Software Engineering CSC648/848 Spring 2016

Kwik-Pix application

Milestone 2

3/18/2016

Team 5 (Local)

Dana Muise, Team Lead dmuise@gmail.com**, SFSU**

Adolfo Von Zastrow adolfovon@gmail.com **SFSU**

Jay Naidu jnaidu@mail.sfsu.edu **SFSU**

Toru Nagoo inmylife777@gmail.com **SFSU**

Nikolay Paulov socomdragon2003@yahoo.com **SFSU**

Hargit Randhawa harjitr@mail.sfsu.edu **SFSU**

revision date	Revision description
3/13/16	Use cases revised with more detail Wireframe drawings revised.
3/13/16	Functional spec revised, expanded and prioritized Glossary expanded, Risk assessments added
3/16/16	Added functional specs Revised executive summary Added technical responsibilities to team description
3/18/16	.Revised Risk assessments Added high-level class definitions an UML charts Added Database entity relationship model Finalized for initial review
3/21/16	Revised Class UML, Database ER model
4/1/16	Revised Wireframe storyboards, Document frozen

Executive Summary

The Kwik-Pix application is a product designed to streamline and complement the existing digital photograph model for Collective Images Inc. It provides a simple online solution for product management, artist media submissions and improves overall client purchasing experience all with a minimal overhead to maintain.

Today's media distribution market is saturated with online presence. Most websites compete by presenting their customers with excessive options, endless controls and complicated interfaces that busy people don't have the time to figure out. The artists whom the media companies depend on for content are almost completely ignored when it comes to designing online experience. Website media administrators are expected to work with unfriendly user interfaces and cumbersome databases.

The difference between Kwik-Pix and the current media websites is in it's simplicity, We streamline the user-experience for everyone, especially the customer.

- Customers can search the media directories with a simple interface. Just enter a keyword and the media type and hit search. They don't need to be a registered customer to browse the vast photo library.
- Artists can upload their files and descriptions to the administrator with one button. They can upload one size and not worry about creating different sizes, they will be generated automatically. They can easily view all thier uploads and license transactions on one page
- Administrators can manage client and artist data in a clean, easy to use interface. Data is organized in a straightforward yet sensible way. Customers and artists are presented on one page, just click on a member to view all thier details.

Use Cases

Kathy: 36, Artist (returning registered artist).

Kathy, a professional photographer is not very computer savvy, but knows internet basics. She has had success with Collective Images Inc in their traditional print service, but she has registered for their new online media distribution and will upload her latest photo for distribution. Kathy logs into the main page and is automatically forwarded to her account page,where she can see all her latest transactions. She clicks on the ***upload new media button*** taking her to the ***Upload* screen**. She enters the photo title, description, three search keywords, and a few other bits of information, then clicks on ***Choose File***. On her hard dive, she selects her JPG file and hits ***enter***. After a few seconds, the photo appears on the ***Upload review* screen** with the associated information, much in the way it will appear on the web site. Pleased with the final layout, she clicks the ***Submit*** button, taking her to a screen that says *We will review your submission and get back to you shortly*, *thank you*. *Submission #39384*. She clicks on the ***home* button** to return to the website home page. A few days later the website administrator informs her by email that her submission #39384 was approved and now resides on the Kwik-Pix website.

Jason: 22, Student, returning customer.

Jason is a competent user when it comes to computers. He needs an image for a school presentation so he turns to Kwik-Pix. He is already a registered user and after he logs in, he is presented with the Kwik-Pix seach page. Already knowing what subject he is looking for, he enters a few related keywords and clicks on the ***Search* button**. A few seconds later he sees the **search results**: A gallery of thumbnail size photo's matching his subject in. He scrolls through the images, finds one he likes and double clicks on it. He is presented with a new page displaying the image in three sizes: small medium & large. Also shown are the license fees, description, file sizes, artist name and information. After consideration, he decides to license a medium size copy of the photo by selecting the options and pressing ***Add To Cart*** button. He needs another photo so he starts the search process again by pressing ***return to search***. He would like to finalize his purchases so he clicks on the ***checkout*** button, which brings him to his ***checkout page***. Here he can review a summary of all his purchases. Satisfied, he press submit and is presented with a ***transaction complete*** page stating *Checkout Complete. Thank you for your purchase, your Kwik-Pix images are on the way!* I He is done for now and closes his web browser.

June: 31, Conference Speaker, new customer.

June is a busy professional and needs a photo of a cityscape for her presentation. She hears about Kwik-Pix and goes to the web site. She is greeted with the Kwik-Pix landing page. Right now she's not a member but that's okay, she can still browse through hundreds of photos. She notices you can search for particular subjects so she types in "Cityscape" in the ***search bar*** and is shown 24 great photos. She adds two of them to her cart and clicks on ***BUY***. At this point Kwik-Pix asks her to please register with a username and password. After she complies she is shown a ***checkout*** page to make a final review of her choices and understands the prices. She selects ***SUBMIT*** to complete the purchase order. A Kwik-pix representative will contact her shortly.

David: 42, IT Administrator

David is an experienced computer professional with a Masters in Computer Science and has worked as a web site administrator for years. David is an employee of Kwik-Pix and is in charge of managing the site and interacting with the artists registered on the site via email. Today David received notification that there are several pending photos that have been uploaded by an artist that need to be approved. He logs on and selects the ***Purchase Orders*** page. He can see a thumbnail view of three photo submissions, along with all its associated information. The first two look fine and he clicks the ***approved*** button. An automated message is sent to the artist mentioning the approval and the website automatically adds the photo to the database, ready to be purchased, The third photo contains objectionable content. The administrator types in *This photo violates our submission guidelines* and presses ***Reject***. An automated message is sent to the artist mentioning the rejection.

