account-specific functionality
4. Search bar allows user to enter a query, pressing enter directs user to results page

#### 4. Search Results Page



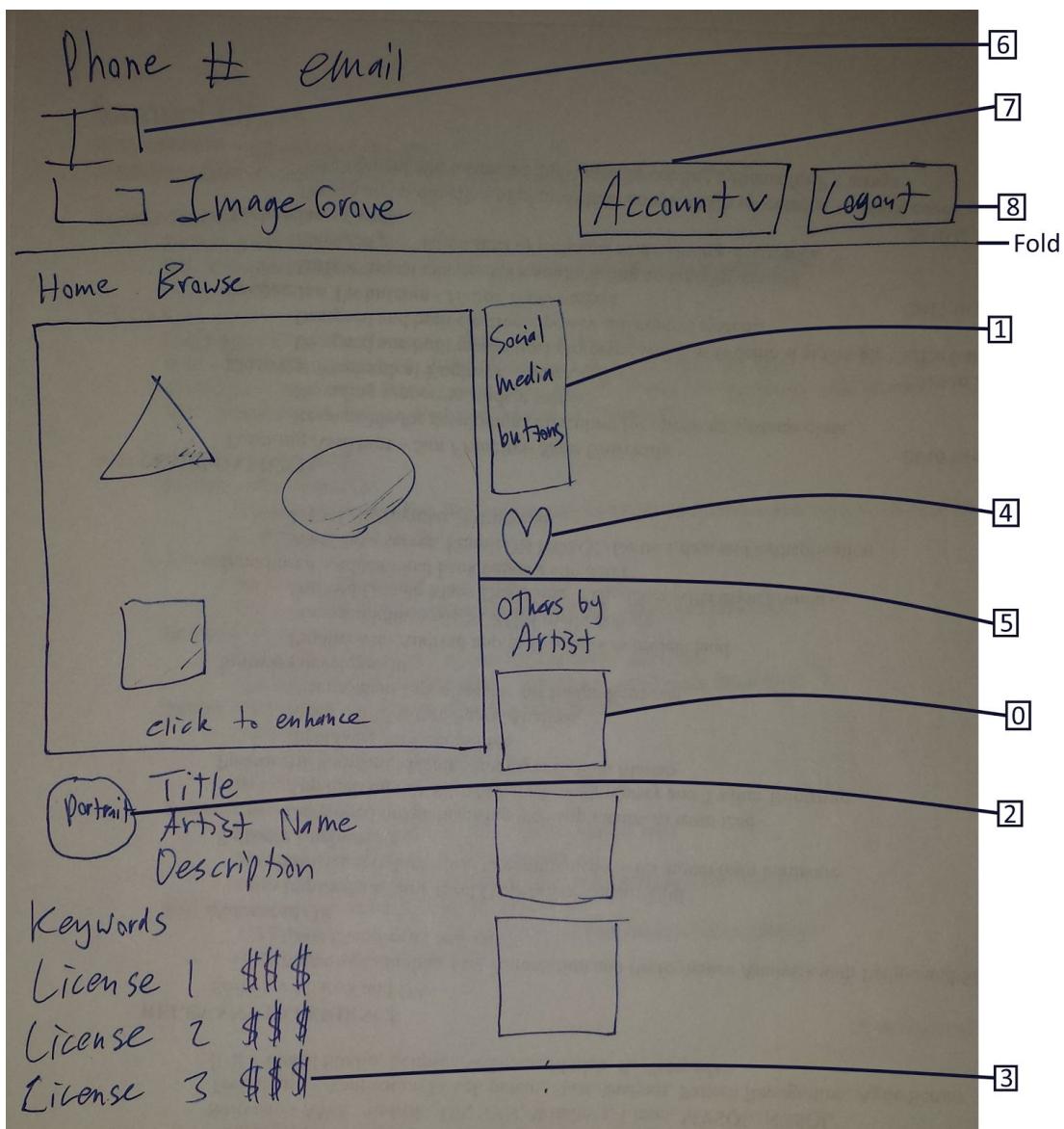
0. Clicking on “Browse by Category” link takes the user to the category page.
1. User can enter a new search and hit return/enter to initiate search.
2. User can click on the filter box to show list of available search filters
3. Clicking on an image in the search results brings the user to the image detail page
4. Clicking on the buy button adds the image the user’s account, if the user is logged in, if not, prompts the user to log in before buying process can continue.

## 5. Search Results (Filter Box Dropdown)



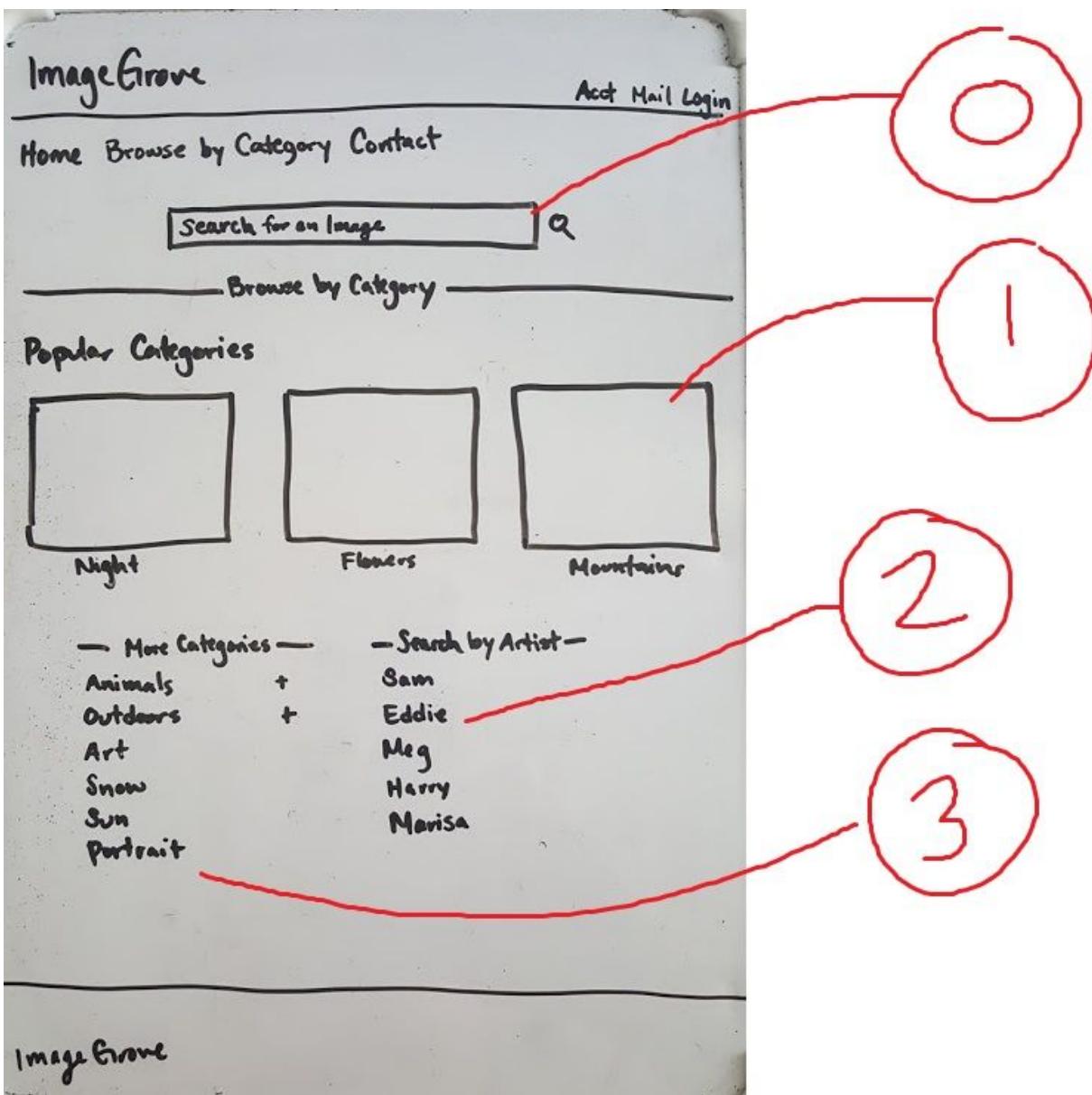
0. Clicking on “Browse by Category” link takes the user to the category page.
1. User can enter a new search and hit return/enter to initiate search.
2. User can click on the filters and choose an option from a drop down menu to filter their current search.
3. Clicking on an image in the search results brings the user to the image detail page
4. Clicking on the buy button adds the image the user’s account, if the user is logged in, if not, prompts the user to log in before buying process can continue.

