

Software Engineering CSC648/848 spring 2016

Project Name: “Pixit”

Milestone 2

Group 6

Andrey Barsukov - Team Lead (email): andr3y3@gmail.com

Nan Den -Tech Lead

Matvey Voishchev

Evan Edge

Anthony Wong

Ammar Naqvi

Date: March 17, 2016

Revision History:

Revision Number	Date	Description
1	3/4/2016	Initial Document
2	3/4/16	Second Revision following SCRUM meeting with Sosnick
3	3/14/16	Final M1 revision, frozen doc
4	3/18/16	First M2 revision
5	3/20/16	Revision after SCRUM with Sosnick on 3/18/16

Contents

Executive Summary:	3
Use Cases:	4
1. Print Ordering Notification:	4
2. Photo geolocation use case:	4
3. Customer Print Ordering use case:	5
4. Administrator use case:	5
5. Photo geolocation customer use case:	6
6. Subscription use case:	6
Glossary:	7
Functional specs:	9
Non-Functional specs:	11
Application Screenshots and Storyboard.....	12
Competitive Analysis	19
High-level system architecture:	20
Architecture Design.....	21
Content Management:	22
Search Algorithm and Prioritization:	23
Additional Tools and Software:	23
LAMP Stack	23
Class and Data Definitions.....	25
UML and UML component Diagram.....	25
ER diagram and DB structure	27
Data content and basic Data types	28
Key Risks.....	29
Skill	29
Scheduling	29
Technical	29
Legal/Content.....	29
Team Configuration:	30

Executive Summary:

We are Pixit. We specialize in creating a sleek and modern website with both the artist and customer in mind. Our website will allow your artist to create a personalized portfolio which can be managed with ease, and will focus on making the user experience as effortless as possible.

CII has an impressive collection of digital media from a wide variety of talented artists. Art that many collectors and companies would love to use. The problem is that there is no easy way for collectors to purchase this media. CII currently have no web presence for its artists to showcase their talents.

As you are well aware, manually placing orders with phone calls is limiting your potential for artist exposure and profit gain. Your artists need an easily accessible portal to showcase their work and customers need a convenient way of acquiring digital works. Each of your artists is uniquely gifted and deserves to have their artwork reach the widest audience possible.

Each artist will have a personal account which they can manage and customize through a variety of options. Artists will be able to upload their digital images and videos directly to our secure server and store their work. Besides being able to add detailed descriptions of their art, artists will be able to add a geolocation to images and have a map displayed showing the origins of that image. Besides just selling digital works, artists will have an option to advertise custom prints.

In order to attract a wide range of customers, the website will provide customers with plenty of options to browse your artists' collections. Customers will be able to browse image galleries based on several filtering options such as popularity, new additions and categories. Customers will have the option to easily share images through an email link and purchase art for others through a gift buying feature. Lastly, since many customers will want to customize the images they purchase to fit their own needs, we will create a watermarked version of images that can be downloaded as samples.

Pixit will to help you become the best and biggest distributor of digital art on the web. This proposal outlines in more detail how we will accomplish that and what tools we will use to create your ecommerce marketplace.

Use Cases:

1. Print Ordering Notification:

Sarah is a photographer who sells her photos through CII. She sells a large number of photos every month, both in the form of **prints** and digital licensing. She is very busy with her photography work, so she doesn't want to spend very much time uploading photos to CII, and doesn't want to look at the website much unless she has to.

Since she sells **prints** as well as digital licenses, sometimes she needs to ship the **prints** from her studio, which means she needs to be notified quickly when a print is purchased. Sarah opens up a browser and navigates to Pixit.com. From the landing page, she clicks the Artist Log In button and a prompt appears. Sarah enters her username and password in the corresponding fields and then clicks LogIn. After her credential are verified she is taken to the Artist Page where she can change her settings. Sarah navigates down the page to the user account information section. She updates her email account by clicking on the dialog box that displays her old email and enters her new email. Then she clicks the save button on the bottom of the page and her account information is updated. When one of Sarah's prints is sold, she receives an **email notification**. The emails contain all the info she needs - the name and address of the customer, a thumbnail of the photo and the size of the print that was ordered.

2. Photo geolocation use case:

Sam Jenkins is a 40-year old landscape photographer and a CII **artist**. Sam goes to the CII website and logs in with his user-name and password in the correct fields. He clicks Log In and is logged in, he sees his own **gallery** and clicks the upload button at the top right of the page, which takes him to an upload screen. Sam clicks the button to select a file, browses to the correct file on his computer, then submits and uploads it.

He enters a price for print license by clicking on the text field area next to print price label and enters a correct dollar amount. Sam disables web and unlimited licenses by leaving those license amounts blank. Below the price fields Sam sees a drop down menu with **tags** to add to his artwork - Sam selects "Photography - Landscapes" by clicking on the drop down and clicking on Landscapes. After that Sam decides to enter a location of the photos that he just uploaded. Sam clicks on the text field labeled "Location" and enters Mongolia. The google map to the right of the text field zooms in on Mongolia. Sam is informed that customers will be able to

see this map when they view his image. After same verifies the rest of the info about the image, he logs out of his account by clicking the “Log Out” button at the top right of the website.

3. Customer Print Ordering use case:

Kim is looking for the perfect picture to hang in her hallway, since there’s an awful lot of naked wall there. While browsing through the **latest additions** to the CII site, Kim spots a somewhat familiar photograph of a waterwheel. Clicking on the image brings her to its **detail page**. There, she sees an enlarged view of the image, as well as a map with the **geolocation** of the image. Kim recognizes the **location** as one near her grandparents’ home.

She clicks the “Buy” button, and is presented with a few options. She selects the size of her print and clicks the button to add them to her cart. She sees in the top right corner of the website that the number next to the shopping cart image updates to “1”. Kim decides that she doesn't want to buy anything and clicks on the **shopping cart** button on the top right. She is taken to the **checkout** page where she can review her order. On the **checkout** page she enters her billing information and checks the box to ship to her billing address. She then clicks **submit** and is taken to a payment processing website.

After she finishes the **checkout** process for her own image, Kim goes back to the same image and selects a different size print to add to her cart. This second print will be for her grandparents, so when she gets to the checkout page, Kim enters her billing information and then her grandparents’ shipping information. She checks the box to indicate this item is a **gift**, and completes her order.

4. Administrator use case:

James is a site administrator. Over the weekend CII receives a DMCA copyright takedown request from a comic book publisher, listing the URL of an art piece it believes are infringing their copyright. James uses the website’s **admin panel** to hide those URLs from being accessed. His **admin panel** lets him search for any artist and view their profile in the same way that the artist views it, with all of the same permissions. He finds the first image that is allegedly violating copyrights and clicks on it to view its **details page**. On this page, a number of options are available, including the option to **suspend** the image, which keeps all of its data, but doesn’t let it appear on the website until the admin decides to un-suspend it. After suspending the rest of the offending images, James emails the artist again informing him of the temporary suspension of his images, hoping to resolve the issue.