Glossary of terms

Account:	A collection of data about a particular user of the application
Artist Account Page:	A page on the site that displays all Artist pending or past submissions,. or purchase orders. They may also edit personal information, such as passwords.
Administrator:	An authorized employee that has access to all controls of the site
Artist:	Creators of media that is to be distributed on the site
Client side:	Browser side view of the site, customer/ artist's viewpoint
Customer:	Members of the general public who purchase media
Browse:	Browsing through media, but not necessarily purchasing
Download:	Retrieving a copy of data from a remote server to a hard drive.
Keyword:	Metadata used to connect a file with a database reference
Landing Page	The Home page or the first page of the website the general public will see After typing in www.kick-pix.com
Media:	Photography, illustration or fine art stored in digital format
Member	Anyone who has signed up with a username and password, and agrees to adhere to the Kwik-Pix Terms of Service Agreement
Member Account page	A page on the site that displays all member pending or past purchase orders. They may also edit personal information, such as passwords.
Site:	The Kwik-Pix project in it's entirety, included data bases, all applications and code
Search:	Submission of database query using keywords to locate media
Server side:	Secured database and application files, controlled by administrator
Purchase order:	The submission of selected media customer is interested in
Transaction:	The completion of media sale to the customer
Terms of Service	An agreement by registered members of Kwik-Pix to adhere to specific policies regarding transactions, and communications between other customers.
Upload:	Media or data transferred from the artist to the administrator for review

Functional Specs

M3 Priorities for friday

Full browsable hardcoded display of 3 X 3 latest photos on the landing page, with the existing search field.

Individual photo information page, when ever any photo is visable

Login/ password functionality (customer) i master Logon script

Customer account page

New user page

Nav bar

Priority 1

1) The **general public** shall be able to search the directory of photos without having to **login** in as a member. Search media by title, keyword or artist.
2) The **general public** shall be able to **browse** through the photos of a search result, without having to login as a member.
3) The **general public** shall be able to click on an individual photo to get more information about it, without logging in as a member.
4) The **general public** shall be able to collect photos they are interested in to their **Shopping Cart** without logging in as a member

5) The **general public** shall be able to login, or become a new member via the website login page, after approving the **Terms of Service agreement**.
6) **Members** shall be able to purchase a **usage licence** for photos in the **shopping cart**, once **logged in**.
7) **Members** shall have access to their own **Member Account Page**
8) **Artists** shall be able to upload their own photos for submission to the site
9) **Artists** shall have access to their own **Artists Account** Page
10) **Artists** shall be allowed to purchase licence agreements.
11) **Admin** shall be able to approve all uploads before seen on the website.
12) **Admin** shall be able to interact with the **database** safely
13) **Admin** shall be able to add submitted photos to the site
14) **Admin** shall be able to delete media.
15) **Admin** shall be able to edit or delete member information
16) **Admin** shall have access to a complete list of members and artists
17) **Admin** shall be able to receive a client's **purchase orders**
18) **Admin** shall be able to communicate with artists & members via email
19) **The website** shall have a **Terms of Service** page
20) The website shall offer up to three sizes of any photo offered for license
21) The website shall offer up to three usage licences for each photo.

Priority 2

22) **General public** shall be able to sort the browsable photos on the landing page by a variety of filters, ie size, artist, subject, color....
23) **Artists** shall be able to delete their own **artwork** from the website
24) **Artists** shall be able to change art work information
25) **The website** shall have a help & support page
26) **The website** shall spotlight popular artists

Priority 3

27) The **landing page** shall display the newest or most popular photos.

28)**The Website** shall be copyright protected with watermark technology.

29) **Members** shall be able to sign up for direct email marketing

30) The website shall offer a discount on large quantities of licences.

Non-functional Specs

1) The website shall load search responses within 1 second
2) The website shall have an expected load of < 5 seconds at 95% confidence level.
3) The website shall have a minimum 64 bit encryption security
4) The website shall permit a maximum of 50MB per image
5) The website shall be available 99.9% of the time
6) The website shall have a fault tolerance availability of 99.9% with no more than a 5 minute average downtime
7) The website shall be built with resilient fault tolerance as per SFSUSWE.EDU performance criteria.
8) The website database shall initially live sfsuswe.edu server
9) The project shall be developed using Subversion version control software

Competitive Analysis

	Kwik-Pix	everystockphoto	Flickr.com	shutterfly	getty images
search	+	+	+	+	+
browse	+	+	+	+	+
upload	+	-	+	+	+
download	+		-	+	+
multi-size	+	+	+	+	-
fine art	+		-	-	-
simple interface	+				

The competitive relationship of our planned product **Kwik-Pix** is a streamlined, stock photo browser that is easy for customers to browse and search for photos and artwork. It also takes artists into consideration by creating a simplified upload system, leaning more on graphic controls rather than complicated forms and options. The administrator control panel is simple, clean and straightforward. Overall, our focus is on stock images - made simple, no more excessive options to get in the way of the user's viewing experience. As can been seen on the chart, our competition pays no attention to simplifying the design of their interface. This will be Kwik-Pix's competitive advantage.