## 6. Image Detail Page



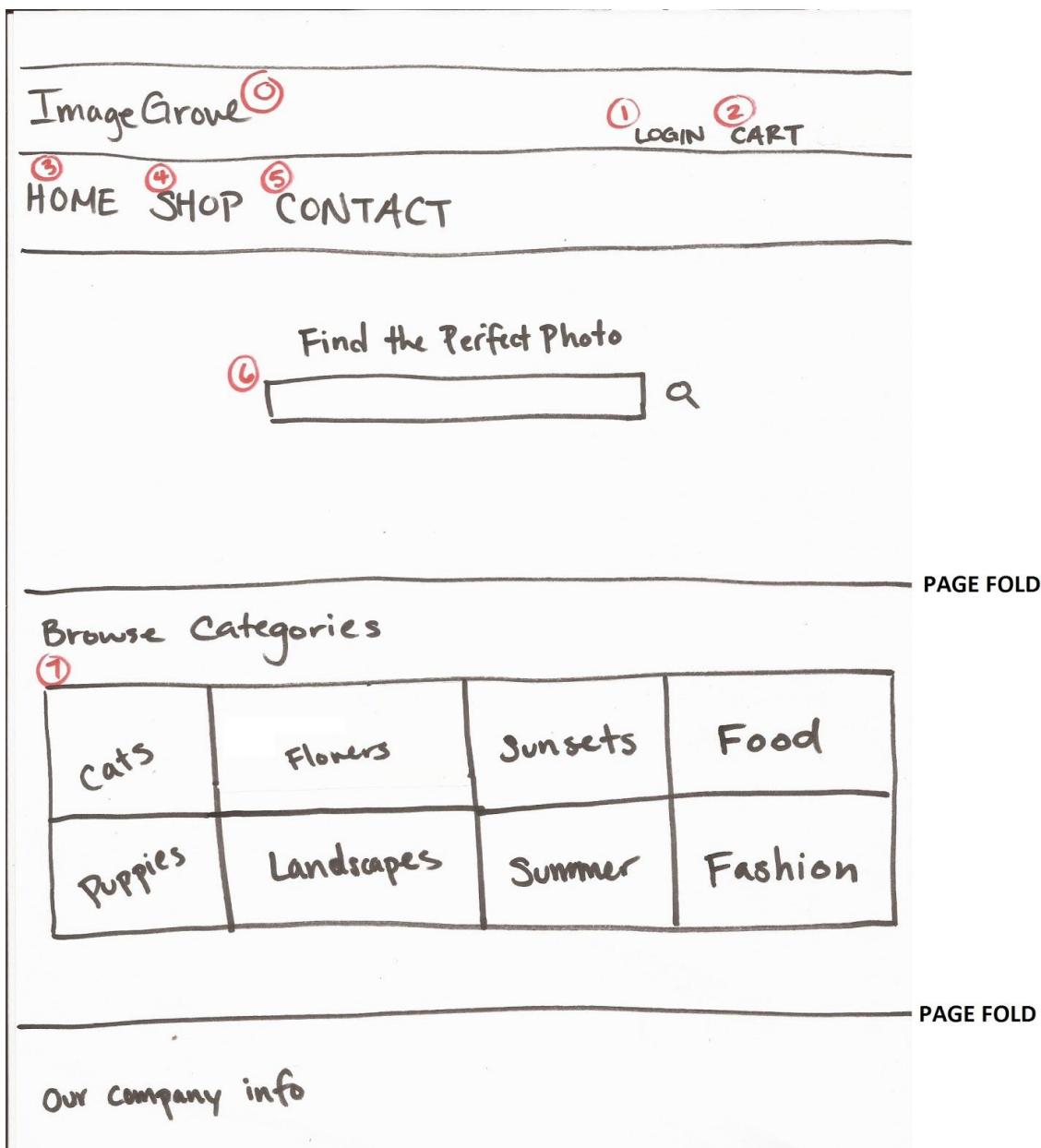
0. Clicking on any image takes user to view image page
1. Clicking on 'social media' button shares image to social media
2. Clicking portrait takes user to artist page
- 2b. Clicking on any keyword takes user to search results page for that keyword
3. Clicking on buy license places that license in the cart. License is then grayed out.
4. Clicking on 'save for later' button save image to favorites list
5. Clicking image zooms to full size. Click anywhere to return.
6. Clicking 'logo' returns user to main page
7. Clicking 'Account' tab allows access to any account-specific functionality
8. Clicking 'Logout' exits from user account

## 7. Browse by Category Page



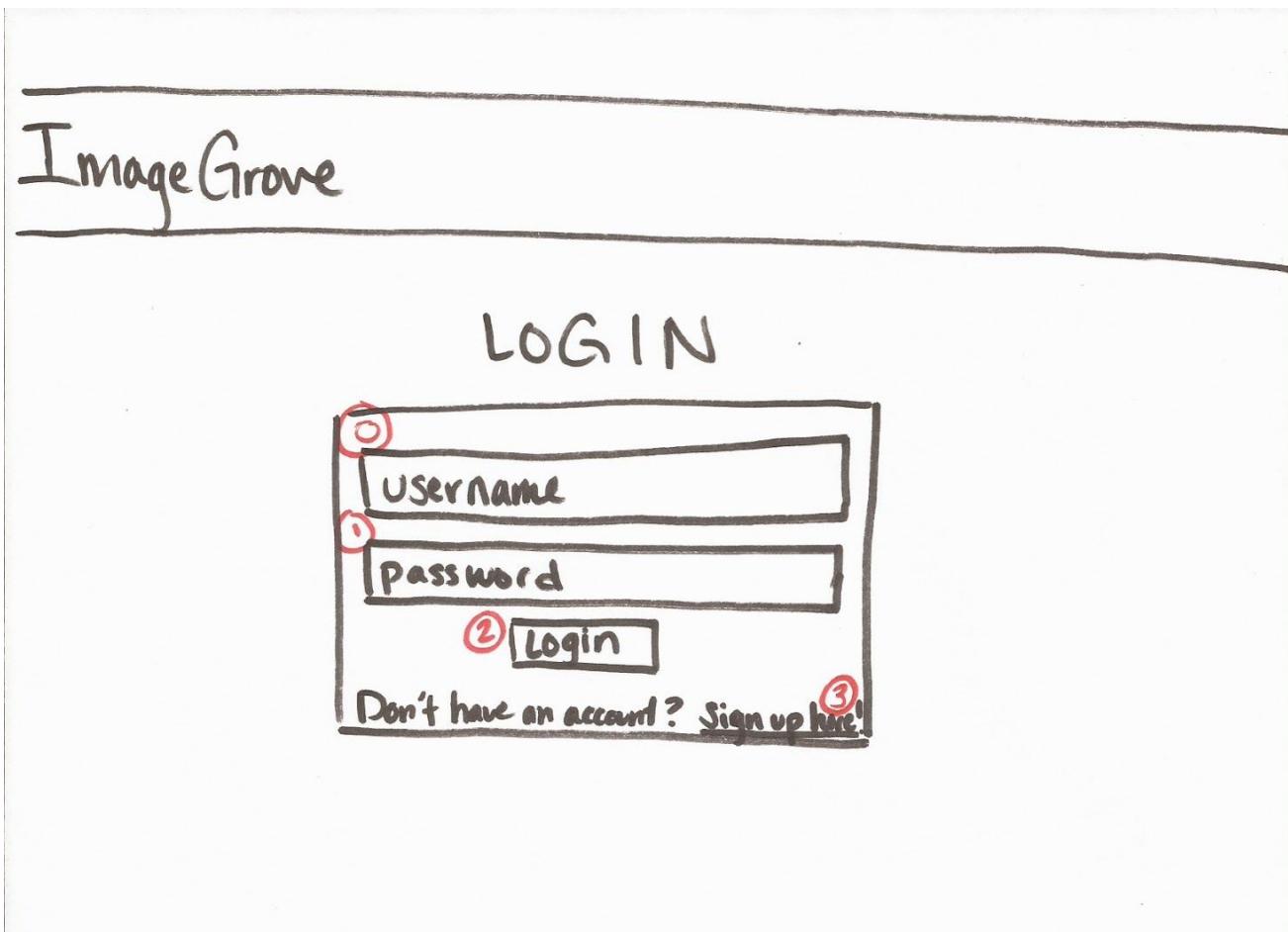
0. Entering in a search keyword and pressing the return/enter key will initiate a new search based on the given keyword.
1. Clicking on a category image will initiate a search with the category name as the search keyword.
2. Clicking on a category link will initiate a search with the category name as the search keyword.
3. Clicking on an artist name will take the user to the artist profile

## 8. Landing page



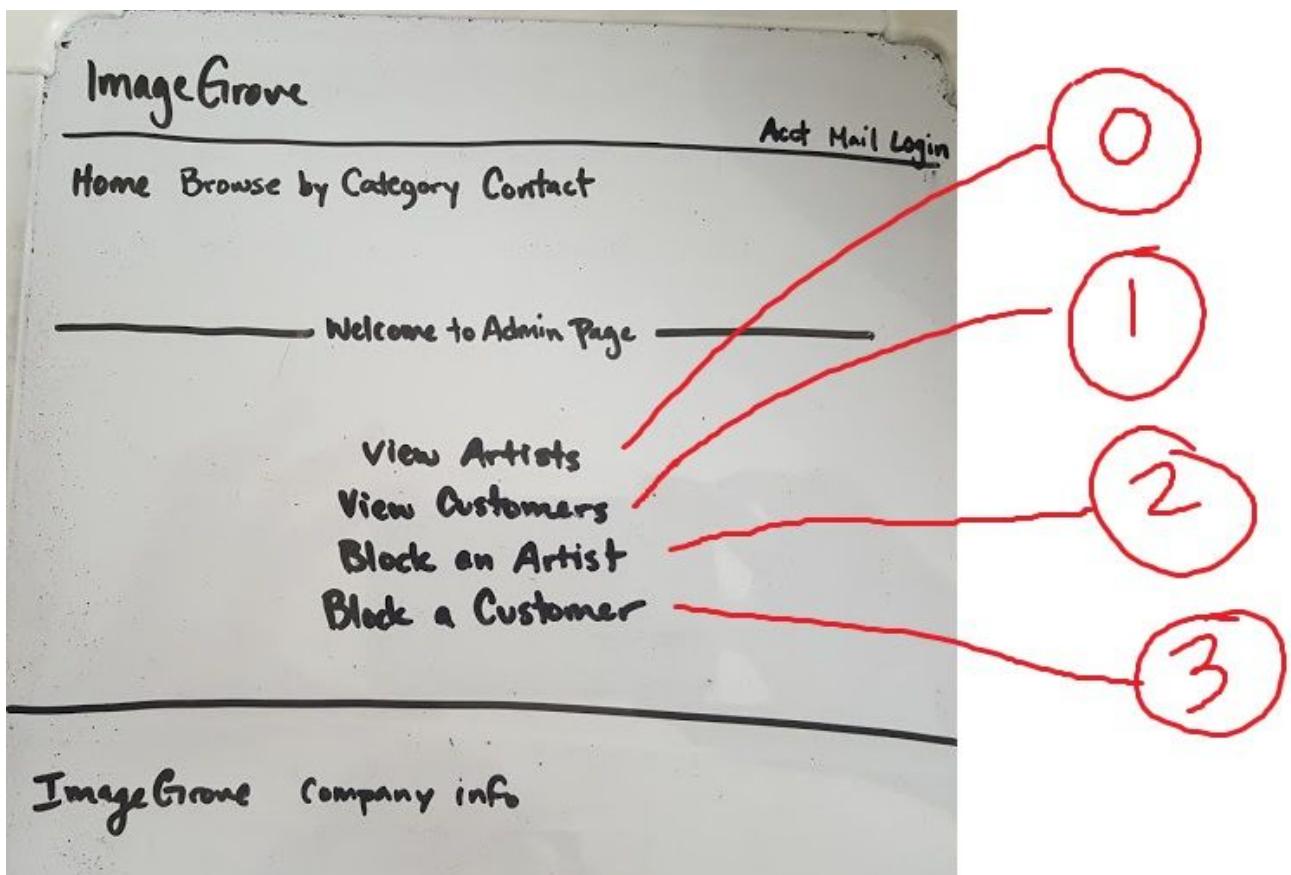
0. Clicking logo returns user to landing page
1. Clicking on Login button takes user to Login page
2. Clicking on Cart button takes user to View Cart page
3. Clicking on Home button returns user to landing page
4. Clicking on Shop button takes user to a page with a search bar
5. Clicking on Contact button takes user to a page with company contact info
6. Clicking on Search bar prompts user to enter in a text search and hitting return key will initiate search
7. Clicking on a category box takes the user to a search results page based on that category

