CEN4010 Principles of Software Engineering, Spring 2023

Milestone 3 Final Project Proposal:

Overview: A website that users can used to save images from the web and put tags on them so they're easily retrievable

Team 03, Project Name: Overview

Team 03

Aira Torres

George Gordon

Daniel Rovira

Tyler Fasoli

Eric Smith

March 31, 2023

Table of Contents

Executive summary

Overview is an app that our team, Team 3, decided to begin testing and development on with the intent to provide our users with a functional, yet efficient way to save images from across the internet. Whether it is from Google, Facebook, or Twitter, our app has the necessities to save all of these images in one place, where our users are able to cement their photos either into a photo bank or a specific folder to be accessed and viewed with ease at a later date. A problem that most of our users have is not being able to find or save photos from other websites into a precise location that can be accessed later. They also may struggle with saving photos, but then at a later date, not being able to find or access these photos, because they have been misplaced or corrupted. Overview offers a swift solution to fix both of these problems.

With Overview, the user must create an account that allows them to access their own data and previously saved photos from their photo bank. Upon creating an account, the user then is able to create a photo bank within their Overview account that allows them to "bookmark" their photos for storage and implementation. The user only simply has to provide Overview with a url from any website of their choice, and the photos will begin to store in their account. Upon url submission, the users are also given the opportunity to provide tags, name, or pin certain images in order for those images to be even more easily accessed from before. Overview provides a simple way to restructure your photos and provides a safe place, free of corruption or confusion, to store your photos.

Overview was created with the intent to save both time and effort when saving photos from the internet and being able to access them at a later date. Team 3 came together to decide upon a well-structured app that would provide benefits to a large group of users, and with this app, the perspective of millions could be included. How many times have you decided to look back at some of your vacation photos from years ago, or a picture of a deceased relative or pet, or even just a picture that had made you laugh? With Overview, those opportunities to reminisce and look back on will not be wasted no longer. Overview removes the long process of manually scrolling through all of your old photos and not being able to pinpoint exactly where a specific photo is, and it allows the user, with the implementation of providing tags on images, to quickly and easily find the exact photo(s) that they are looking for. Especially when taking images from a large amount of resources on the internet, this can often prove to be quite the tedious task, having to look through each website or social media to find a specific photo, but Overview allows for the user to have a one place to stop and privately share all of your important photos for easy access. Overview provides both integrity and a multitude of features to save and cherish your photos from any era and any website.

Competitive Analysis

App Name	Overview	Pinterest
Audience	Everyday People	Everyday people
Price	Free	Free
Account Management	Viewing and saving images in account profile	Viewing, saving and putting albums of images on the site
Homepage	Yes	Yes
Add Image	Yes	Yes
Compile Album of alike images	No	Yes
Content	Appropriate images found on internet	Images that people find "aesthetic"

Main goal of OVERVIEW is to make is simpler for user to save and retrieve images they find on the internet that they wish to keep. It is known that Pinterest has this same logic. However, OVERVIEW enables user to name their images for a faster way to search for it once it's saved.

Pinterest

Pinterest is a known site for people who are looking for aesthetic images, ideas how to decorate their home and even recipes. In Pinterest user is able to make an account and save items they see on Pinterest that they want. It has been helpful for many people because it has such a wide variety of things you can look up.

Overview

Overview is similar to Pinterest because it is also able to save images you want to keep. The difference between Overview and Pinterest is the fact that Overview's main focus is image saving. Overview also uses URL from the internet to be able to save each images the user want. The user is able to name each image they save. Overview uses tags to be able to retrieve the images faster. For example an image of earth can contain tags such as space, outer space, world and so on. Overview also lets user see everything they have saved on their account by simply pressing on the see everything button.

Data Definitions

Name	Meaning	Usage	Comments
Overview	Domain Name	Use Case Scenario	A website that acts as an image saver. Users can save images and name it as long as they have the image url.
Images	User interface	User interface	Images that user found that they want to keep. It can be kept and organize in our website.
Pinterest	User interface	User interface	A platform where user can make albums of pictures that are alike and where user can share albums in public
Log in	Service	Site User Service	The first page the user sees. To have the full access to *Overview* the user must be a member
Create an Account	Service	Site User Service	Making an account to use the features of Overview
Homepage	User Interface	User Interface	The main page of Overview where user can see all the navigation bar containing all the tags they can use to search up images
Navigation Bar	Service	Site User Service	Where user can choose between searching up images by tags or looking at all the photos in our site.