High-level system architecture

The product is a web application, and therefore the user access it through a web browser. The supported browsers are:

- Google Chrome 48.0.2564, on Mac OSX and Windows.
- Mozilla Firefox Firefox 44.0.2, on Mac OSX.
- Mozilla Firefox Firefox 38.6.1esr, on Windows.

The application is organized following the Model-View-Controller (MVC) architecture pattern. It runs on a LAMP stack, with the following components:

- The front end, written in HTML, Javascript and CSS:
    - Our application Views consist of plain html/css files.
    - Javascript scripts communicate with the server and populate data into the views.
- The backend, written in PHP:
    - The Controllers consist of PHP classes, and handle request for specific types of resources, by interacting with the respective Model.
    - The Models consist of PHP classes that communicate with the Database, and pass data back to controllers.
- And a MySQL database:
    - The database holds data about the different types of users (artists, clients and admins), media metadata, and purchase orders.

Tools and platforms used:

- JQuery:
    - Provides a simplified way to perform AJAX requests.
    - Allows for easy and intuitive DOM manipulation.
- JQuery UI:
    - Premade UI components and controls.
- CSS Bootstrap:
    - Provides well structured and consistent styles.

These tools are cross-browser compatible, and will provide consistency to our application.

Media storage:

All media will be stored as files in the server's file system. This decision was based on the following reasons:

- Simpler implementation.
- Media data will be kept in its natural format.
- Easier to load and access media data, without database overhead.

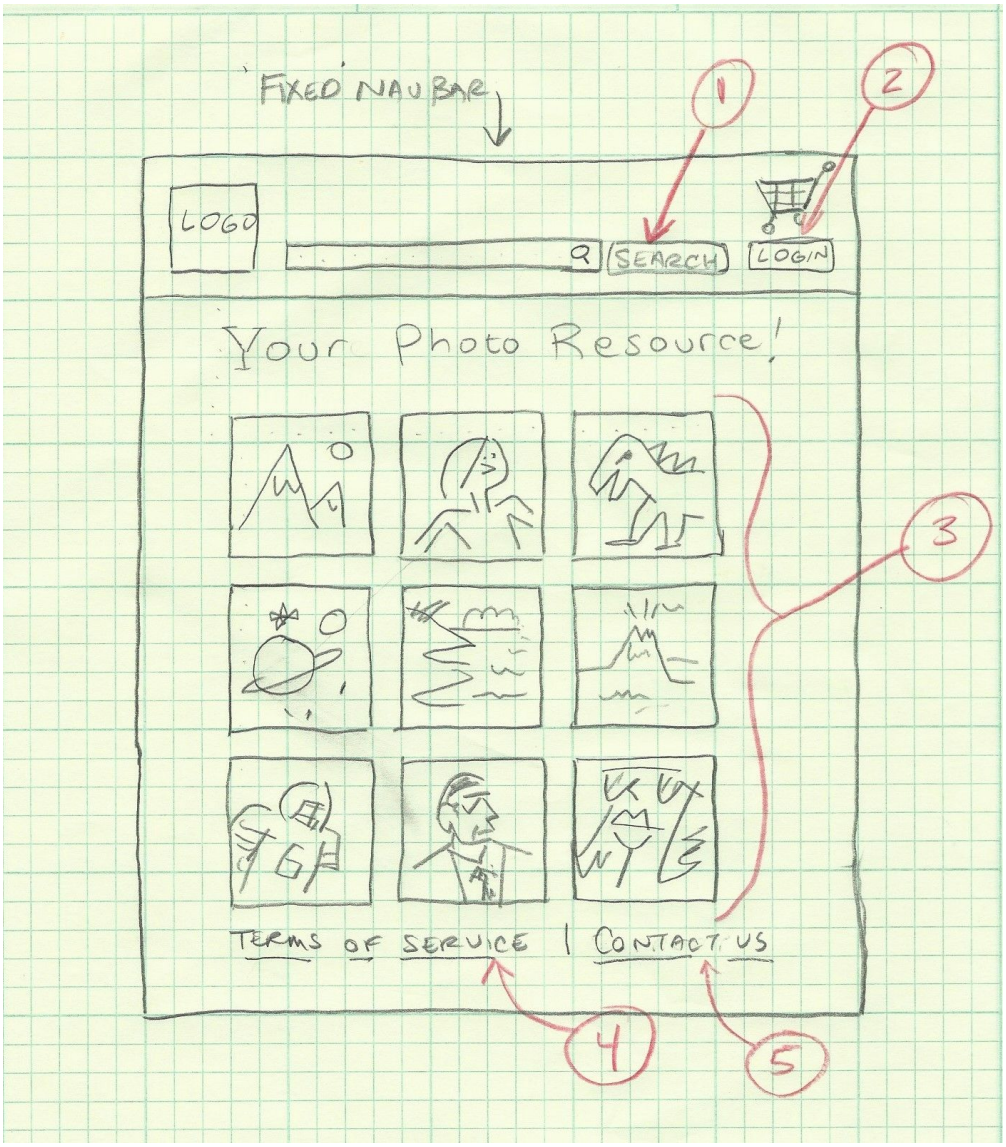
Search architecture:

Our search functionality consist of the following components:

1. Search user interface: HTML form with a single text input and a button.
2. Search script: Simple script that sends request to the server, formats the response and adds it to the user interface.
3. Pictures Controller (PHP class): This class takes the 'keywords' param sent by the search script, splits them by empty spaces, and queries the database. The database query utilizes MySQL's REGEXP, and matches the keywords against the title and description fields in the media table. The query returns any records with the keywords present in either or both of the title and description fields.
4. The search functionality returns media even if the keyword is part of another word, e.g: searching for 'cat' will return media with the word 'cats'.