## 9. Login page



0. Clicking on username field allows user to enter in username
1. Clicking on password field allows user to enter in password
2. Clicking on login button submits username/password combination and takes user to account page if login is successful
3. Clicking sign up button takes user to sign up form

## 10. Admin Account Page



0. Clicking on view all artists button takes admin to a list of all registered artists
1. Clicking on view all artists button takes admin to a list of all registered users
2. Clicking on block an artist button takes admin to block form page
3. Clicking on block an customer button takes admin to block form page

## 11. List of Artists page

The sketch depicts a user interface for managing artists. At the top left is the logo "ImageGrove". To its right are links for "ACCOUNT LOGOUT" and "[Return to Account](#)". Below this, the title "List of all Artists" is centered. The main content is a table listing six artists:

Artist Name	Action	Action
Daniel	<a href="#">View profile</a> ①	<a href="#">Block</a> ②
Eddie	<a href="#">View profile</a>	<a href="#">Block</a>
Sam	<a href="#">View profile</a>	<a href="#">Block</a>
Meg	<a href="#">View profile</a>	<a href="#">Block</a>
Harry	<a href="#">View profile</a>	<a href="#">Block</a>
Marisa	<a href="#">View profile</a>	<a href="#">Block</a>

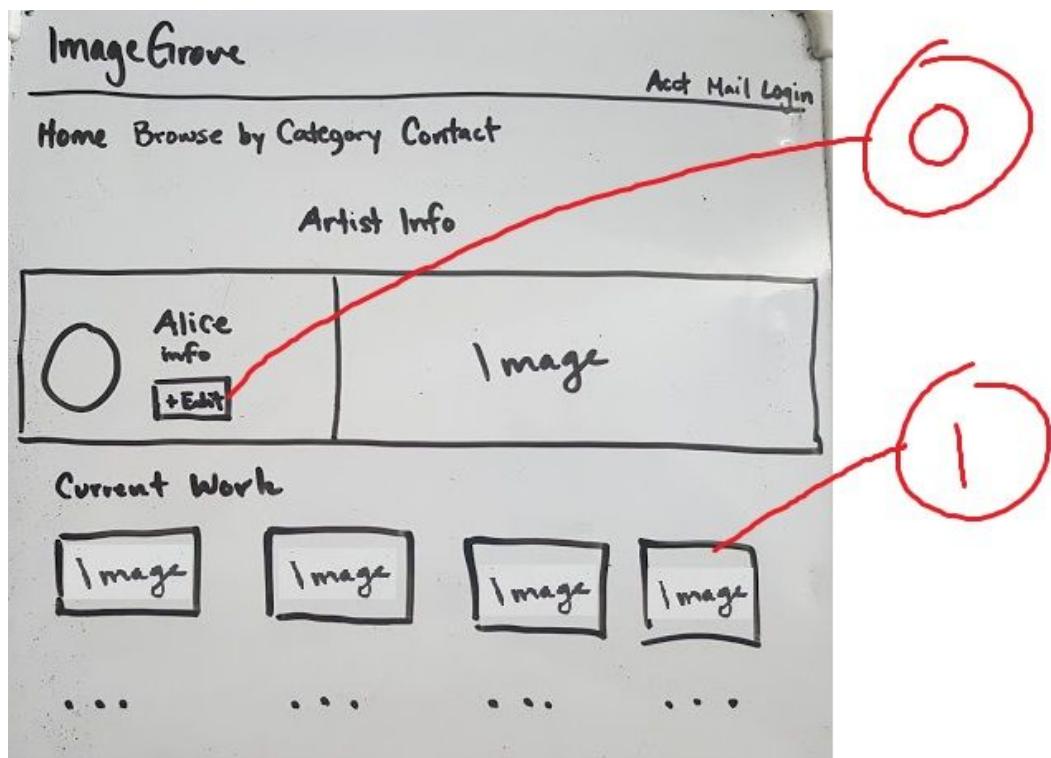
0. Clicking on return to account returns admin to account information page
1. Clicking on view profile takes admin to artist's profile page
2. Clicking on block button takes admin to blocking user form

## 12. Blocking Artist Form

The image shows a hand-drawn wireframe of a web page titled "Image Groove". At the top right is a "ACCOUNT LOGOUT" link. Below it is a link "[Return to List of Artists](#)". The main title "Blocking Artist" is centered above a large rectangular input field. To the left of this field is the label "Reason:" with a circled number "1" next to it. Below the input field is the label "Blocked Until:" with a circled number "2" next to it. Underneath this is a smaller rectangular input field containing "MM/DD/YYYY". At the bottom right is a button labeled "Submit" with a circled number "3" next to it.

0. Clicking on return to account returns admin to account information page
1. Clicking on view profile takes admin to artist's profile page
2. Clicking on block button takes admin to blocking user form
3. Clicking on submit button completes artist block and returns admin to account information page

### 13. Artist Account Page



#### PAGE FOLD

0. Clicking on edit button allows an artist to edit their profile (current page)
1. Clicking on an image will take the user to the image detail page for that image.

#### 14. Artist Account Page (after page fold)

A hand-drawn diagram of an artist account upload form. The form includes fields for 'Upload Image' (with a 'Choose File' button showing 'No file chosen'), 'Title' (text input field), 'Description' (text input field), 'License Prices' (with 'Web', 'Print', and 'Unlimited' dropdown menus all set to '\$'), and a large 'Upload' button. Red numbers 0 through 4 are circled and connected by lines to specific form elements: 0 points to the 'Choose File' button, 1 points to the 'Title' input field, 2 points to the 'Description' input field, 3 points to the 'Unlimited' price dropdown menu, and 4 points to the large 'Upload' button.

0. Clicking “choose file” button opens file selection menu for artist to choose an image
1. Artist can enter a title for the image he or she is uploading
2. Artist can enter a description for the image he or she is uploading
3. Artist can enter in the prices for the available licenses
4. Clicking “upload” button submits the form and uploads the image to the database

## 15. Competitive analysis:

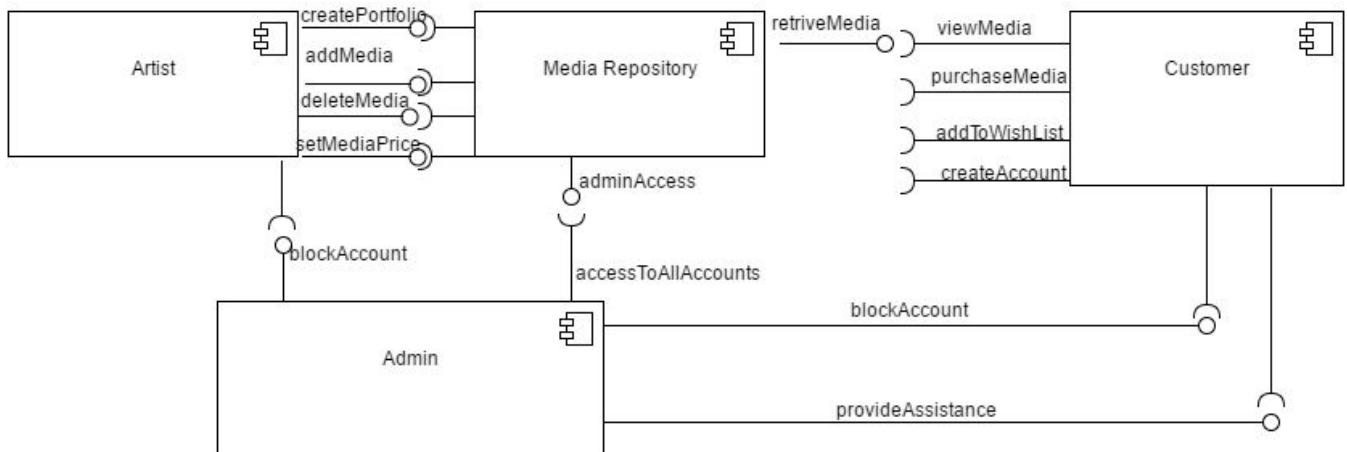
Features	ImageGrove	Snapwire	GettyImages	Fotolia
Easy to use UI/Mobile-Friendly	+	+	+	-
Keyword Tags/Search by Text	+	+	+	+
Photographer Info	+	+		
Pricing of Images	+			-
Social Integration	+	+	+	-
Video Upload	+	+	+	+