5. Photo geolocation customer use case:

Audrey is browsing photos on the CII website and comes across one of an old farmhouse in some hills. She clicks the location button near the bottom of the image and a Google Map pops up with a pin dropped at the location the photo was taken. From there, Audrey is able to drag the map around to see nearby locations. It turns out that the farmhouse is in a town relatively close to where she lives. She clicks the directions button on the map and a new tab opens with the full Google Maps website, waiting for her to enter her start location to find driving directions to the farmhouse.

6. Subscription use case:

Marco browses the CII website fairly often to find new images to use in his work as a UX designer. Today he found a really great artist name Tom Jones, whose **gallery** contains lots of images that complement his design style well. Marco purchases digital licenses to a number of images for current projects, and wants to remember Tom Jones for future use. On Tom's gallery page, Marco clicks the button to subscribe to the artist. A pop-up window asks for Marco's email, which he enters.

A few minutes later, Marco receives an email with the details of his new **subscription**. It says he will receive an email on Monday if Tom Jones, or anyone else Marco is subscribed to, has uploaded new content in the last week. Now Marco will never lose track of his new favorite artist, and won't need to check the CII profile all the time to know about new images.

Glossary:

Account	A set of username and password stored in the database, allowing users to log in.
Admin Panel	The interface administrators use to access the website. Allows the admin to Reset Passwords, Edit Pieces, Hide Pieces, Delete Pieces, and Delete User Accounts
Artist	A user who provides products for sale on the website.
Categories	Groups of pieces that have something in common. Examples: “Art by Sarah Whitman”, “Surreal Landscapes”, “Animal Photography”
Customer	A user who is a potential purchaser of products from the website.
Email Notifications	Emails sent to artists when their art is sold. These let the artist know when they should expect payment, and if there is any action they need to take for a particular purchase, eg. Shipping a print.
Gift Buying	An option for customers to purchase products for another person.
Product	Media the artist is looking to promote on CII's site. May include image and video licenses and art prints.
Portfolio	An artist’s page displaying the products they have for sale on the website, along with information about the artist.
Purchase	A successful transaction in which a customer obtains a product in exchange for payment.
Prints	CII not only manages digital portfolios for their artists- they also give artists a simple platform from which to sell their own art prints or original, one of a kind pieces.
Sample	A freely downloadable copy of a product with a watermark to prevent unlicensed use. This is provided to potential customers for them to assess the usefulness of a particular product in their particular application before paying.
User	Artists and Customers.
Watermark	A semi-transparent logo overlayed on the product to prevent unlicensed use.
Geolocation	The latitude/longitude coordinates at which a photograph was taken. Can be displayed on a google map in the photo’s details page.

Subscription	A function allowing a customer to get email updates when artists they like add new art.
Gallery	The page that displays a certain artist's work.
Tag	A word describing the product that can be used to find similar products.

Functional specs:

<u>Priority</u>	<u>Number</u>	<u>Description</u>
1	1	The website shall have a home page
1	2	The website shall have a navigation bar
1	3	The website shall display the company logo
1	4	The website shall allow artist to upload the image
1		4.1 The website shall allow artist to add a description to an image
1		4.2 The website shall allow the artist to add a title to an image
1		4.3 The website shall allow the artist to set a price for an image
2		4.4 The website shall display artists contact information
2		4.5 The website shall allow artist to remove a previously uploaded image
1	5	The website shall allow artist to create an account
1	6	The website shall allow artist to login to the website
1	7	The website shall provide a filter function to filter image
1	8	The website shall allow customer to browse images in a gallery
1		8.1 The website shall customer to browse images by popularity
1		8.2 The website shall allow customer to browse images by artist
1		8.3 The website shall allow customer to browse images by category
3		8.4 The website shall allow customer to subscribe to an artist and receive notifications about new content
1	9	The website shall allow user to buy image from the website
1		9.1 The website shall allow user to download image after purchase
1	10	The website shall allow the artist to add a geolocation to any uploaded image
1	11	The website shall allow artist to set a price for each license of an image or video
1		11.1 The website shall allow artist to set a price for unlimited license
1		11.2 The website shall allow artist to set a price for print license
1		11.3 The website shall allow artist to set a price for web license
1	12	The website shall allow customer to purchase art with a variety of license options:
1		12.1 The website shall allow customer to purchase image with unlimited license
1		12.2 The website shall allow customer to purchase image with print license

- | | | |
|---|------|--|
| 1 | 12.3 | The website shall allow customer to purchase image with web license |
| 2 | 13 | The website shall allow administrator to manage accounts |
| 2 | 13.1 | The website shall allow administrator to remove/disable inappropriate content on artist accounts |
| 2 | 14 | The website shall allow customer to share an image by sending an automated email to a specified email address with a link to the image |
| 2 | 15 | The website shall notify the artist when their image/video was purchased via email |
| 2 | 16 | The website shall allow administrator to disable a customer account |
| 2 | 17 | The website shall allow administrator to disable an artist account |
| 2 | 18 | The website shall allow administrator to deactivate an image |
| 2 | 19 | The website shall allow administrator to organize the image for an artist per his/her request. |
| 2 | 20 | The website shall create a watermarked version of an image upon upload. |
| 3 | 21 | The website shall allow artist to upload a video |
| 3 | 21.1 | The website shall allow artist to add a description to a video |
| 3 | 21.2 | The website shall allow the artist to add a title to a video |
| 3 | 21.3 | The website shall allow artist to set a price for a video |
| 3 | 21.4 | The website shall allow artist to remove a previously uploaded video |

Non-Functional specs:

<u>Priority</u>	<u>Number</u>	<u>Description</u>
1	1	The website shall work on Google Chrome, Firefox, IE
1	2	The website shall use Bootstrap
1	3	The website shall use LAMP stack architecture
1	4	The website shall contain privacy policy
1	5	The website shall allow artists to upload images of up to 50mb
1	6	The website shall support the following browsers: Google Chrome version: 48.0.2564.116 m; and Mozilla Firefox version: 44.0.2.
1	7	The website shall implement LAMP stack architecture
1	8	The website shall use Bootstrap as the primary framework for front end development
1	9	The website shall use Google Fonts API for font styling
1	10	The website shall implement RESTful API
1	11	The website shall allow artist to add a location to an image to show where the image was taken
1	12	The website shall allow artist to advertise a print copy of an image
2	13	The website shall provide proper feedback to the artist if an image upload was successful/unsuccessful
2	14	The website shall provide proper feedback to the customer if a purchase was successful/unsuccessful
2	15	The website shall allow a customer to receive a free watermarked version of an image

Application Screenshots and Storyboard.

Pixit Site Layout

1. Home page
2. Search Results page
3. Image
4. Buy dialog
5. Log In page
6. Register page
7. Artist home/upload page
8. Image location

Each page has a navbar and footer. Navbars look like this:

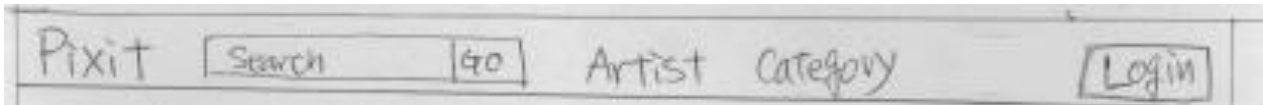


Fig. 0.1: The navigation bar

Clicking on Pixit takes user to home page. (1)

Using the search takes user to search results. (2)

The links between search bar and the login button allow users to jump between sections on the same page. They be different for each page.

