SW Engineering CSC648/848 Section 01 Summer 2017

Jizhou Yang- Team Lead- jyang15@mail.sfsu.edu

Rafsan Saadi- CTO

Raghav Parti

Takahiro Odaka

Brandon Yu

Zach Melamed

Team # 3

Stock Photo Marketplace

Milestone 1

7/6/2017

Document Revisions

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

I. Executive Summary

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

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

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

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

II. Use Cases:

Unregistered users – Johnny visits the website by going on his laptop. He's new to the website, so he can browse the website and search for media content, but won't have the same privileges as registered users. He can search for media using categories, type of media, as well as any matching free text on the description. In addition, Johnny will be able to view low-resolution thumbnail of the image or video he is interested in buying. However, he will not be allowed to purchase media until he registers on to the website.

Registered users – Alice chooses to register to the website in order to purchase media content, via her email address/ID and password. After doing so, Alice is allowed to browse the website, search for media, and purchase media. Alice likes spending time on the website, and is more active. As such, she is given more privileges than unregistered users. Alice will be able to message contributors to set up buying of media. If needed, Alice is able to flag inappropriate or undesirable content on the website for an admin to see.

Admins – Jay is an admin who monitors the website and manages the media database to ensure a better user experience for customers. Jay has the privilege to remove media content, edit descriptions, manipulate media categories, and remove users. He will be in charge of maintaining the content on the website, but is not responsible for backend maintenance. However, Jay should keep a log of any server or database issues for ease of website maintenance.

Contributors – Bob is a freelance photographer. He wishes to sell his photos and videos. Through the website as a contributor, Bob is able to add an appropriate description/caption, set price point, and include various details about his media, such as size or length of the file. Bob should able to interact with website UI easily to promote more uploading of media and have an overall larger stock of media content available on the website. Bob should also be given full recognition of the media he uploads, and allowed to manage his own media content freely. Payment will be left for Bob and the buyer to decide through website messaging.

III. Data Definitions

- Registered Users: Users who have an account, and are logged into it, that are using the site. Registered Users can be both buyers and contributors.
- Guest Users: Users who do not have an account, and can only browse the site without being able to post items for sale or buy items.
- Buyer: User who has an account, and is buying an item.
- Contributor: User who has an account, and is selling an item.
- Administrators: Users that have special privileges, and have the ability to remove posts from the site, remove items, issue warnings and bans from the site, and generally enforce the Code of Conduct for the site. Administrators also are responsible for helping users when needed.
- Items/Services: Any object which is for sale, or for purchase on the site.

IV. Functional Requirements

1. Non-registered Users shall be able to search for photos and videos by using a keyword or category.
2. Non-registered Users shall be able to browse photos and videos.
3. Non-registered Users of the website shall be eligible to become Registered Users of the website.
4. Non-registered Users shall be allowed to view previews of videos on the website.
5. Non-registered Users shall be allowed to view previews of photos of the website.
6. Non-registered Users shall be able to view thumbnails of the images
7. Registered Users of the website shall be allowed the functions of Non-registered Users, along with the following:
    - Registered Users shall be able to purchase photos and videos.
8. Contributors of the website shall be allowed the functions of Non-registered Users, along with the following:
    - Contributors shall be able to upload photos and videos to the website.
    - Contributors of the website shall be able to set a "price tag" for their uploaded media. If a Contributor chooses to, they are allowed to put up media for free.
9. Admins of the website shall be allowed the functions of Non-registered Users, along with the following:
    - Admins shall be able to view the list of Registered Users and Contributors on the website.
    - Admins shall be able to maintain the list of Registered Users and Contributors.
    - Admins shall be able to remove content that is flagged as "inappropriate".