**Key:** “+” superior feature exists; “-” feature exists not superior; blank, feature does not exist

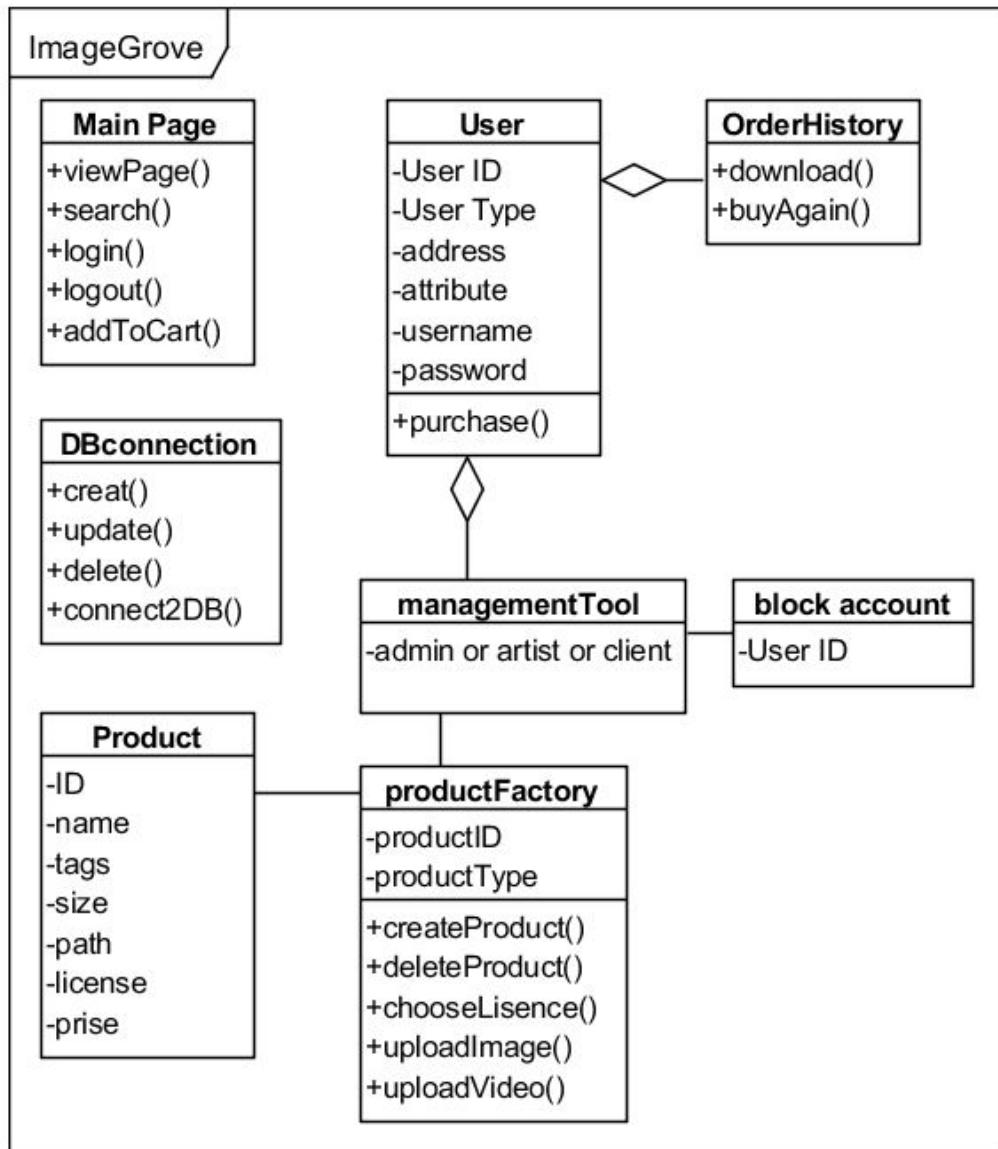
We plan to use simple user interface through the use of Bootstrap. All three competitors have eye-catching, but simple landing pages with featured image categories as well as a search bar clearly visible as soon as the front page loads. We would like to apply a similar, engaging style to attract users to our website. To allow customers to find the image(s) they are looking for, there will be an image search by use of keyword image tags. Some websites offer artist information, while others only provide a keyword search based on the artist's name. We want promote a more personal relationship between the customers and the artists and therefore want to show a short profile for each artist with information such as background, portfolio, contact information, etc. ImageGrove will allow artists to specify the price of each image, while most other websites have preset pricings for artist contributions. Fotolia in particular has a credit system where an artist earns credits for each purchase of his or her photos, which complicates the system and could be inconvenient for artists. In terms of social integration, all the websites we visited had some form of social media sharing. We would also like to incorporate social media sharing as it seems standard. All of the three websites have the option of uploading video, which as before with the social media we would like to be able to implement to keep up with the competitors.

## 16. Class and Data Definitions (UML diagrams)

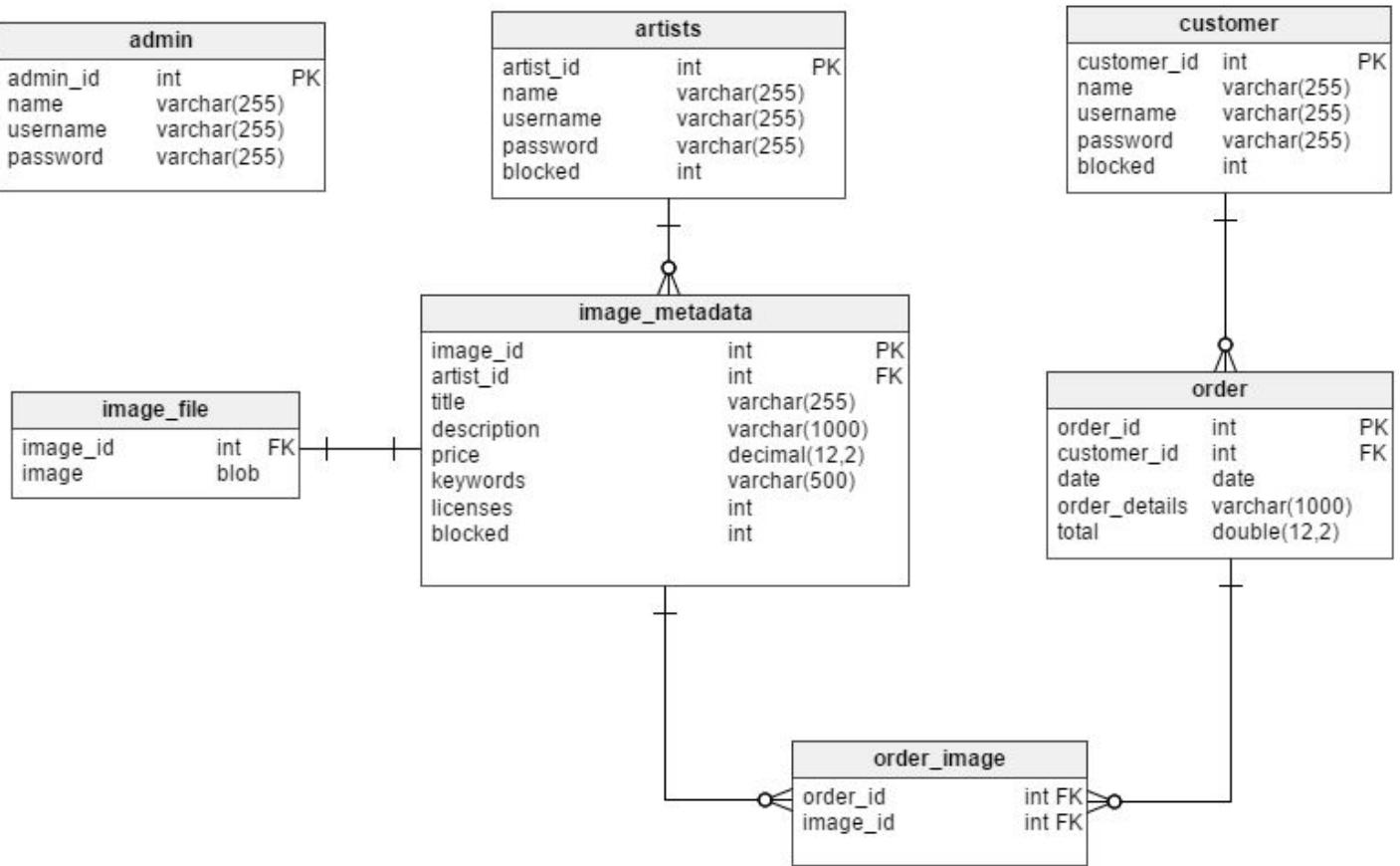
### Component Diagram



## Class Diagram



## Database Schema



## **17. High-level system architecture:**

This following section will describe the tools, frameworks, and APIs being used in assisting our development of this project and how they will tie into the different components that brings this application together.

### **LAMP**

Our application will be developed following the traditional LAMP model environment that consists of the following:

- Linux
- Apache
- MySQL
- PHP

### **Frameworks and APIs**

As for the front-end of our application, we'll be utilizing the following frameworks to assist in our development:

- Twitter Bootstrap
  - A web development framework that allows us to get started quickly by providing a series of useful and tasteful templates and themes.
- jQuery
  - A JavaScript library that eases cross-browser development using its own API.

### **Supported Browsers**

Our application shall be able to run smoothly in the following browsers:

- Google Chrome 49.0.2623.75, 48.0.2564
- Mozilla Firefox 44.02, 44.01

## 18. Key Risks

**Skill Risks:** Our team is somewhat deficient in specific knowledge of PHP. This means a greater emphasis on research, and supporting each other will be necessary. In addition, the entire group was assigned tutorials to help get everyone on the same page and up to speed. These tutorials included HTML/CSS, Bootstrap, PHP, jQuery, JavaScript, and Form manipulation. This was during the first assignments at the beginning of the semester, and proof of completion was given by predefined dates.

**Schedule Risks:** With the prioritization given in our Functional Specs, there are some risks that we committed too much to our final project. However, for now we have under-promised, with priority 1 being only the basics, so we will have a functioning Web site.

**Technical Risks:** Due to some of our skill risks, some features may be harder to implement than we are anticipating.

**Teamwork Risks:** Lack of face-to-face time means we must work harder than usual to stay in touch, and keep organized and motivated as a team. Our team is geographically distant from each other, and members have many obligations competing for time with team meetings. This means we must be more flexible and coordinated, so that we can effectively prosecute the project even if we are not in close proximity.

**Legal Risks:** When it comes to creating content, copyright is always an issue. In order to combat this, we have researched free content, and all of our images will come from royalty free sites such as [www.gettyimages.com](http://www.gettyimages.com). Aside from royalty free websites, all our creations will be heavily researched and scrutinized. We will do our best to create anything we need to prevent copyright issues.

**19. Team Configuration:** list student names, name of the group leader, and initial roles for each member)

<b>Team Member</b>	<b>Position</b>	<b>Role</b>
Sam Gluss	Team Lead	Front End
Marisa Abril	Tech Lead	Back End
Harry Cao		Back End
Meghana Dayananda		Front End
Edilson (Eddie) Araujo		Full Stack

# **Software Engineering CSC648/848 Spring 2016**

## **ImageGrove - Milestone 3 Document**

Team 7:

Samuel Pope Gluss - sgluss@mail.sfsu.edu

Meghana Dayananda

Marisa Noelani Abril

Edilson J Araujo

Harry Cao

04/22/2016

Revision History:

Revision #	Date	Description
1	4/22/16	M3 tech demo

Checkpoint Due 4/24/16 23:59:59 by email

1. Detailed description of how each class will be used in the system. The classes should reflect what is actually already programmed, and any further classes envisioned by the team. Keep it simple. I expect approx 1 paragraph per class. This description should be included in your M3 document.