The login button takes users to the login page (5)

If the user is logged in, the login button is replaced with a log out button.

This button logs user out of their account and returns them to home page (1)

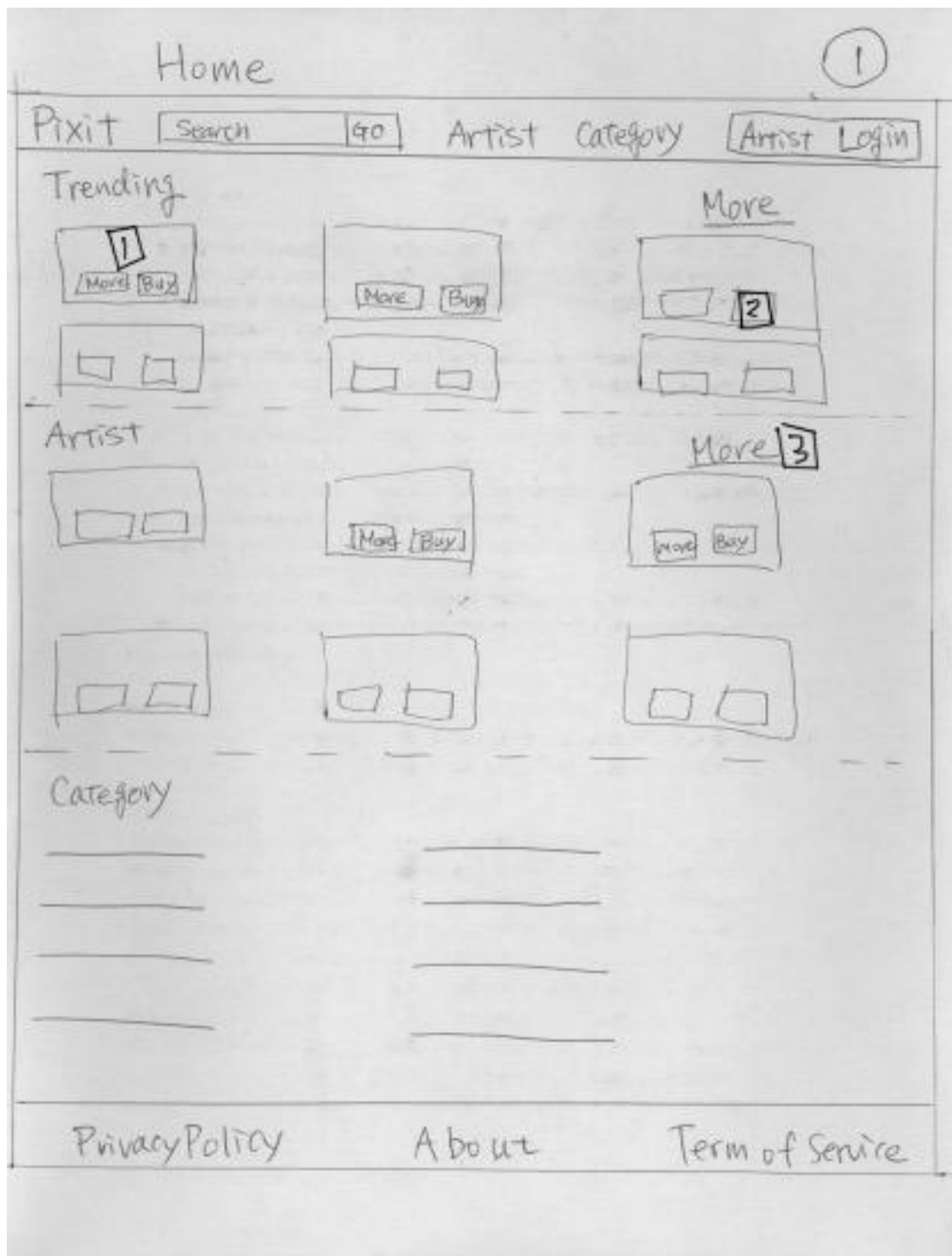


Figure 1: the home page

1. Clicking on the image opens a popup full-sized version of the image (3)
2. Clicking on "Buy" takes user to the buy page(4)
3. Clicking in "More..." takes user to artist Results page (2)