High Level Storyboards

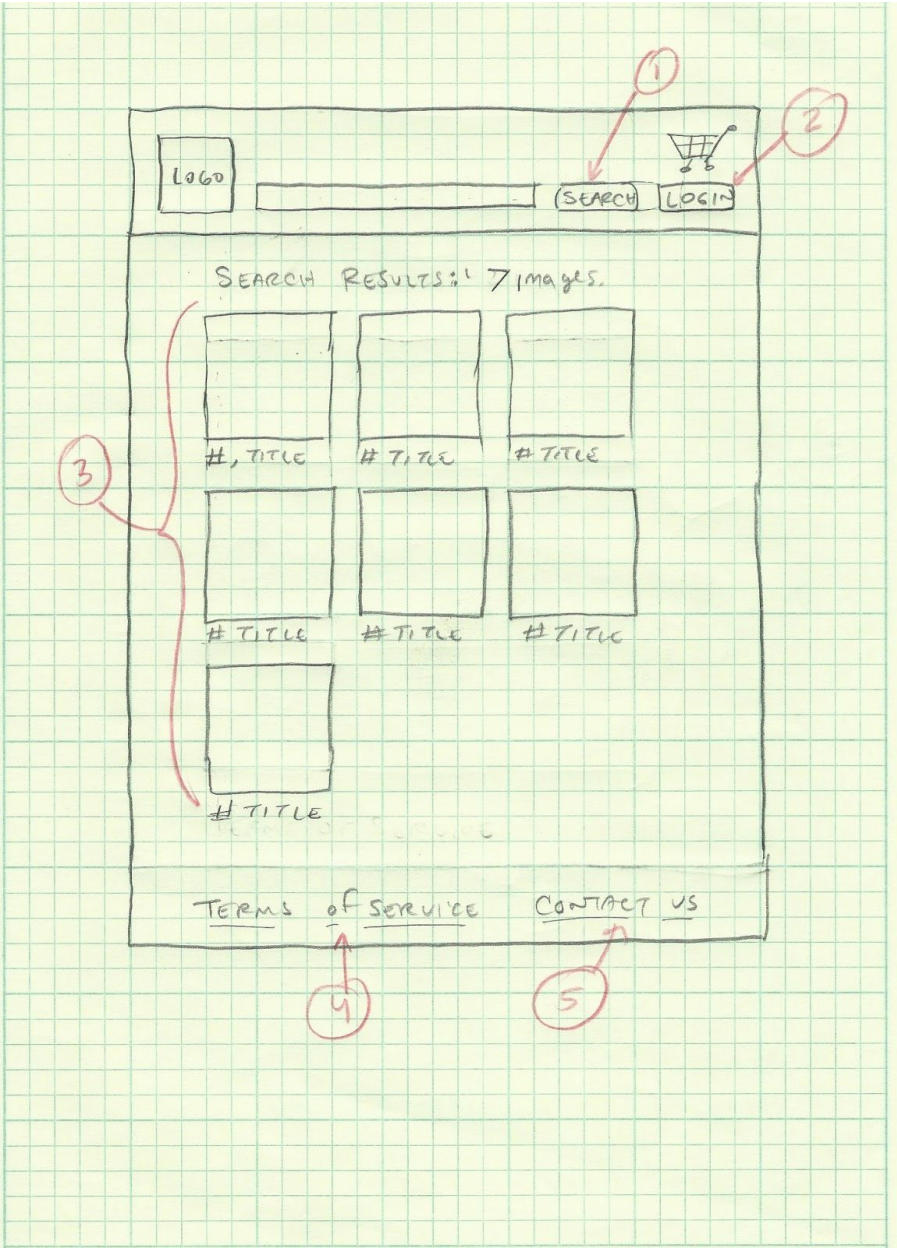
Landing Page



The **Landing page** is the first page the user will see when visiting the website. It will display the latest 9 photos. There is a *fixed* navigation bar, that is present on every page on the website.

1. Search bar: executes search on keywords, then goes to the Search Results.
2. Login: Goes to the login/new user page
3. Clicking on any photo takes you to the Individual Photo Information Page.
4. This hyperlink takes you to the terms of service page
5. This hyperlink takes you to the Contact Us Page