### **DBConnection Class**

This class has the Database config properties and methods to get/close the database connection. This class will be used in other classes where we need a database connection.

### **ProductInfo Class**

ProductInfo class has the following methods to access database.

- `getPriceByLicense(MediaID, LicenseID)` This Method gets the price for a given mediaID,LicenseID from database and returns it.
- `deleteProductByID(productID)` This Method deletes the product with the specified ID from database.
- `getProductImageByID(MediaID)` This Method gets the product with specified ID from database.
- `getThumbNails()` This Method fetches all the thumbnails from the database .
- `getproductInfoByID(MediaID)` This Method gets all the columns related to the specific MediaID from database.

### **ArtistInfo Class**

ArtistInfo class has the following methods to access database

- `getArtistInfo(artistID)` This Method fetches the row corresponding to the given artistID.
- `blockArtist(artistID)` This Method updates the corresponding artist row in database as blocked.
- `authenticateArtist(artistID)` This method checks if the userName and password of the given artistID match with the one in the database.

- ArtistProfileController  
Class which organizes data for presentation on artist portfolio page
- getArtistDataFromController() method which organizes data for various views in Artist profile page.
- ArtistProfileModel  
Class which organizes data for presentation on artist portfolio page
- getArtistDataFromDB() method which gets data from database necessary to populate artist profile page.

### **OrderInfo Class**

- getOrderInfo(userName) This Method fetches all the order related data corresponding to the given userName.

### **CustomerInfo Class**

- getCustomerInfo (CustomerID) gets the customer related data from database corresponding to the given CustomerID
- blockCustomer(CustomerID) This Method updates the corresponding customer row in database as blocked.
- authenticateCustomer(CustomerID) checks if the userName and password of the given CustomerID match with the one in the database.

2. The overall architectural design pattern your team is using. It is OK to not use MVC, and saying that it will be “Object Oriented PHP” is sufficient.

We are using object oriented PHP.

3. Detailed description of the roles for the next two to three weeks of each team member, and their expected contribution. This should be approx. a paragraph for each team member.

Sam

I will be finishing work on artist portfolio page display. I am also helping coordinate the team to meet changing requirements, shifting focus to our backend and organization. I have been

working on improving file system structure, and integrate MVC like class structure. I will meet with Marc to ensure that the trajectory of the project will enable us to satisfy our requirements by the specified deadlines. As the semester closes, I will make sure the team stays organized, refining the site to provide a high quality product.

#### Meghana Dayananda

I will be working on database. This will include uploading data to the database and working on ArtistInfoClass, ProductInfoClass and Dbconnection class. I will be helping the team with database queries or any issues they face with respect to the project. I have been helping the team to stay organized, finding places where other team members can help to deliver maximum effect.

#### Marisa Abril

I will be working on connecting backend to frontend. I will be working on database and the queries to the database and displaying data retrieved through ArtistInfo class, ProductInfo class to the appropriate pages. I will also work on login and auth. I have been providing technical support to other members of the team to make sure that the development process is moving smoothly.

#### Eddy

I will be working on improving our UI. The pages I'm working on are the Home, Search Results, Image Details, Contact, Register, Login, About us, and FAQ pages. Additionally I am also in charge of up keeping our SVN repository so that all of our code is well maintained but also so our commit messages stay complete and descriptive of the work being done. I will help with connecting the backend which is being developed to the frontend. Specifically when it comes to the way the images are being displayed and how we are getting them from the database. I will make sure our logic is as efficient as possible so that all the pages load correctly and in a timely manner.

#### Harry

I have been working on CustomerInfoClass and OrderInfoClass to help flesh out the backend of the application. I added an include file which made the header generic, and easy to add to any page on the site. This also means any changes to the header and title bar are applicationwide, making maintenance and stylistic updates on the site easier. As the semester moves on, I will continue to support the team in connecting the backend, and ensuring that our code conforms to best practices.

# **Software Engineering CSC648/848 Spring 2016**

## **ImageGrove - Milestone 4: Beta Launch**

Team 7:

Samuel Pope Gluss - sgluss@mail.sfsu.edu

Meghana Dayananda

Marisa Noelani Abril

Edilson J Araujo

Harry Cao

05/06/2016

Revision History:

Revision #	Date	Description
1	5/6/16	Changes made from class meeting
0	5/5/16	Initial document

## **1. Product summary**

ImageGrove is a high performance product that helps get artists' work recognized, and purchased, by clientele around the world. The system features an advanced search system that lets users find exactly what they're looking for, with neatly listed search results. Once a user finds an image they are interested in, they either get purchasing information immediately, or learn more about the art piece, and discover more about the artist who made it. At the artist page, users can see all the different pieces by an artist, and even contact the artist if they would like to make a special request.

For the artist, using this application is like going to the familiar shop on the corner, where everything is within easy reach, and comfortable to access. Login requires only an email and a password, then the artist is redirected promptly to their portfolio page, where they can manage their account. They can easily survey their portfolio, and add or remove images easily with the click of a button. Upload buttons are conveniently placed at the top and bottom of the portfolio, so that the artist can add a new image without any disagreeable scrolling. Should an image suddenly strike an artist as being unpalatable, they can quickly strike it from the system: each piece has a delete button right under it. One click in the confirmation window is all it takes to clean the portfolio up. Should an artist decide to check out the competition, they can investigate the search results and portfolios of others on the application to get a feel for the zeitgeist of ImageGrove.

The value of this application is derived from its intuitive user interface as well as its responsive performance, making it an addictive tool to explore the latest artistic creations of a digitally-attached generation. So, come give it a try now at <http://sfsuswe.com/~s16g07/>, and see what you'll find today!

## **2. Usability test plan**

### **2.1 Test Objectives**

For usability testing, we have decided to test the usability of the Search Function on our website's home page. Since our website is largely about works of art – which, of course, have to be found first – we think that it's important to make sure that the simple yet appealing UI design as promised in the Product Summary works for the user's advantage when searching for potential images. That is why we want to verify that ImageGrove truly provides its users a smooth and efficient working Search function that will make their search quick and dynamic.

The users will be appointed search related tasks in order to test the functionality of the Search function used to look for works of art uploaded by artists. The users shall start their testing from our home page just as any other new user. Upon searching for specific terms, images that somehow fit into the category that was searched for will be displayed. The user will find their way without any assistance to an Image Details page where they would be able to buy an image that was displayed during the search. This will test if the Search function is helpful for any user who wants to look for an image of a specific category. We will then ask questions about the easiness and duration of the tasks and the user's experience on our website with room for comments.

### **2.2 Test Plan**

System Setup	Windows 10 computer running Chrome 50.0.2661.94
Starting Point	<a href="https://sfsuswe.com/~s16g07">https://sfsuswe.com/~s16g07</a>
Task 1	Find an image of a cat
Intended User	Anybody over the age of 16
Successful Completion Criteria	User is able to locate each image in under 1 minute.

URL	<a href="http://sfsuswe.com/~s16g07">http://sfsuswe.com/~s16g07</a>
-----	---

## 2.3 Questionnaire

Please circle the answer that best describes your feelings towards the given statements.

### Question 1: The Website's interface was intuitive and easy to use.

1	2	3	4	5
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Comments:

### Question 2: The Search bar location was useful.

1	2	3	4	5
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Comments:

### Question 3: The Search bar location made searching easier.

1	2	3	4	5
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Comments:

### Question 4: The website's search function returned relevant search results

1	2	3	4	5
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Comments:

### Question 5: It was easy to find more images by a specific Artist.

1	2	3	4	5
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Comments:

### Question 6: If I was looking for images, I would gladly use ImageGrove.

1	2	3	4	5
Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree

Comments:

### **3. QA Test Plan**

#### **3.4 Formal QA Test Plan**

##### **3.4.1 Test Objectives**

The objectives of this test are to ensure the performance and capabilities of ImageGrove satisfy the requirements as laid out in the milestone document, requirements sections 4 and 5. The purpose of the beta release is to demonstrate the satisfactory performance of as many key priority one features as possible, and determine if there are any prevailing issues in performance or features that demand immediate attention as deadlines draw closer. The principal features which are available in the application at present are image searching and artist portfolio management. These are priority 1 features of great importance to the project, so it is of importance to evaluate their performance, functionality, and ease of use. The bottom line is that performance and usability can determine whether a user embraces our application, or goes to the competitor, so it is our aim to eliminate our weaknesses to provide the best product possible to our client. Crashes other software errors should be as nonexistent as possible at this phase, and the UI should be quick and responsive. The software should give feedback to user action, and do so promptly.

##### **3.4.2 HW & SW Setup**

Testing will be done in Chrome browser 50.0.2661.94, on desktop environments only.

Chrome utilizes the V8 javascript engine, so it is possible that certain browser configurations may have incompatibilities with some of our client side code. The server runs PHP 5.3 on an apache server, utilizing MySQLi for database management. This configuration remains constant for all client configurations. The server machine is a common AWS EC2 instance running Ubuntu 14.04. Front end frameworks in use are

Twitter Bootstrap and Jquery, which are well known for their ease of use with regards to browser compatibility, but which can cause unintended slowdowns.

### 3.4.3 Feature to be Tested

The search feature will be tested in this document. This is the most important feature on the application, because it connects the user, who is trying to buy image licenses, with the artist, who is trying to sell them. This is the very core of what this application is trying to deliver as a service to our client, and so it is crucial that it perform exceptionally in every category. It is desirable for the search function to be easy to access and use, and subsequently quick and accurate with its delivery of results. These results should be in a sensible grid style layout on the page, without any glitches or distracting features.