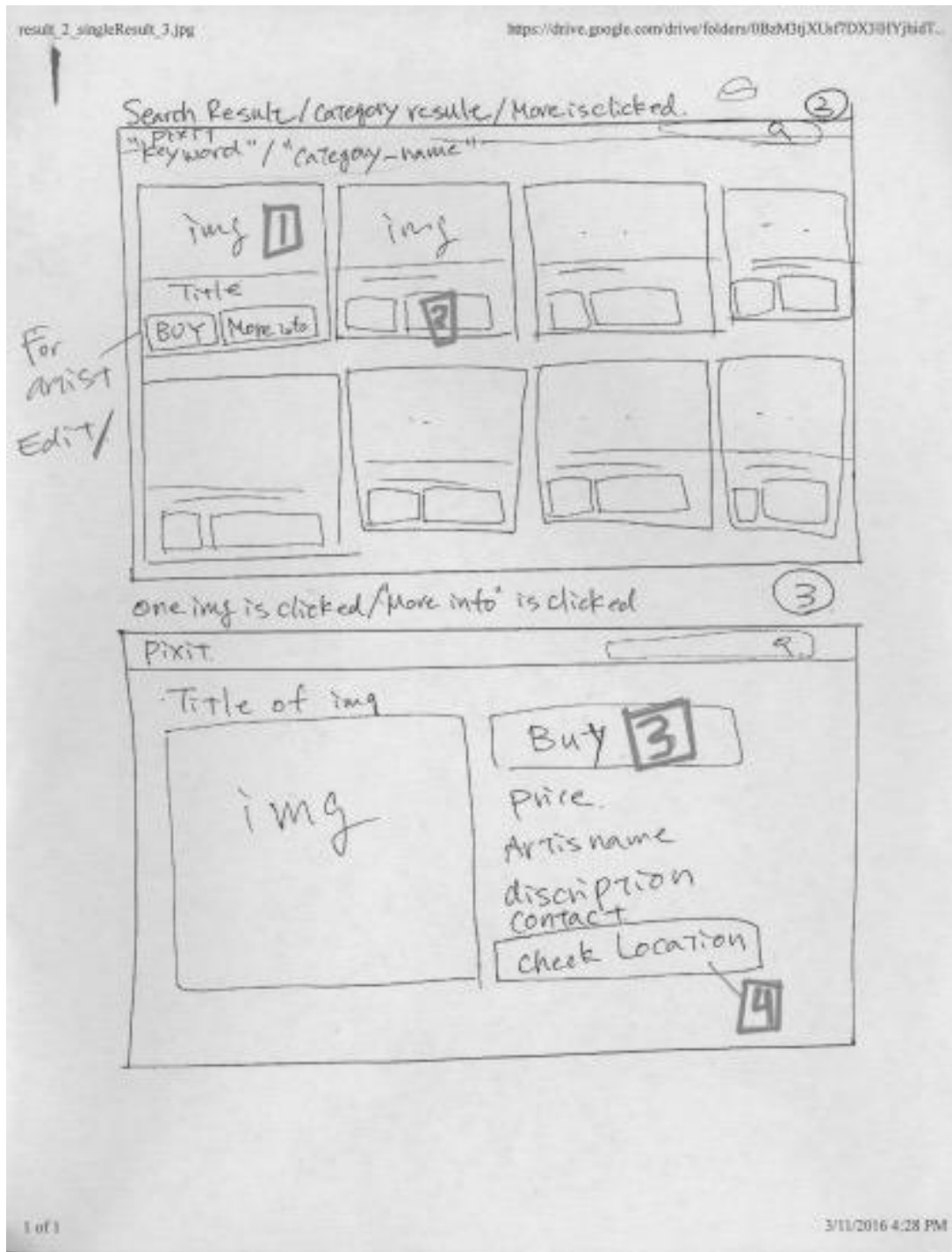


Figure 2, 3: Results Page and Image

1. Clicking on image opens popup fullsized image(3)
2. Clicking on "buy" button takes user to buy page(4)
3. Clicking "Buy" while viewing image also takes user to buy page(4)
4. Clicking on check location displays image location on map(8)

Buy (4)

Billing Address		payment info
Country		Name
Address		#
City		CVS
State	ZIP	Expiration

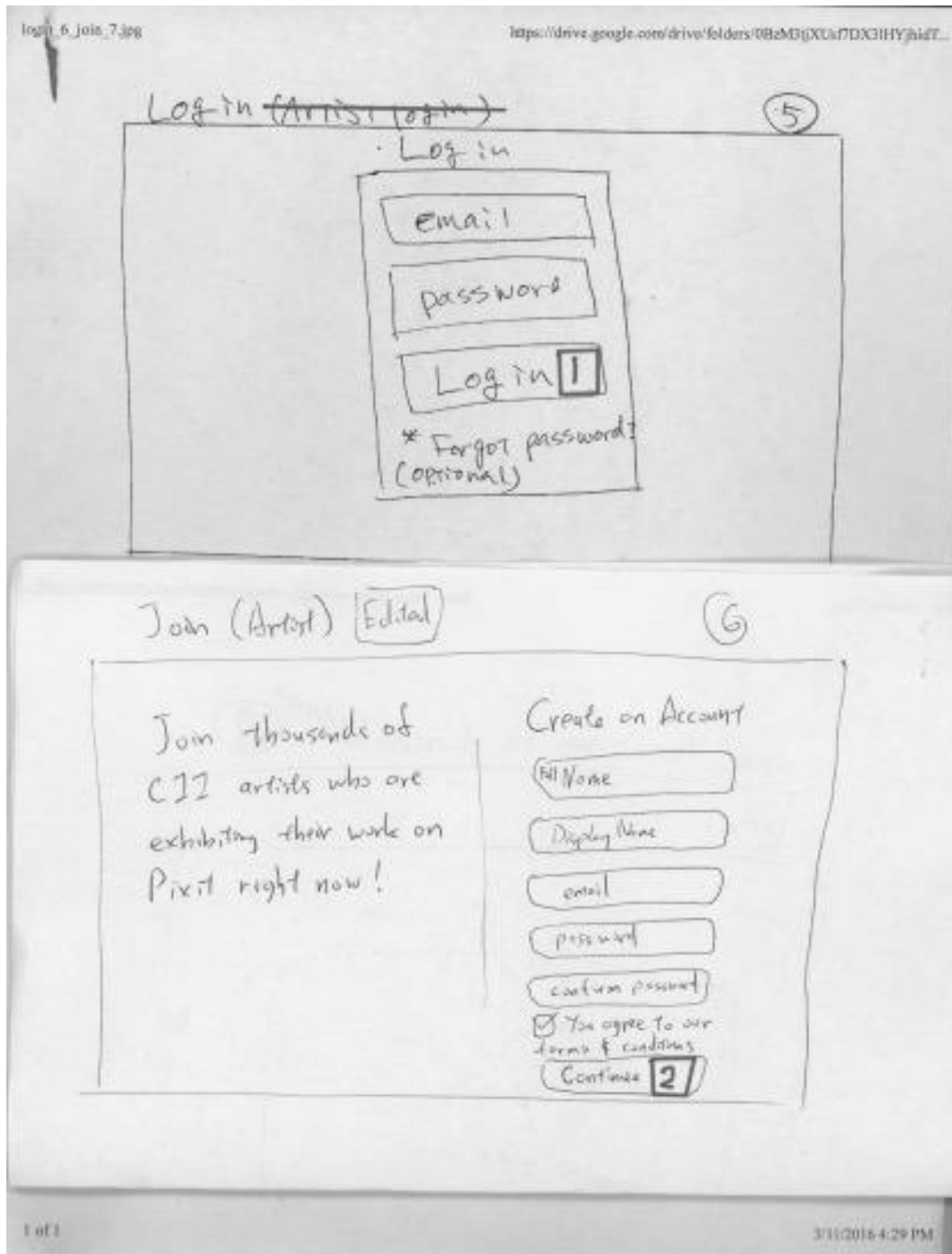
Total Amount

\$ 10,000.00

Purchase

Figure 4: Buy page

Clicking purchase shows the user a confirmation dialog. If the user confirms the purchase, they are directed back to the home page (1)



Figures 5,6: Login and Registration pages

1. Clicking on "Login" with correct email and password takes user to the home page.(1)
If the user is an artist, they will be taken to the artist home page instead (7)
If either email or password is incorrect, the user will stay on Login page and be notified of the error. ("Your email or password is incorrect. Please try again")
2. Clicking on "Continue" with valid information creates an account for the user. They will then be automatically logged in and sent to the home page.