Search	Service	Site User Service	If user need to search images using tags
Account Profile	User Interface	Use Case scenario	Main page where you can see the user profile and all the Album of photos they saved
Edit Profile	User Interface/Service	Site User Service	User is able to edit their profile on the site. They’re able to change their name and password. And write and “about me” page.
Image URL	User Interface	Use Case scenario/ User Interface	Obtained by user from outside our site. User need image URL to be able to save their images and retrieve it in our site.
Save Image	Service	Site User Service/ Use Case Scenario	User saving images from the web using its image URL and making it easy to find in their account by saving it under an image id and tags.
Delete Image	Service	Site User Service	Users are able to remove images that they no longer want in the account
Image Id	User Interface	User Interface	Enables user to name their images they choose from the web
Tags	User Interface	Site User Service	Enables a quick search of the image the user saved
Setting	User Interface/Services	Site User Service	Service where changes to your account can be made

Privacy	User Interface	Site User Service	Service provided to user where they can make their account private or public

Overview, scenarios, use cases

Member expectation

1. Adding Accounts

- **1a.** The system shall allow the user to link accounts across multiple supported platforms. The system shall not allow users to add platform accounts that are not supported by Overview or invalid account logins. Also, the system shall prevent the user from logging into an account if the User’s chosen password does not match the password field of the account .
- **1b. Actors**
    - Users
    - System
- **1c. Preconditions**
    - User has an active internet connection
    - System is available
    - User has an active account
    - User has social media account to link to Overview

1d. Flow Response Sequence

- i. User enters a UserID (same as email)
- ii. User enters a Password
- iii,.System shall check if UserID is available
- iv. System shall validate Password
- v. System shall confirm that the account was created to the User
- vi. System will have a button that logs user into the main page.

- 1e. Alternate Flows
    - i. User enters a UserID (same as email)

  - ii. User enters a Password
  - iii,.System shall check if UserID is available
  - iv. System shall validate Password
  - v. System shall confirm that the account was created to the User
  - vi. System will have a button to redirect the user back to the Main Page upon password fail

- **1c. Function requirement label**
- i. REQ 1.1 Creating Account

2. Loading Profiles

- 2a. User should be allowed to move between accounts seamlessly and have multiple accounts visible simultaneously at any given time. The system shall allow users to tab between or adjust the size of the tabs and view their profiles at all times. These tabs will be manageable in which they
- **2b. Actors**
  - Users
  - System
- **2c. Preconditions**
  - User has an active internet connection
  - System is available
  - User has an active account
  - User is logged into system
  - User has social media account linked to Overview
- **2b. Flow Response Sequence**
  - i. User selects tabs
  - ii.User can select multiple tabs at once
  - iii. User can add or delete pages that are unnecessary
  - iv. User can adjust pages width to fit to page or slide pages over
- **2e. Alternate Flows**
  - i. User enters a UserID of supported social media

  - ii. User enters a Password
  - iii,.System shall check if UserID is available
  - iv. System shall validate Password
  - v. System shall confirm that the account was created to the User
  - vi. System will have a button to redirect the user back to the Central Hub upon password fail
- 2c. **Function requirement label**
  - REQ 2.1 Loading Profiles.

3. Searching for Image

- **3a.** Users can search for an image by using a URL image link and it will be updated in real time. Users image search will be able to appear on the page and can be done from the profile
- **3b. Actors**
  - Users
  - System
  - Search Bar
- **3c. Preconditions**
  - User has an active internet connection
  - System is available
  - User has an active account
  - User is logged into system
  - User has searchable image link available
- **3d. Flow Response Sequence**
  - i. User selects search bar
  - ii. User can select image and retrieve URL
  - iii. User can add or delete images that are saved within account
- **3e. Alternate Flows**
  - i. User selects search bar

- ii. User can select image and retrieve URL
- iii. Image link is not valid
- iv. User's image not retrieved to site.

4. Naming Image

- **4a.** User can delete name and tag images
- **4b. Actors**
  - Users
  - System
  - Image
- **4c. Preconditions**
  - User has an active internet connection
  - System is available
  - User has an active account
  - User is logged into system
  - User has searchable image link available
- **4d. Flow Response Sequence**
  - i. User selects tabs
  - ii.User can select imgage
  - iii. User can add or delete tags that are unnecessary
  - iv. User can adjust name of tag/image after initial naming
- **4e. Alternate Flows**
  - i. User presses image
  - ii. User renames image but doesn't save renamed image

iii. Image does not get renamed

High-level functional requirements

Non-Member:

1. Creating an Account

- System should allow the user to create an account by using a UserID, Password, Date of Birth, First Name, Last name. The system will not allow the User to Create an account if the UserID chosen already exist in the System.User must choose a password and it must match when asked to re-enter it. Fields that have to be filled are First Name, Last Name, Location, UserID, Password, Re-enter Password.