### 3.4.4 Test Cases

#### Search Function:

#	Actions	Expected Results	Pass/ Fail	Comments and Observations
1	Navigate to <a href="http://sfsuswe.com/~s16g07">http://sfsuswe.com/~s16g07</a>	ImageGrove homepage promptly loads (less than 6 seconds)		
2	Click on the search bar in the center of the screen	Cursor appears in search bar, search bar changes color in response to focus change		
3	Enter 'cat' in search, press enter	Search results page loads in under 6 seconds, page reports 3 images found. Grid of kitten images should appear.		
4	Enter 'dog' in search bar, which is above the search results. Press enter key	Search results page loads in under 6 seconds, page reports no search results found. User is prompted to try again with different search		
5	Enter 'a' in search bar, press magnifying glass icon next to search bar, or enter key	Search results page loads in under 6 seconds, page reports 13 hits. Images are grid viewed correctly		

6	Hold keyboard 'control' button and zoom out with mouse wheel to 50%	See that search results spread out to fill horizontal space appropriately		
7	Hold keyboard 'control' button and zoom in with mouse wheel to 150%	See that search results retile so they don't go over the screen edge		
QA test completed by:		Date:		

### 3.5 Formal QA Test Results

**Search Function test, performed:** 23 year-old female with some college education and non-technical background.

#	Actions	Expected Results	Pass/Fail	Comments and Observations
1	Navigate to <a href="http://sfsuswe.com/~s16g07/">http://sfsuswe.com/~s16g07/</a>	ImageGrove homepage promptly loads (less than 6 seconds)	P	Fast
2	Click on the search bar in the center of the screen	Cursor appears in the search bar, search bar changes color in response to focus change	P	Works
3	Enter 'cat' in search, press enter	Search results page loads in under 6 seconds, page reports 3 images found. Grid of kitten images should appear.	P	Glass Icon not in the described location. Search results for cats works as described.
4	Enter 'dog' in search bar, which is above the search results. Press enter key	Search results page loads in under 6 seconds, page reports no search results found. User is prompted to try again with different search	P	No results for 'dog' as described and prompted to try again. Strange graphical artifact on header bar.
5	Enter 'a' in search bar, press magnifying glass icon next to search bar, or enter key	Search results page loads in under 6 seconds, page reports 13 hits. Images are grid viewed correctly	P	Graphical artifact identified as search results not displaying correctly. 13 hits
6	Hold keyboard 'control' button and zoom out with mouse wheel to 50%	See that search results spread out to fill horizontal space appropriately	P	Images repositioned correctly.

7	Hold keyboard ‘control’ button and zoom in with mouse wheel to 150%	See that search results retile so they don't go over the screen edge	P	Images dont go over the screen edge
QA test completed by: Ana Mora		Date:	5.5.16	

## 4. Code Review

### 4.1 Coding Style

We have used object oriented programming style and shall adopt the following coding convention practices:

- No mixed-up tags
- No missing or mismatched end tags
- No missing items (some tags, quotes, ...)
- All class names shall begin with a Capital letter.
- All variable names shall be in camelCase.
- All Method names shall be in camelCase.
- All constant names shall be in Capital letters.
- Code shall be indented according to the below convention.

Number of spaces per indent :4

Tab size :8

Right margin:80

Continuous indentation : 8

Array declaration indentation:4

- Code shall include comments at necessary places.
- Any changes made in the code shall be preceded with the appropriate comments describing the purpose for the change .
- Database table names shall be in UpperCamelCase.

## 4.2 Chosen Code - Filtered Search

Snippets from ‘search-results.php’ (Full file can be found:  
[http://sfsuswe.com/svn/s16g07/m3/public\\_html/search-results.php](http://sfsuswe.com/svn/s16g07/m3/public_html/search-results.php))

```
<!DOCTYPE HTML>
<!--code review 5/6/2016: Name of the file needs to be changed to reflect the functionality-->
<!-- As per code review (5/6/16): File name changed to search-results.php -->
<html>
    <?php
        $page_title = "Search Results";
        include("./includes/head.php");
        include("./app/controllers/SearchResultsController.php");
    ?>
    <!--code review 5/6/2016: style tags should be moved to main css file-->
    <!-- As per code review(5/6/16): CSS moved to compare-modal.css -->
    <link rel="stylesheet" type="text/css" href="css/compare-modal.css">
<body>
    <?php
        $home_page = "false";
        include("./includes/header.php");
        if (!isset($_GET['search_string'])) {
            $searchResults = SearchResultsController::browseAll();
        }
        else {
            $searchResults = SearchResultsController::searchResults($_GET['search_string']);
        }
        $numOfResults = count($searchResults);
    ?>
    <!--Code review:5/6/2016 :dead tag should be closed or removed-->
    <!-- As per code review (5/6/16): removed dead tag -->
    <div class="col-md-2 stock_left">
        <button class="accordion">Compare</button>
        <div class="panel">
            <?php
                echo "<br />Select <b>two</b> images to compare: </ br>" . "<form name='compare' id='compare' action='./image-Compare.php' method='GET'>" . "<input class='single-checkbox' type='checkbox' name='compare[]' value='{$searchResults[$i]->getMediaId()}' onclick='chkcontrol({$i});' />" . "</br>" . "<br /><button style='margin:15px 0 0 0; padding: 6px 18px;' class='btn_3'disabled='disabled' id='submitBtn' type='submit'>Compare</button>" . "</form>";
            ?>
            <div class="clearfix"> </div>
        </div>
    <!--Code review:5/6/2016 :Probably it is a good idea to move js functions to separate file-->
    <?php include("./includes/footer.php"); ?>
    <!-- As per code review (5/6/16): Moved javascript into separate file -->
    <script type="text/javascript" src="js/search-results/compareButton.js"></script>
</body>
</html>
```

## Email Correspondence

5/6/2016

Re: [CSC 648] M4 Code Review: Search Results Page - Marisa Noelani Abril

Re: [CSC 648] M4 Code Review: Search Results Page

Marisa Noelani Abril

Fri 5/6/2016 1:02 PM

To: Meghana Dayananda <mdayanana@mail.sfsu.edu>;

Hi,

Thank you for the input! I have made changes according to your comments and committed them to SVN.  
You can view the changes here: [http://sfsuswe.com/svn/s16g07/m3/public\\_html/search-results.php](http://sfsuswe.com/svn/s16g07/m3/public_html/search-results.php)

Thank you again,  
Marisa Abril

---

**From:** Meghana Dayananda  
**Sent:** Friday, May 6, 2016 11:46 AM  
**To:** Marisa Noelani Abril  
**Subject:** Re: [CSC 648] M4 Code Review: Search Results Page

Hi Marisa,

The code looked good except for some minor changes in coding conventions. I have included the review comments in code([http://sfsuswe.com/svn/s16g07/m3/public\\_html/stock.php](http://sfsuswe.com/svn/s16g07/m3/public_html/stock.php)).

-Thanks  
Meghana Dayananda

---

**From:** Marisa Noelani Abril  
**Sent:** 06 May 2016 23:50:49  
**To:** Meghana Dayananda  
**Subject:** [CSC 648] M4 Code Review: Search Results Page

Hi Meg,

I have checked in the current version of the search results page ([http://sfsuswe.com/svn/s16g07/m3/public\\_html/stock.php](http://sfsuswe.com/svn/s16g07/m3/public_html/stock.php)). Please take a look at the code and let me know if any changes are needed.

Thank you,  
Marisa Abril

<https://outlook.office.com/owa/?viewmodel=ReadMessageItem&itemId=AAMkAGNmYVY5Njk4LTk4ZTk1NDdINS1hNDI4LWJIMmY5Yzg4N2ExMQBG...> 1/1

# Product Screen Shots

Landing Page:



## Featured Images

*Popular*



*New Releases*



## *Image Grove Categories*

Animals

Fashion

Landmarks

Celebrities

Holidays

Education

Objects

Miscellaneous

Model

Nature

Parks

Science



## Search Results for “Cats” (with Pagination):

Image Grove Search for images  HOME | SUPPORT | SIGN IN

Showing 1-8 of 11 results for "CATS"

Compare ▼

Previous 1 2 Next



**cutie cat**  
ID#322  
Price: 10.00 (Web)

**Buy**



**Cats**  
ID#323  
Price: 10.00 (Web)

**Buy**



**Sleepy cat**  
ID#326  
Price: 11.00 (Web)

**Buy**



**Kitten**  
ID#382  
Price: 10.00 (Unlimited)

**Buy**



**Lazy cat**  
ID#384  
Price: 11.00 (Web)

**Buy**



**Kitten**  
ID#385  
Price: 11.00 (Web)

**Buy**



**Cute Kittens**  
ID#386  
Price: 11.00 (Web)

**Buy**



**Cat in basket**  
ID#387  
Price: 11.00 (Web)

**Buy**

Previous 1 2 Next

### Image Grove Categories

Animals	Celebrities	Objects	Nature
Fashion	Holidays	Miscellaneous	Parks
Landmarks	Education	Model	Science

## Search Results (after clicking on a listed Category):

Image Grove Search for Images

HOME | SUPPORT | SIGN IN

Compare ▾

SHOWING 1-2 OF 2 RESULTS FOR "FASHION"

Previous 1 Next



Jewels  
ID#353  
Price: 11.00 (Web)



Red Dress  
ID#362  
Price: 10.00 (Web)

Previous 1 Next

**Image Grove Categories**

Animals	Celebrities	Objects	Nature
Fashion	Holidays	Miscellaneous	Parks
Landmarks	Education	Model	Science

Terms of Use Pricing Plans Support About Us FAQ Categories

## Pricing/Licensing Options:

The screenshot shows the 'LICENSING OPTIONS' section of the Image Grove website. It features three main categories: 'Unlimited', 'Web', and 'Print'. Each category has a list of features and a 'Register' button.

LICENSING OPTIONS	Unlimited	Web	Print
<ul style="list-style-type: none"><li>✓ Access to all images</li><li>✓ Access to all videos</li><li>✓ Both Web and Print Licensing</li></ul>	<ul style="list-style-type: none"><li>✓ Access to all images</li><li>✓ Access to all videos</li><li>✓ Web Licensing</li></ul>	<ul style="list-style-type: none"><li>✓ Access to all images</li><li>✓ Access to all videos</li><li>✓ Print Licensing</li></ul>	
<a href="#">Register</a>	<a href="#">Register</a>	<a href="#">Register</a>	

**Image Grove Categories**

Animals      Celebrities      Objects      Nature  
Fashion      Holidays      Miscellaneous      Parks  
Landmarks      Education      Model      Science

Terms of Use    Pricing Plans    Support    About Us    FAQ    Categories

## Support Page:

Image Grove   HOME | SUPPORT | SIGN IN

### CONTACT US



[View Larger Map](#)

#### CONTACT FORM

Name...	Email...	Subject...
---------	----------	------------

Message..