V. Non Functional Requirements

1. Application shall be developed using class provided LAMP stack
2. Application shall be developed using pre-approved set of SW development and collaborative tools provided in the class. Any other tools or frameworks must be explicitly approved by Anthony Souza on a case by case basis.
3. Application shall be hosted and deployed on Amazon Web Services as specified in the class
4. Application shall be optimized for standard desktop/laptop browsers, and must render correctly on the two latest versions of all major browsers: Mozilla, Safari, Chrome.
5. Application shall have responsive UI code so it can be adequately rendered on mobile devices but no mobile native app is to be developed
6. Data shall be stored in the MySQL database on the class server in the team's account
7. Application shall be deployed from the team's account on AWS
8. No more than 50 concurrent users shall be accessing the application at any time
9. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.
10. The language used shall be English.
11. Application shall be very easy to use and intuitive. No prior training shall be required to use the website.
12. Google analytics shall be added
13. Messaging between users shall be done only by class approved methods and not via e-mail clients in order to avoid issues of security with e-mail services.
14. Pay functionality (how to pay for goods and services) shall not be implemented.
15. Site security: basic best practices shall be applied (as covered in the class)
16. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development
17. The website shall prominently display the following text on all pages *"SFSU Software Engineering Project, Spring 2017. For Demonstration Only"*. (Important so as to not confuse this with a real application).

VI. Competitive Analysis

	download without register	Quick Search using Tag	Trend and popular product	upload format type	review downloaded product
Gratisography (http://www.gratisography.com/)	X	O	X	only photo like jpeg,png,.Gif	X
YouTube (https://www.youtube.com/)	X	X	O	only movie like .WMV.mp4FLV	X
Negative Space (https://negativespace.co/)	X	O	X	only photo like jpeg,png,.Gif	X
Shutter Stock (https://www.shutterstock.com/)	X	O	X	Only photo like jpeg, png, gif	O
Our Planning Product	O	O	O	include above photo and movie format	O

A. Summary:

Comparing our planning product and others, we attempt to have the multiplicity of uses for users. It can be uploaded many format types of photos and movies, such as, jpg, png, gif, mp4, and .WMV…, etc. We plan that the user can quickly access and find the media divided by each category and tag. Also, using Google analytics our website shows recommended and most accessed photo and movie. In addition, it is easy to download without any complex process like creating account. Of course users can create account which allows user to view history downloaded and uploaded. The user can review the photo and movies which uses get downloaded, and based on these rating we decide what picture and movie is most popular and interesting. We try to make the SW that any user equally can receive some fundamental service: search, view, and download the photo and movie.

VII. High Level System Architecture:

A. Primary Dev Component: Our application will be developed provided LAMP stack environment that consists of the following components:

1. Linux

a) *We are using Ubuntu Linux as our primary platform for both application development and deployment.*

2) Apache

b) *We are using Apache web server to allow users to access our application in the web*

2. MySQL

a) *User data will be stored and managed using MySQL database system.*

3. PHP

a) *PHP is the scripting language that allows creating web application with dynamic content.*

b) *Our application backend will be developed using PHP.*

B. Development Framework and APIs: We will be using the following framework and components which will allow rapid PHP application development along with front and backend solutions for our team application:

1. CakePHP

a) *CakePHP is an open-source web, rapid development framework that makes building web applications simpler, faster, and require less code.*

b) *Our application will be developed using CakePHP by utilizing their rapid development framework*

2. Bootstrap

a) *A web development framework that allow us to start on frontend development quickly by providing a series of templates and themes.*

b) *Allows to design responsive web pages for different screen sizes easily.*

3. JQuery

a) *JavaScript library that eases cross-browser development.*

b) *Helps to speed up frontend development by providing wrappers for many Java Script functionalities.*

4. Google analytics

a) *Our application will be integrated with Google Analytics Solutions which offers free and enterprise analytics tools to measure website, app, digital and offline data to gain customer insights.*

C. Application Deployment:

1. Application will be deployed on Amazon Web Services (AWS), the cloud-computing platform.

D. Supported Browser Matrix:

1. Following browsers must be used for faster and smoother use experience with our application:
2. Google Chrome- *Latest*
3. Mozilla Firefox- *Latest*

E. Other Tools:

1. MySQL workbench will be used for database maintenance.

IX. Team:

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

X. Checklist:

A. Team decided on basic means of communications- DONE

B. Team found a time slot to meet outside of the class- DONE

C. CTO chosen and working out well so far- DONE

D. Github master chosen- DONE

E. Team ready and able to use the chosen framework- DONE

F. Skills of each team member defined and known to all- DONE

G. Team lead ensured that all team members read the final M1 and agree/understand it before submission- DONE