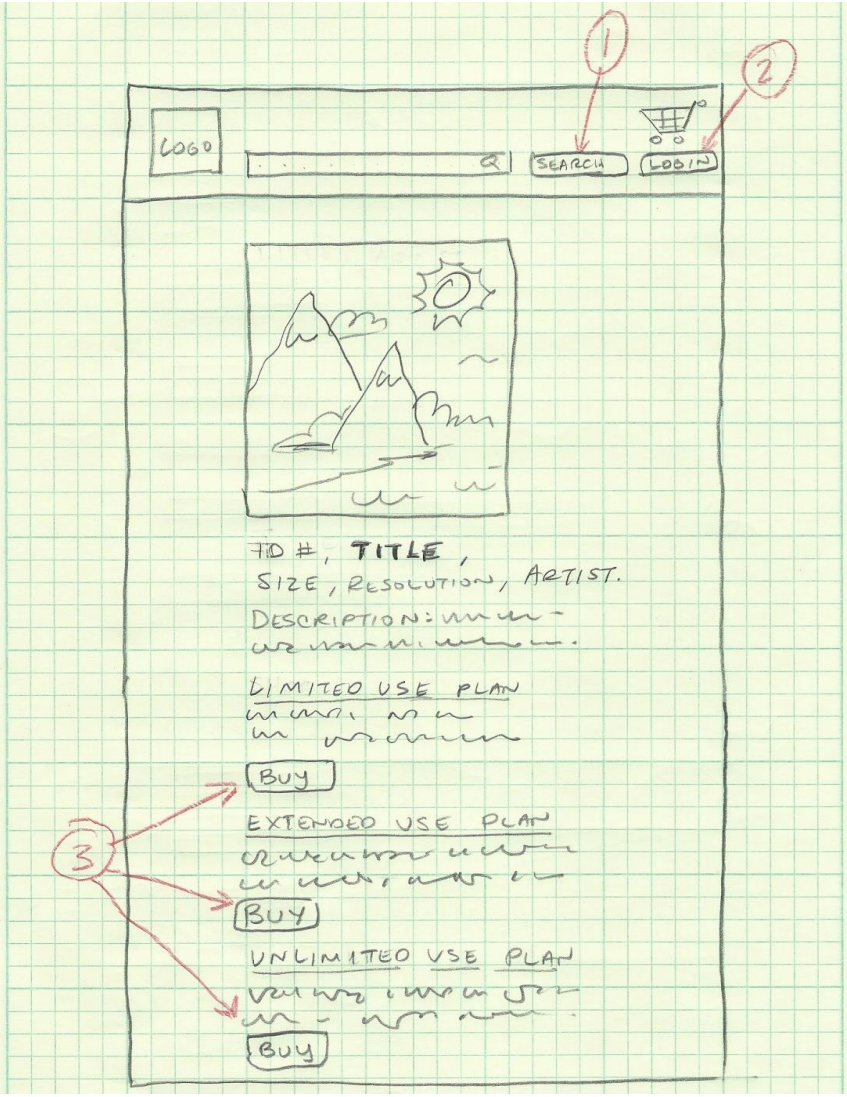
Search Results Page



The **Search Results page** displays thumbnails, ID number and titles of search results. There is a *fixed* navigation bar, that is present on every page on the website.

1. Search bar: executes search on keywords, then goes to the Search Results.
2. Login: Goes to the login/new user page
3. Clicking on any photo takes you to the Individual Photo Information Page.
4. This hyperlink takes you to the terms of service page
5. This hyperlink takes you to the Contact Us Page

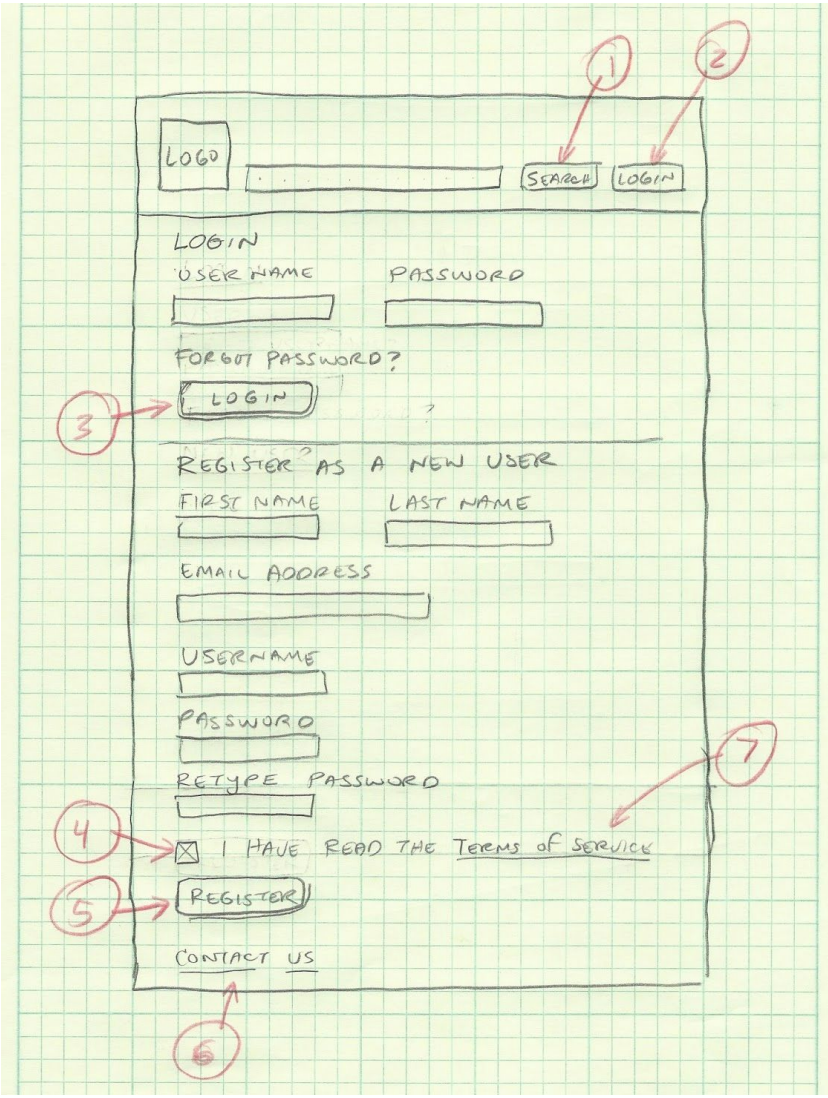
Individual Photo Information Page



The **Individual Photo Information Page** displays all related information about the chosen photo. There is a *fixed* navigation bar, that is present on every page on the website.