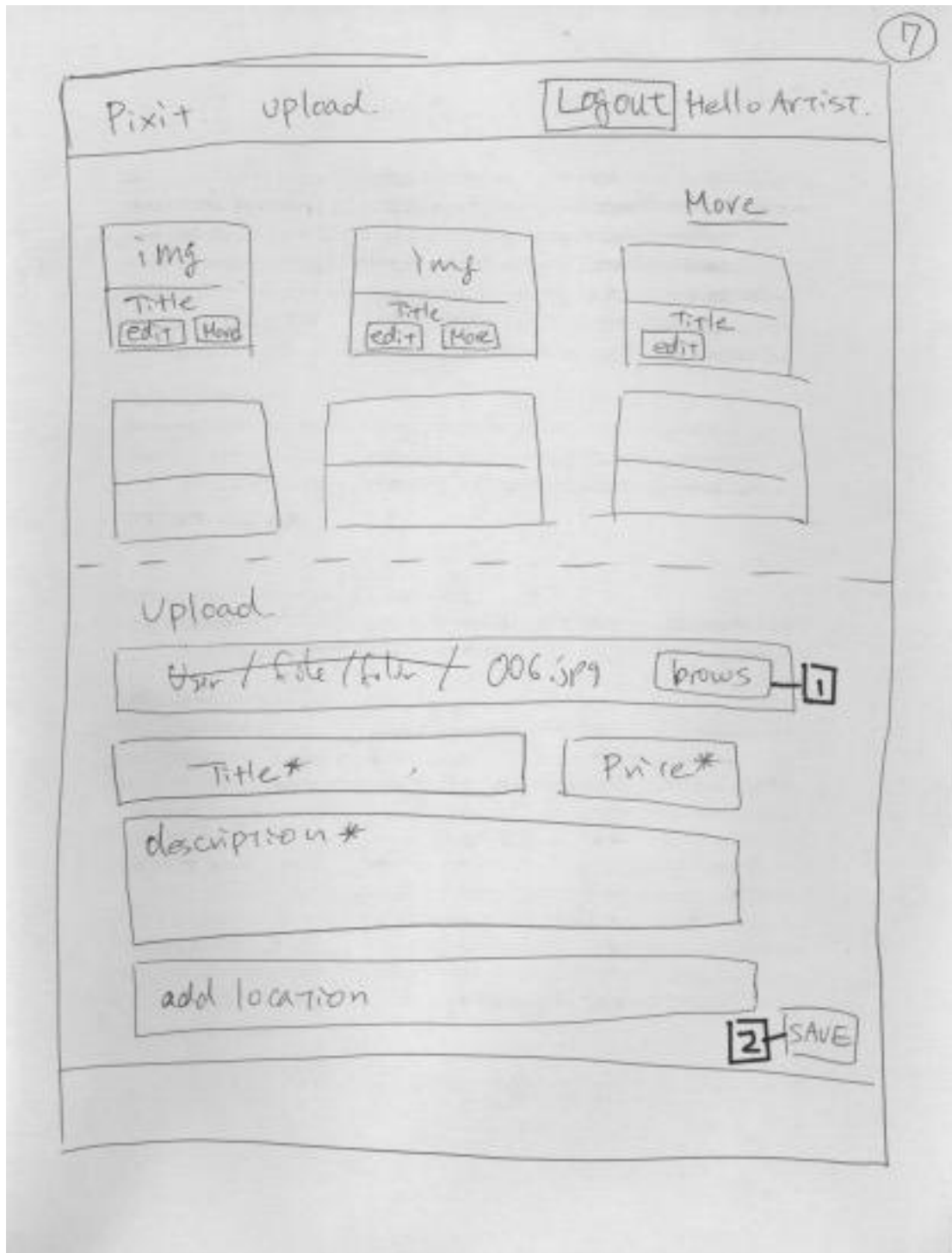


Figure 7: Artist Home Page

1. Clicking browse opens file dialogue for user to pick image to be uploaded.
2. Clicking save starts the upload process.

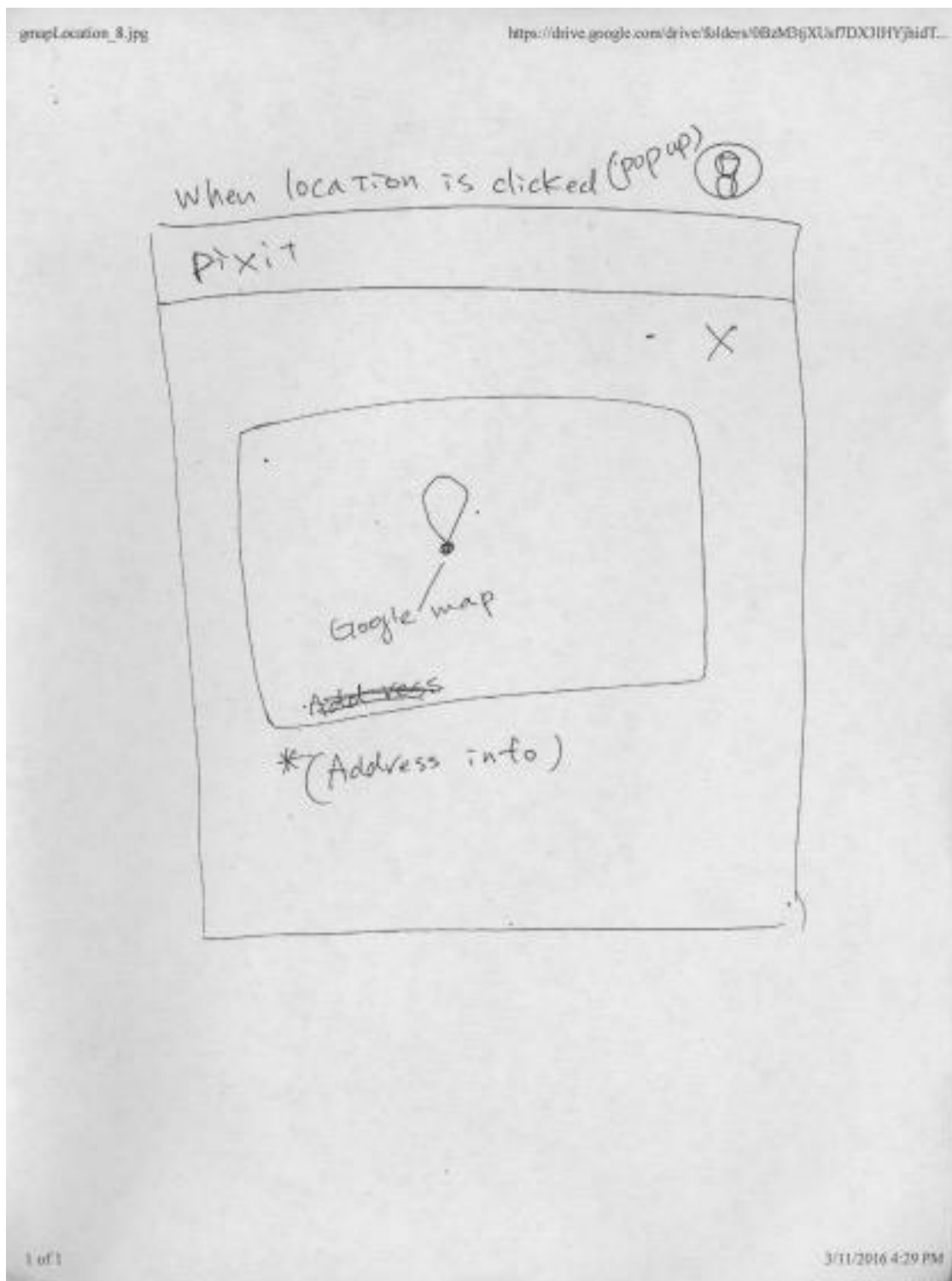


Figure 8. Image location

Clicking the X closes the map and allows the user to continue browsing whichever screen they opened the map on.

Competitive Analysis

Features	Pixit	Art.com	iStockPhoto	Shutterstock
Artist Subscription	+	-	-	+
Image Sharing	+	+	-	+
Location Map	+	-	-	-
Sell Print + Art	+	+	-	-
Watermarking	+	+	+	+
Gifting	+	-	-	-

There are 3 main competitors to our product, namely: “Art.com”, “iStockPhoto” and “ShutterStock”. These products have the same concept of an online media gallery where artists can upload, list and sell their art. While all of our competitors have similar business plan, it is the different features offered by each product that allows a client to choose one over the other. The features noted in the comparison table above are what would make our product stand out and shine. Our product allows users to subscribe to their favorite artists; keeping them updated on artists’ recent activity on the site, share images with their social networks. The product allows the artist to add a location to their media which would be displayed on a map, sell softcopies as well as hardcopies of art. Media is watermarked allowing users to preview the media before buying it. A gifting option is also available which would enable the users to buy media as a gift for their friends and family. Our product is the recommended choice for everyone since there is no similar product that offers all of these features on the Web platform under one umbrella.