#### CONTACT INFO

1600 Holloway,

San Francisco, CA,

USA

Phone : 1 415 555 1600

Fax : 1 415 555 1601

Email : support@imagegrove.com

Follow on : Facebook, Twitter

**SUBMIT**

### Image Grove Categories

Animals

Fashion

Landmarks

Celebrities

Holidays

Education

Objects

Miscellaneous

Model

Nature

Parks

Science

[Terms of Use](#) [Pricing Plans](#) [Support](#) [About Us](#) [FAQ](#) [Categories](#)

## About Us Page:

The screenshot shows the top navigation bar of the ImageGrove website. It includes the logo "Image Grove", a search bar with placeholder text "Search for Images", a magnifying glass icon, and links for "HOME", "SUPPORT", and "SIGN IN".

### ABOUT US

ImageGrove is an innovative new system which allows artists to reach their customers. This enables art buyers to find the exact digital masterpiece they are looking for. Our design is built from the ground up to provide a seamless and responsive interface that gets the user what they want. Using the latest techniques in dynamic web application programming, we can provide a product so intuitive that any consumer will enjoy using it.

Artists want the capability to upload and manage licensing for anything they create. ImageGrove also allows artists to manage their portfolios of movies and pictures. By adding descriptions to these products, customers can easily find new pieces to enjoy, and new artists to love. Our comprehensive authorization system will enable artists to manage their own accounts, customers to browse and buy, and administrators to address any issues. This streamlines the experience for everybody, making sure their account privileges offer just the features they need.



ImageGrove's social media integration means artists can easily keep in touch with their sales base. Artists are given easy control of pricing and licensing, so they'll be able to spend less time chasing sales and more time following their passion! The users will also appreciate a well designed system which quickly allows them to find just the piece they were looking for using simple but robust keyword searches. These features are crucial to retention of clientele, who expect sleek performance from the apps they use.

Our expert team of software developers are highly talented and committed to creating industry leading user experiences. Their mastery of application development means your customers will get to spend more time looking at the finest art products and giving your clients their hard-earned money!



### Image Grove Categories

Animals	Celebrities	Objects	Nature
Fashion	Holidays	Miscellaneous	Parks
Landmarks	Education	Model	Science

[Terms of Use](#)   [Pricing Plans](#)   [Support](#)   [About Us](#)   [FAQ](#)   [Categories](#)

## FAQ:

The screenshot shows the top navigation bar of the Image Grove website. It includes the logo "Image Grove", a search bar with placeholder text "Search for Images", a magnifying glass icon, and links for "HOME", "SUPPORT", and "SIGN IN".

### FREQUENTLY ASKED QUESTIONS

#### What is Image Grove?

We are a media hosting solution created to handle the issues CII has regarding the licensing and sale of artist's images.

#### Where does Image Grove get it's media?

Image Grove offers images and videos from artists from all over the world. Artists sign up for an account with Image Grove and upload their work specifying the type of license(s) their individual art piece can have and be sold with.

#### What types of licenses does Image Grove offer?

Image Grove provides three different types of licenses. One for Web distribution, one for a physical Print distribution, and one for both Web and Print. For more information on pricing and licensing please refer to our [Pricing Plans](#) page.

#### How do I sell my art work with Image Grove?

You can register for an Artist account with Image Grove and upload your work to our store to be sold. With an artist account you'll be able to upload or take down your media as often whenever and as often as you'd like. Please note that by registering for an artist account you agree to comply with our [Terms of Use](#) and will only upload images or videos that fall within our guidelines for approved media.

#### Do I need to register for an account in order to purchase?

No, you are welcome to purchase as a guest without the need of registering for an account.

#### What form of payment can I use?

We accept Visa, MasterCard, American Express and Discover credit cards as well as Bitcoin.

### Image Grove Categories

Animals	Celebrities	Objects	Nature
Fashion	Holidays	Miscellaneous	Parks
Landmarks	Education	Model	Science

[Terms of Use](#)   [Pricing Plans](#)   [Support](#)   [About Us](#)   [FAQ](#)   [Categories](#)

## Sign In/Sign Up Page:

The screenshot shows the sign-in and sign-up interface for Image Grove. At the top, there's a dark blue header bar with the "Image Grove" logo, a search bar containing "Search for Images", a magnifying glass icon, and links for "HOME", "SUPPORT", and "SIGN IN".

The main area is divided into two sections: "SIGN IN TO YOUR ACCOUNT" on the left and "NEW USER SIGN UP!" on the right. Both sections feature input fields for "Email address" and "Password", and a "Sign in" or "Sign up" button.

A central "OR" button is positioned between the two sections. Below the "SIGN IN" section is a "Keep me signed in" checkbox and a "Sign in" button. Below the "NEW USER SIGN UP!" section are radio buttons for "Artist" and "Customer", and a "Sign up" button.

At the bottom left, there's a "Image Grove Categories" link. The bottom right features a grid of category names: Animals, Celebrities, Objects, Nature; Fashion, Holidays, Miscellaneous, Parks; Landmarks, Education, Model, Science.

At the very bottom, a dark footer bar contains links for "Terms of Use", "Pricing Plans", "Support", "About Us", "FAQ", and "Categories".

## Artist's Account Page (signed in as an Artist):

Image Grove   HOME | MY ACCOUNT | SUPPORT | IRIS -

### Iris



Upload Image

Artist Description  
Hi I'm Iris, let me show you something new!

Artist Focus

- People
- Food
- Pets
- Special Occasions

More by Iris



Delete



Delete



Delete



Delete

Upload Image

### Image Grove Categories

Animals  
Fashion  
Landmarks

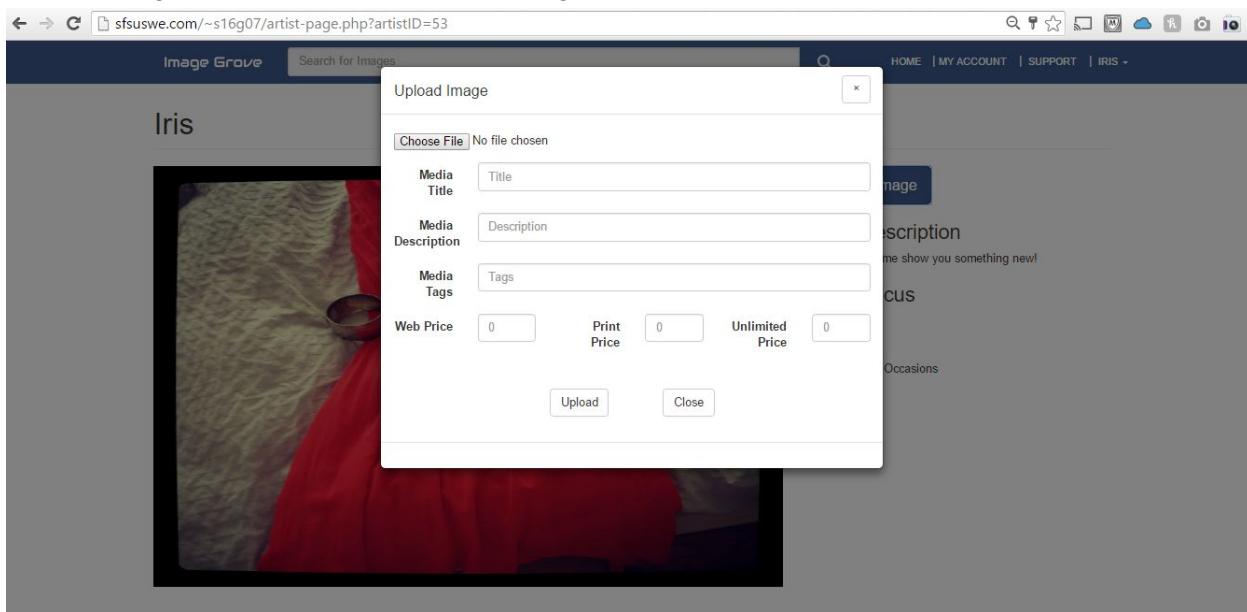
Celebrities  
Holidays  
Education

Objects  
Miscellaneous  
Model

Nature  
Parks  
Science

[Terms of Use](#) [Pricing Plans](#) [Support](#) [About Us](#) [FAQ](#) [Categories](#)

## Upload Image Modal (on Artist Account Page):



## Sign Out (on Artist Account Page):

Image Grove Search for Images   HOME | MY ACCOUNT | SUPPORT | SIGN OUT | ARTIST PAGE

Iris



Upload Image

Artist Description  
Hi I'm Iris, let me show you something new!

Artist Focus

- People
- Food
- Pets
- Special Occasions

More by Iris



Delete Delete Delete Delete

Upload Image

### Image Grove Categories

Animals	Celebrities	Objects	Nature
Fashion	Holidays	Miscellaneous	Parks
Landmarks	Education	Model	Science

[Terms of Use](#) [Pricing Plans](#) [Support](#) [About Us](#) [FAQ](#) [Categories](#)

## Image Compare Page:

Image Grove   HOME | SUPPORT | SIGN IN

### Image Comparison



Title	Kitten	Lazy cat
Artist	Elizabeth@gmail.com	Elizabeth@gmail.com
ID #	382	384
Description	Kitten	Lazy cat
Web Price	\$11.00	\$11.00
Print Price	\$12.00	\$12.00
Unlimited Price	\$10.00	\$15.00

Web Price: \$11.00  
 Print Price: \$12.00  
 Unlimited Price: \$10.00

Web Price: \$11.00  
 Print Price: \$12.00  
 Unlimited Price: \$15.00

**Image Grove Categories**

Animals      Celebrities      Objects      Nature  
Fashion      Holidays      Miscellaneous      Parks  
Landmarks      Education      Model      Science

[Terms of Use](#) [Pricing Plans](#) [Support](#) [About Us](#) [FAQ](#) [Categories](#)

## Product Details Page:

Image Grove  Search HOME | SUPPORT | SIGN IN

Back



cutie cat

Artist Name: Elizabeth

Web ID: 322

Description: cat in pink towel

Pricing:

Web Price:\$10.00

Print Price:\$20.00

Unlimited Price:\$100.00

[Buy this Photo](#)

### Image Grove Categories

Animals

Fashion

Landmarks

Celebrities

Holidays

Education

Objects

Miscellaneous

Model

Nature

Parks

Science

[Terms of Use](#) [Pricing Plans](#) [Support](#) [About Us](#) [FAQ](#) [Categories](#)

## Artist's Profile Page:

Image Grove   HOME | SUPPORT | SIGN IN

Elizabeth



### Artist Description

Liz loves cats! do you love cats?