1. Search bar: executes search on keywords, then goes to the Search Results.
2. Login: Goes to the login/new user page
3. Choose from three media license plans. Clicking on the Buy button creates a pop-up stating “Thank you for your interest in this photo. A representative will contact you shortly”.

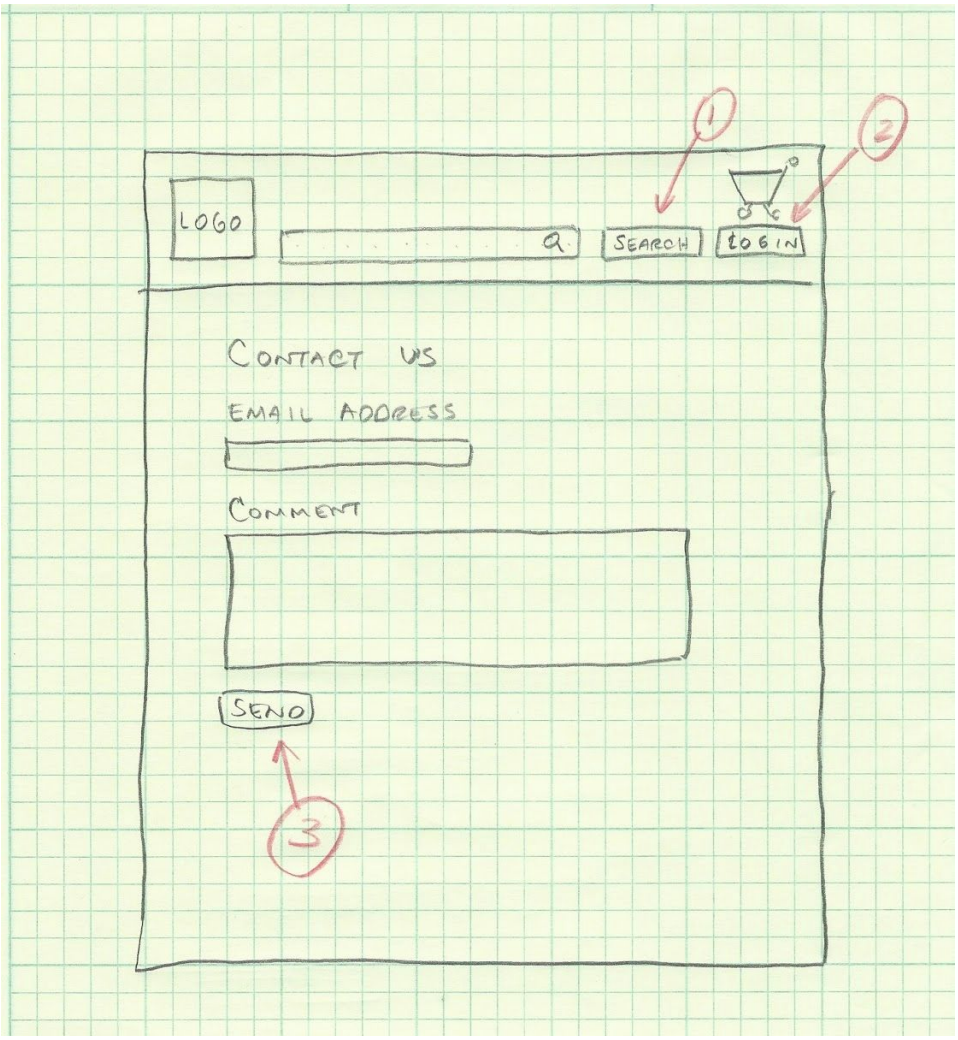
Login / New User Page



The **Login / New User Page** Allows users (customers or artists) to login to the website in order to view account transactions. There is a *fixed* navigation bar, that is present on every page on the website.

1. Search bar: executes search on keywords, then goes to the Search Results.
2. Login: Goes to the login/new user page (resets current page).
3. Login Button: Submits login information, on success transfers user to their account page.
4. Terms of Service checkbox must be checked in order to register
5. Register button logs in new users.
6. This hyperlink takes you to the Contact Us Page