High-level system architecture:

Our application will use several various tools in helping the team develop a more complete and effective product. The application will be implemented with PHP on two different web browsers, using LAMP stack configuration, and communicating with several third party APIs for improved workflow of the application.

Web Browser:

- Google Chrome: version 48.0.2564.116m
- Mozilla Firefox: version 44.0.2

jQuery plugins:

- Spotlight Effect,
- BlogSlideShow jQuery Plugin,
- Rotating Billboard System with jQuery,
- Text Over Image with jQuery.

Third Party API:

- Google Fonts
- RESTful API

Architecture Design.

For our application we are going to use MVC structure for the entire design to design and implement efficient and unique dynamic webpage that will allow Pixit to stand out from competition while keeping Artist concern a priority. With the MVC structure will allow our team to work on various aspects of the application without having any issues with code blocking and code merging and at the same time design a modern responsive and dynamic web application.

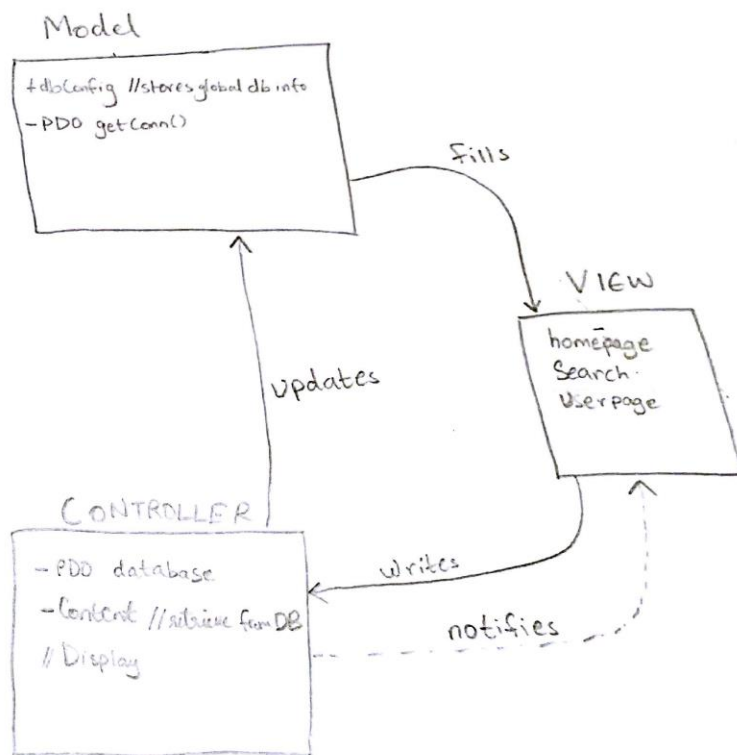


Figure 9 MVC structure

Content Management:

PixIt will allow the Artist to upload and manage image content on the website, the file format and metadata along with storage setup is described below:

GIF:

- Extension: .gif.
- Web Support: Chrome and Firefox.
- MetaData support: free-text comment field.
- May be replaced by PNG
- Interlacing and transparency support by most Web browsers

JPEG:

- Extension: .jpeg, .jpg, .jif, .jfif
- Web Support: Chrome and Firefox.
- MetaData Support: Free-text comment field
- JPEG widely supported by Web Browsers

PNG:

- Extension: .png
- Web Support: Firefox and Chrome.
- MetaData Support: Basic set of labeled tags plus user-defined tags.
- May replace GIF to improve quality.

The content will be stored in file systems referenced to in the database to provide the artist and any customer with the ability to search and view any images defined by specific search parameters. The artist will be able to upload the image through personal account using the following MetaData.

MetaData:

- Title of the image.
- Description of the Image.
- Any additional tag to describe the image and to be used for search.
- The individual pricing for each of the licenses.
- Provide geographical location of the image (Optional)

Search Algorithm and Prioritization:

The application will implement 2 types of searches, generally referred to as simple and advanced searches. The basic search will allow the customer or artist to type anything in the search box and the application will refer to the Content Data Table in the Database and match the search string to the Image with similar string in the Description column or Tag column.

The Advance search will allow the Customer or Artist to select a specific value in the drop down menu and search based on Categories which will be prefilled values. Based on the Artist name which will also be prefilled values. With the implementation of auto complete, the Customer or Artist will have a limited number of options as the string expands.

Prioritization algorithm is implemented to setup Rating and Popularity of images on the website. With each click, a particular image will get an increment of the counter used. When the controller references the database, the counter is one of the values to be returned and that will be used to arrange the view of the main page, results page and Artist page.

Additional Tools and Software:

For this application we will also be using NetBeans IDE for development of our application and our application will run on virtual Ubuntu 14.01 LTS server running in the AWS (Amazon Web Services) cloud. Our main working code will be stored remotely on SVN so that the rest of the team will be able to have access to the code while keeping operational. Our content will be stored on MySQL database to be manage and be uploaded to the web application.

LAMP Stack

PixIt will utilize LAMP to enhance the dynamic webpage. PHP is will be implement MVC control structure, to establish the entire back end communication with MySql database to upload and download images for both the Artist and the Customer. The breakdown of the framework is as follows:

Linux will:

- Compile the dynamic webpage (HTML, CSS, jQuery and PHP).

- Present the Artist and user opportunity to interact with the application.
- Handle multiple requests between server and database.

Apache web server:

- Will host and upkeep connection to the application.
- Maintain codebase (HTML, CSS, jQuery and PHP.)
- Support the MVC framework of back end design.

MySQL:

- Will hold tables that will hold content information.
- Will respond to requests from the server.
- Provide output based on the query request from MVC

PHP:

- Will implement MVC framework.
- Setup up the dynamic webpage.
- Handle from Artists and Customers.

Class and Data Definitions.

There are three main classes in our project; Admin, Artist and Content. The Admin class is the administrator of the website who can edit and moderate the media on the website. The Artist class consists of clients of the company who upload their media to our website for display and sales. The Content class consists of the media that is uploaded by our artists on the website.

UML and UML component Diagram.