### Artist Focus

- People
- Food
- Pets
- Special Occasions

More by Elizabeth



### Image Grove Categories

Animals  
Fashion  
Landmarks

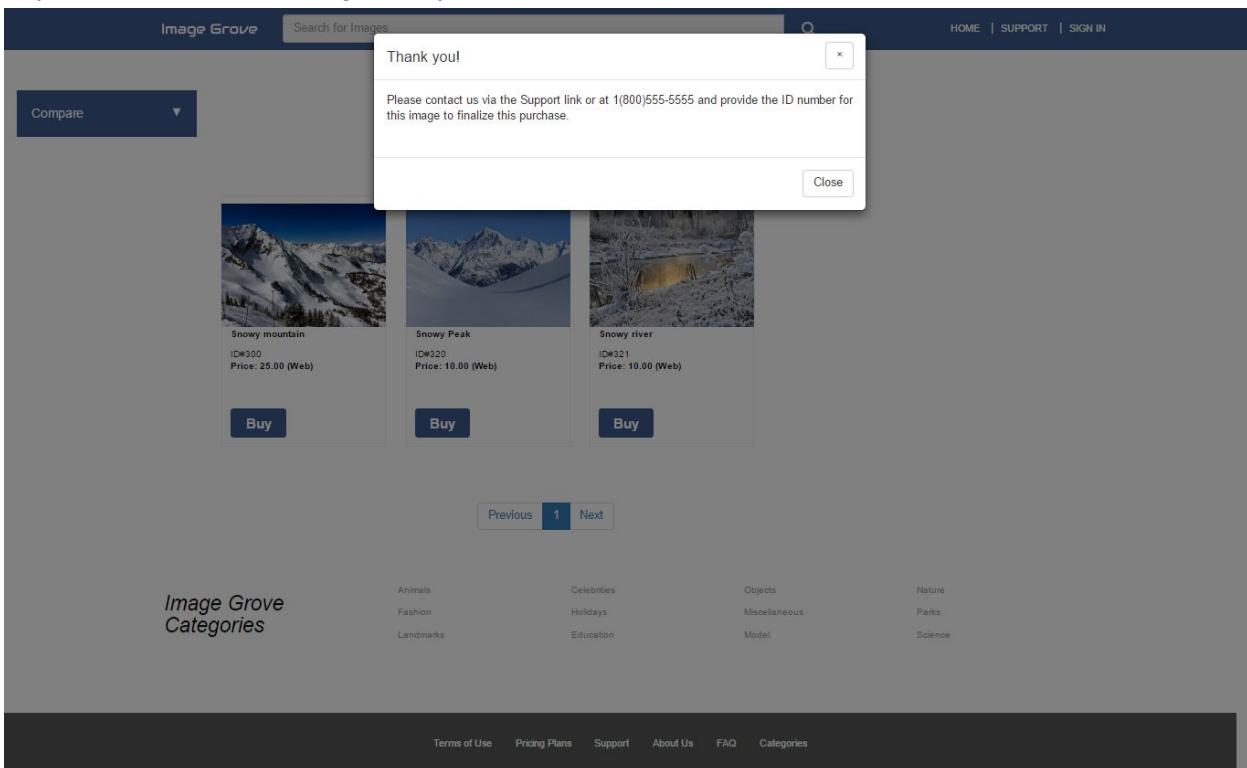
Celebrities  
Holidays  
Education

Objects  
Miscellaneous  
Model

Nature  
Parks  
Science

[Terms of Use](#) [Pricing Plans](#) [Support](#) [About Us](#) [FAQ](#) [Categories](#)

## Buy Modal Pop-up (clicking on buy button on search results):



# Team Member Contributions

Microsoft Corporation [US] | https://outlook.office.com/owa/projection.aspx

Reply all | v Delete Junk | v ...

## CSC 648 M5: Team Contributions

Marisa Noelani Abril  
Thu 23:53  
Meghana Dayananda; Samuel Pope Gluss; Edilson J Araujo; Harry Cao

Inbox

You replied on 19/05/2016 23:58.

Hello everyone,

Here is my list of contributions. Let me know if you approve.

- Researched other established websites for competitive analysis.
- Designed initial database schema.
- Revised database schema.
- Helped create and revise mock up designs.
- Expanded upon search results from our vertical prototype.
- Helped work on image details page.
- Implemented "compare" feature.
- Revised "compare" page UI.
- Reworked minor UI on search results page and homepage.

Thanks,  
Marisa Abril

Microsoft Corporation [US] | https://outlook.office.com/owa/projection.aspx

Reply all | v Delete Junk | v ...

## Re: CSC 648 M5: Team Contributions

Meghana Dayananda  
Thu 23:58  
Marisa Noelani Abril; Samuel Pope Gluss; Edilson J Araujo; Harry Cao

Sent Items

Hi Everyone ,

Thanks for sharing Marisa . Below are my contributions. Please let me know if you have any concerns.

- Worked on the requirements part of the document.
- Created Database.
- Uploaded Database.
- Worked on the Admin Throwaway prototype.
- Worked on the revised db schema diagram.
- Worked on initial version of the Compare Image Module.
- Worked on ImageDetails Page.
- Worked on db connect Module.
- Helped the team with db queries.
- Worked on MVC POC .

-Thanks  
Meghana Dayananda

---

From: Marisa Noelani Abril  
Sent: 19 May 2016 23:53:47  
To: Meghana Dayananda; Samuel Pope Gluss; Edilson J Araujo; Harry Cao  
Subject: CSC 648 M5: Team Contributions

Microsoft Corporation [US] | https://outlook.office.com/owa/projection.aspx

Reply all | v Delete Junk | v ...

## Re: CSC 648 M5: Team Contributions



Samuel Pope Gluss

Fri 11:14

Meghana Dayananda; Marisa Noelani Abril; Edilson J Araujo; Harry Cao

Inbox

Looking good, guys!

I was the team lead, and these were my contributions:  
-rebuilt artist page, incorporated artist logged-in site functionality  
-helped develop pseudo-MVC architecture  
-I assisted in the coordination of teamwork  
-I made sure that the changing demands of our principal were recorded and that proper responses were organized in a timely fashion  
-I helped populate the database with images, and reconfigured data as needed  
-I helped develop the documentation, and made sure team members had an easy time contributing to collective docs  
-I improved the robustness of the site by finding weak points through ongoing smoke testing

From: Meghana Dayananda

Sent: Thursday, May 19, 2016 11:28:18 AM

To: Marisa Noelani Abril; Samuel Pope Gluss; Edilson J Araujo; Harry Cao

Subject: Re: CSC 648 M5: Team Contributions

Hi Everyone ,

Thanks for sharing Marisa . Below are my contributions. Please let me know if you have any concerns.

- Worked on the requirements part of the document.
- Created Database.

Microsoft Corporation [US] | https://outlook.office.com/owa/projection.aspx

Reply all | v Delete Junk | v ...

## Re: CSC 648 M5: Team Contributions



Harry Cao

Fri 19:31

Samuel Pope Gluss; Meghana Dayananda; Marisa Noelani Abril; Edilson J Araujo

Inbox

Hi team,

Thanks a lot for your hard work throughout the whole semester.  
It's a great pleasure to have the opportunity to work with you guys.  
You guys are awesome!

These were my contributions:

- Designed class diagram
- Implemented "login" and "register" feature
- Implemented "popular" and "new releases" feature
- Revised class diagram
- Revised "login" page UI and functionality
- Revised "search bar" UI and functionality
- Revised "header" "footer" pages
- Worked on CustomerDetails Page
- Worked on artist page prototype
- Incorporated db connect Module

Please let me know if you have any questions.

Best regards,  
Harry Cao

Microsoft Corporation [US] <https://outlook.office.com/owa/projection.aspx>

Reply all | [Delete](#) Junk | ... [X](#)

RE: CSC 648 M5: Team Contributions

 Edilson J Araujo  
Fri 20:43  
Meghana Dayananda; Marisa Noelani Abril; Samuel Pope Gluss; Harry Cao

Inbox

Hi,

Here is my list of contributions:

- Designed and implemented search algorithm for vertical prototype
- General UI implementation
- Kept Subversion commits organized and well documented
- Created QA test plan
- Helped work on Artist profile page UI and implementation
- Worked on general UI fixes for whole site
- Defined the high level system architecture
- In charge of server-side organization and deployment
- Worked on search results page UI

Thank you,  
Eddie

## Main Challenge

We started the semester with an extremely strong team. Between us, we had plenty of experience with team projects, internet applications, and object oriented programming. As we were developing our front-end UI architecture and flow, we did not allocate sufficient resources to ensuring that we had a robust and flexible backend ready to go for data storage and delivery. Everyone was under the impression that we are required to deliver the throwaway horizontal prototype alone. The entire team focused on delivering the user interface. Had we read the expectations posted on ilearn we could have overcome this and would have been right on track.

We will be more careful going forward to make sure everyone on the team knows the expectations of each milestone in every project we are involved in. It was a very good learning experience for all of us. With the help of the professor's feedback on milestone 3, we put in extra hours and made sure we delivered all the expectations to the best of our ability right on time. In addition, MVC also provided a challenge for us as no one on the team had experience with MVC, specifically MVC using PHP. Considering our deadline for the final project, we decided to use Object Oriented PHP with MVC concepts as a guideline to better organize our classes and file directory.

By creating a standard for MVC architecture within the team, we were able to create consistent code across many developers, and rapidly fill out the website with dynamically generated content. MVC enables us to implement the DRY (Don't Repeat Yourself) software design principle. What this means is that wherever we need to implement a new piece of dynamic content, that view will only need to be created once, then simply included wherever that content is needed.