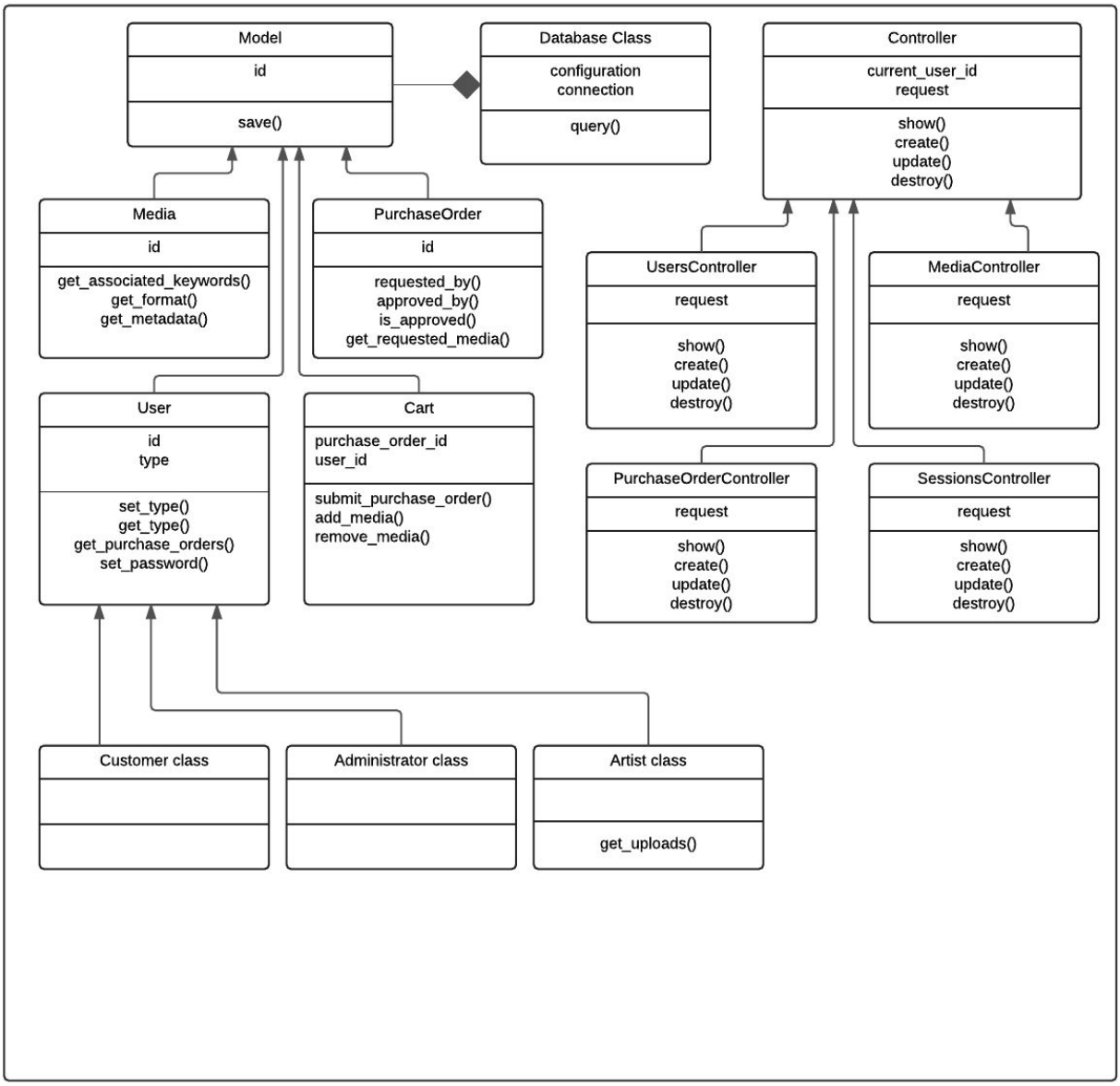
Contact Us Page



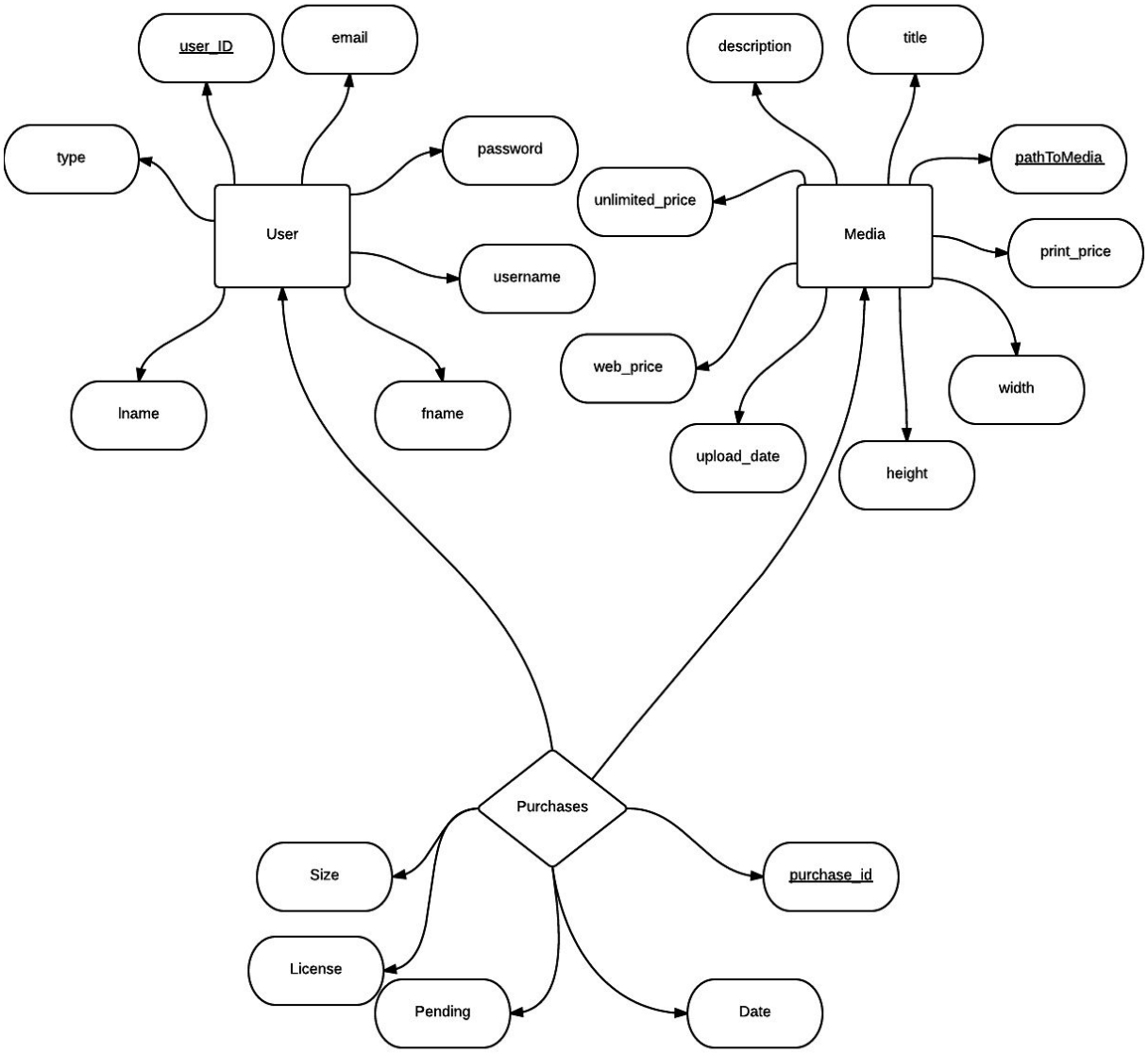
The **Contact Us Page** Allows users to contact the site administrator directly. There is a *fixed* navigation bar, that is present on every page on the website.