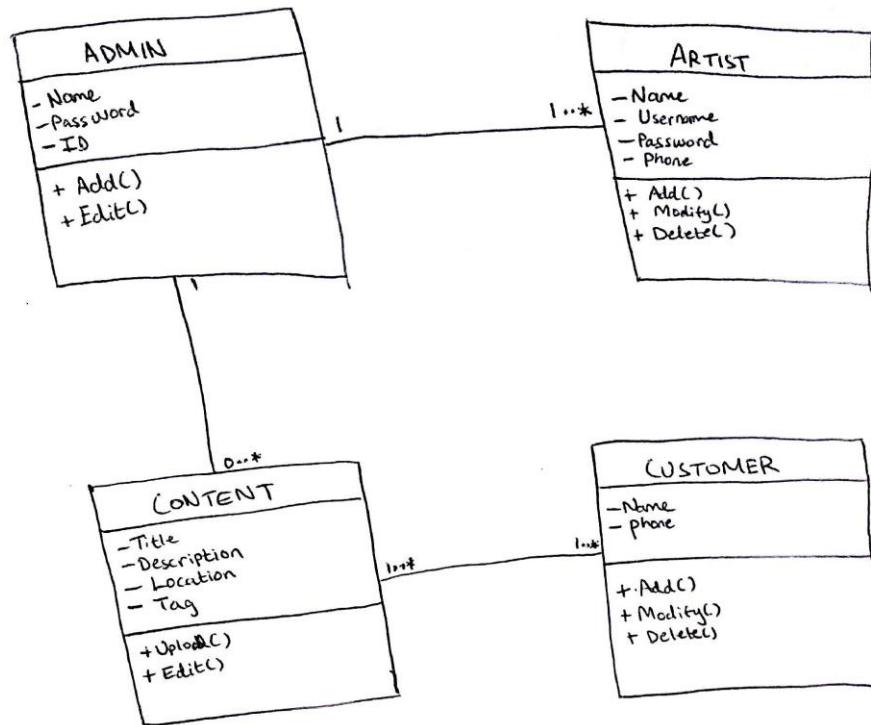


Figure 10 UML class diagram

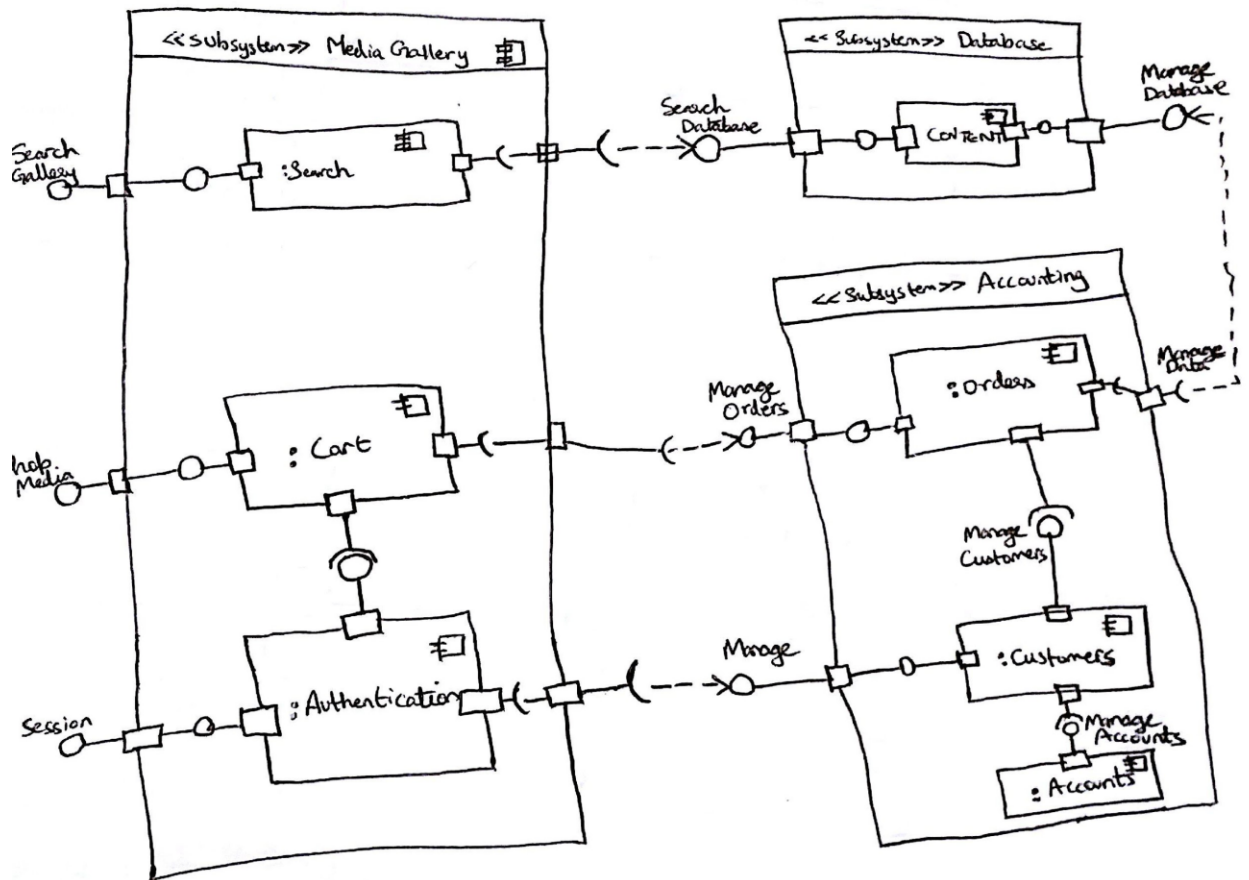


Figure 11 UML component Diagram

ER diagram and DB structure

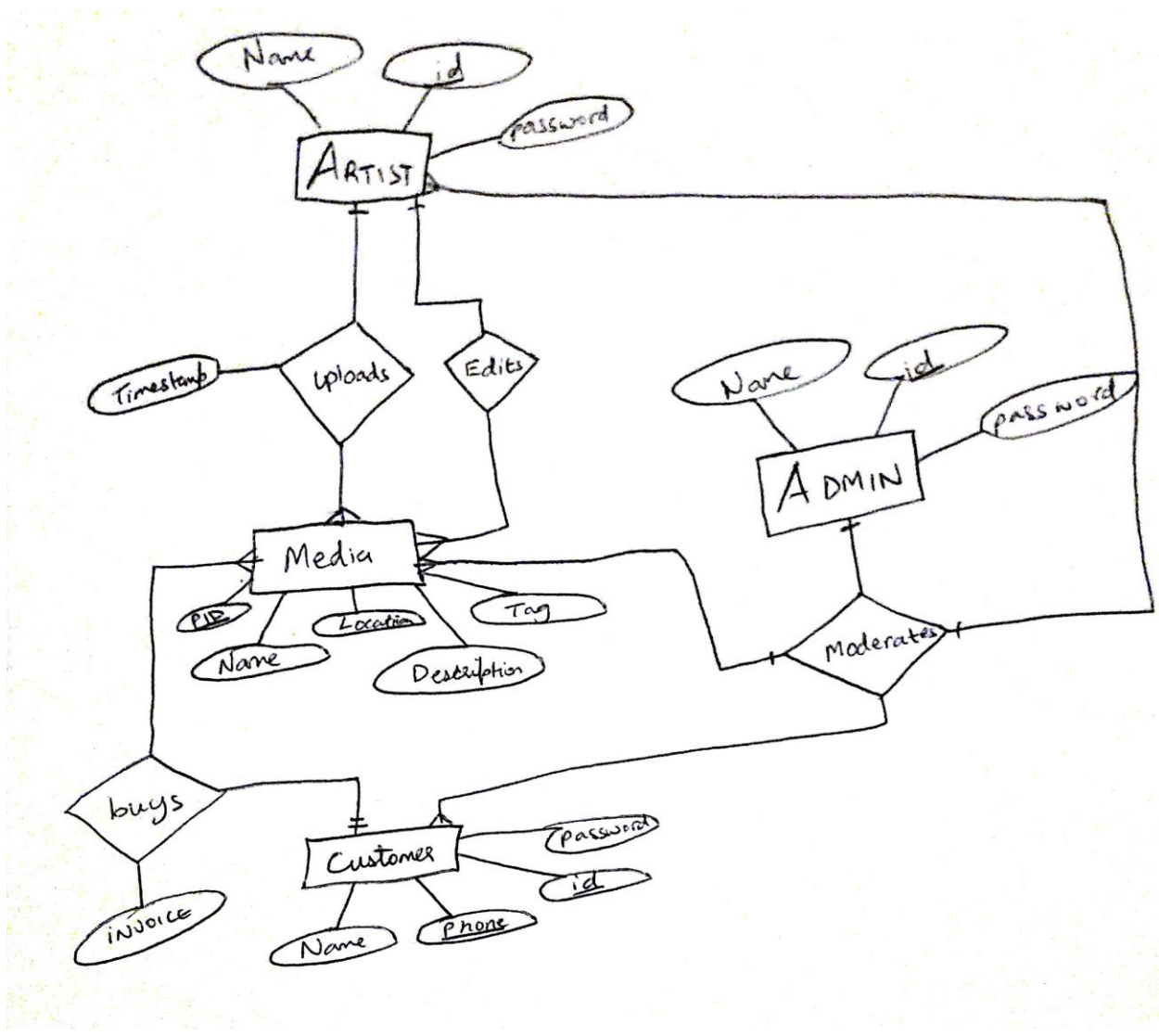


Figure 12 ER diagram

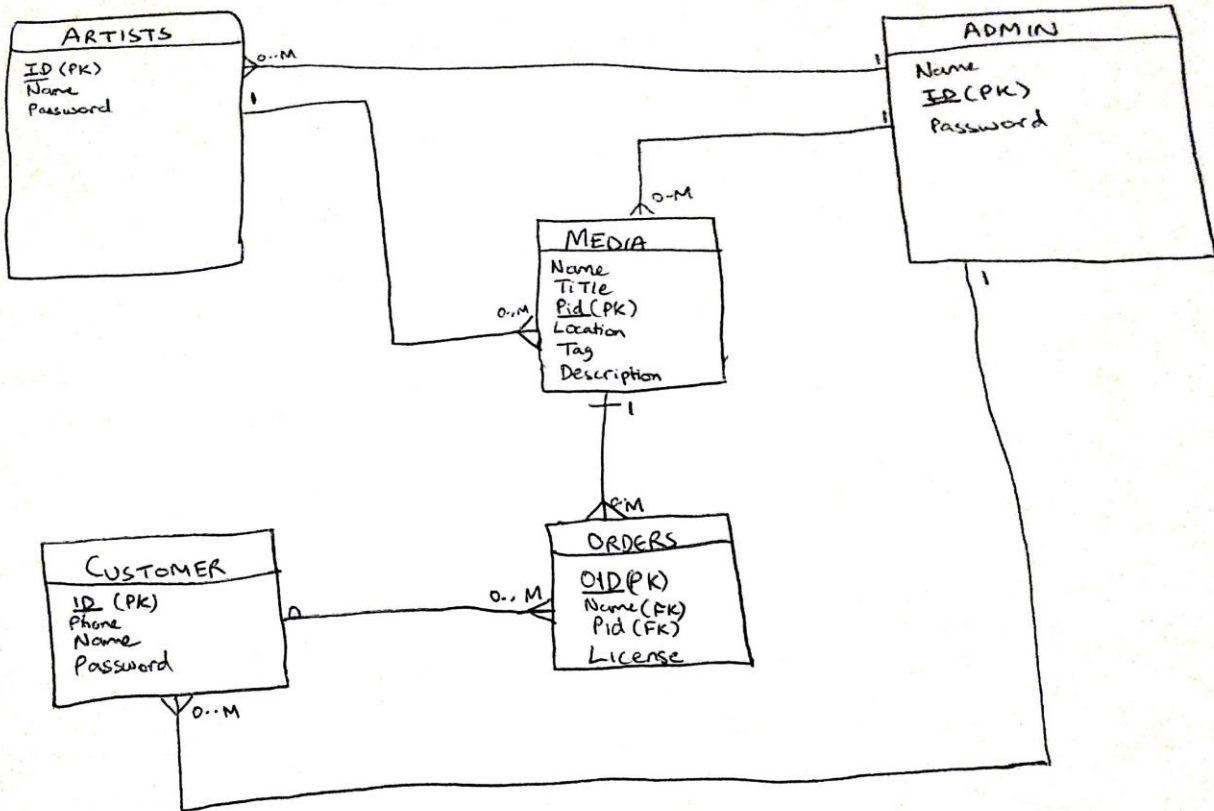


Figure 13 ER Schema diagram

Data is stored in a sql database. The database consists of three tables, namely Admin, Artist and content. The content table is our storage of media uploaded by artists and hosted on our website. The artist uploads the media and the administrator moderates the media uploaded by artists.

Data content and basic Data types

Images that are uploaded by the artists will be stored in our database in a file format. Each uploaded image will have a timestamp that marks its entry into our database as well as string fields for title, description and location of the image. The data will be provided by the artist at the time of upload. This data will used to display the hosted images and their information to customers on our website.

The names, ids and passwords of artists that host their media on our website will be stored in our database in string format. This data will be provided by the artist at the time of registration and shall be used for login authentication.

The name, id and password of the administrator will also be stored in our database which shall be used to authenticate and grant access to the administrator account. This data will be manually inserted into the database by us.

Key Risks

Skill

We are all fairly new to web development. Most of the team members have not had web development experience past CSC 412. Because of this we will not be able to implement complex web desing models.

Our solution to this is to minimize the workload and work with simple design structures. We are all doing our best to learn php to implement the MVC model.

Scheduling

Due to team members' schedules is it difficult for everyone to meet face to face regularly. We have a regular meeting set up Wednesdays 5:00PM - 7:00. We usually have one team member Skype in.

To work around scheduling we constantly keep in touch and update each on our weekly schedule and change the meeting time if needed.

Technical

This risk is connected to the skill risk. Because it is the first time any of us are designing a full stack website, we are still learning all of the technology that goes into web design.

Legal/Content

Since we are dealing with digital art, there is always a risk of copyright infringement. We require a large collection of images for testing and we need to make sure they are legal to use.

We will make sure to use only "free to use and manipulate" content that we obtain with advanced google image search.

Team Configuration:

Name	Roles
Anthony Wong, Nan Den	UX/ UI
Andrey Barsukov (lead)/Matvey Voishchev	Back end design and Database Schema design
Ammar Naqvi	Database schema implementation
Ammar Naqvi/Andrey Barsukov (lead)	Backend/External APIs
Matvey Voishchev/Evan Edge	Content Management/Documentation/SQA Testing