1. Search bar: executes search on keywords, then goes to the Search Results.
2. Login: Goes to the login/new user page (resets current page).
3. The Send button submits the information to the admin, as long as they have included an email address.

High-Level Class UML Diagram



High-Level Database Entity Relationship Model



Risk Assessment

Skills Risks:
In general the team has a good amount of working knowledge given to their specific tasks. However, universally the team lacks PHP, JavaScript skills. And while members have working knowledge, some are mostly basics that need to be expanded upon when advanced implementation is called upon.
The measures to resolve such skill risks are simple – research. Members show competent knowledge on their assigned tasks, and enough know how to further expand their technical skills. Should members become stymied in their assignment, other members may have a wider range of information on the subject in question to assist the member in need. Worst comes to worst, where a group member cannot acquire sufficient data to further progress, and other teammates have no way to lend a hand, the tech lead will show initiative to learn about the topic; And if the tech lead is successful in their endeavor, they will teach the other members to overcome the dilemma to further progress.

Schedule Risks:
As of now, the overall commitment to team meetings has been positive. All members have shown up on given scrum dates, unless complications arose, and meetings between team members to assist each other have shown encouraging results. Possible risks may include an issue in individual schedules. Members have displayed a large amount of work piled on them by other classes besides CSC 648. Issues may arise when one may need to forgo a scrum meeting in an effort to complete assignments for another subject. To counter this risk, the best course is to keep in communication to the rest of the group through the agreed upon communication medium. Letting other members know they will be absent allows attending members to convey scrum information to the aforementioned associate. Medical or personal issues may arise, causing a team member to fall behind in their responsibilities. Communication using Slack is in real time and is very effective in helping team members coordinate workloads.

Technical Risks:
Up to milestone 2, there have been very few technical risks. The testing server sfsuswe.edu has been working without fault. There is a .1 chance of server downtime and since the SVN documents are not replicated on each person's machine (like Github) we would be a delay in pushing new content.

Applications that are out of date, or are not compatible with a certain operating system can cause inconsistencies. Perhaps a member requires access to the net and is unable to receive a signal from his current location. More serious technical risks may include the team database server crashing, or even a mistake made in the repository when committing code. A member could have his computer die on him and lose all his work, which would be a huge setback to the rest of the team as well. Technical Risks can be avoided, for the simple ones at least, with

backup planning and researching ahead. If an application is out of date, find an update. Do some research to make sure a particular application is able to run on all needed platforms.

Teamwork Risk:

As of this moment, all members have displayed outstanding effort in their contributions to the group project. However, there has been issue in choosing methods for communicating outside of meetings. Some team members are texting and some are using Slack, This may lead to serious issues down the road when heavy coding comes into play. We will agree to only use the team account on Slack for communications and nothing else.

Another possible solution to this would be to address the issue directly during a group scrum, and devote some time to make sure members are aware that associates may be seeking aid; therefore, group members should be of a mind to check the communication medium(s) for any possible messages left by other team members.

Legal/Content Risk:

Legal risk is an ongoing issue that needs constant awareness. Anywhere from borrowing code, to using images, a license risky may come into play. The key, as mentioned earlier, is to be aware. Members should understand that when using another's code, or uploading an image, there may be legal consequences. They should careful read terms of use, if any, and properly site sources. The client is responsible for researching uploaded digital imagery for possible trademark infringement. The artists cannot directly upload to the site for this reason. Administrators should only add photos to the library after they have been cleared by the client's legal department. The developers are not liable and such conditions are stated in the work-for-hire contract.

Team 5 configuration

Name	Role	email	phone
Dana Muise	Team lead Project Management UML	dmuise@gmail.com	(415) 999-9034
Adolfo Von Zastrow	Tech lead PHP/AJAX/SQL	adolfovon@gmail.com	(415) 509-7653
Jay Naidu	SVN Master HTML/CSS Boostrap	jnaidu@mail.sfsu.edu	(650) 302-5791
Toru Nagoo	JQUERY / AJAX	inmylife777@gmail.com	(415) 813-0251
Nikolay Paulov	Data Base Documentation	socomdragon2003@yahoo.com	(415) 810-5982
Hargit Randhawa	SQL / Database	harjitr@mail.sfsu.edu	(916) 271-6749

WEEKLY SCRUM MEETINGS:
Tuesdays @ 2:00 SFSU library
Fridays @ 6:00 Class