- 1.2 Stimulus/Response Sequence

  i. User enters a UserID
  ii. User enters a Password
  iii. User re-enters Password for confirmation
  iv. User shall enter their First and Last Name
  v. System shall check if UserID is available
  vi. System shall validate Password
  vii. System shall store user Name and Password
  viii. System shall confirm that the account was created to the User

- 1.3 Function requirement label

  i. REQ 1.1 Creating an Account

2. Browse Public Accounts

- 2.1 User can use the search bar using tags to view alike pictures from other users
- 2.2 Stimulus/ Responsive Sequence
  i. User can use tags that goes with the picture they're looking for (ex: rainbow could have a tag that says colorful)
  ii. System will filter out tags that are alike to show user the images they're looking for.
- 2.3 User will not be allowed to delete tags off other people's images

3. About Us

- 3.1 System provides information about OVERVIEW and its functionality. It describes what it can and can't do. It gives users a summary of things that the site allows and don't allow.

- 3.2 Stimulus/Responsive Sequence
  i. User uses the navigation bar and clicks "About us"
  ii. User is redirected to About us page
  iii. System has a button that allows the User to return back to Home

Member:

1. Edit Profile

- 1.1 User is able to change their password or their name. Use can use the same password if they want but user must make sure to type the password correctly when re-entering their password in the “re-enter your password” slot. User is also able to add an “about me” paragraph if they want.
- 1.2 Stimulus/Responsive Sequence
  - i. User uses the Edit profile button
  - ii. User is redirected to a page that list the following categories: First and Last name, Username, Password, About me.
  - iii. User is not able to change their username but is able to change everything else.
  - iv. User is able to input a summary about their likes and dislikes on their profile
  - v. When user is done they press the submit button
  - vi. User is redirected back to their profile which shows their first and last name, username and their “about me” summary
- 1.3 Function requirement label
  - i. REQ 1.1 Edit Profile

2. Images saved

- 2.1 User will use Image URL to be able to save the photos they want. They will use our site to save their photo and give it a name. They can give it tags to be able to look up and retrieve the images faster.
- 2.2 Stimulus/Responsive Sequence
  - i. User finds the image they want/need online
  - ii. They log in to our site and uses the save image button
  - iii. User will see a pop up on their screen where they can paste the image URL
  - iv. User needs to fill in Image ID and tags for the images they’re saving
  - v. When users is done they press submit
- 2.3 Function requirement label
  - i. REQ 1.1 Save Image

3. User Privacy

- 3.1 User is able to make their account visible to public. Accounts visible to public can be viewed by nonmembers. If user doesn not want this user can make their account private.
- 2.2 Stimulus/Responsive Sequence
  - i. User goes to their profile and uses the Privacy button
  - ii. User will be redirected to a page where it has a private and public button

iii. Below the public button it explains that the user profile and images can be seen by nonmember and members.
iv. When done, user can press submit.

- 2.3 Function requirement label
  ii. REQ 1.1 Privacy

List of non-functional requirements

Performance Requirements :

1. Responsiveness: The system will also be responsive, operating on various monitor sizes, ranging from 10” netbooks to 24” desktop monitors. It will also be responsive with a wide variety of resolutions, from 1024 x 480 through 1920 x 1080.

2. Test Requirements: the test requirements for performance will include an expected load test as well as testing on all of the functional specs listed and their speed per transaction.

3. Number of instances: The project should be able to withhold at least 50 instances of the site at a time. And the speed of each instance will process 10-50 transactions per second within a 20-100 millisecond response time between each transaction.

4. Bug Count: The project will have no more than 6 bugs during the Beta testing phase and no more than 3 bugs can remain within the system after final release.

5. Execution Speed: Execution speed of the initial home page on a high-speed internet connection should load within 100-200 milliseconds, depending on the current cycle time.

6. Emphasis on usability, should be easier to add images when compared to using a native social media platform for image saving such as Pinterest

Ease of Use:

1. Training Time: Training time should be as much as it takes for users to learn

2. Accessibility: user interface items should be distinct and easy to discover

Interoperability Requirements:

1. Browser Compatibility: The system will be a web-based web app that will operate on at least two major browsers focusing on Google Chrome and Internet Explorer provide alternatives if the browser does not have JavaScript installed on it.

2. Computer and OS Compatibility: The system will operate on various types of operating systems, including Windows, OS X, and Linux. It will also operate on any type of computer which can run a browser that is supported.

Fault Tolerance:

1. Exception Handling: There will be exception handling provided in all situations where an exception could occur. This will provide the user with an explanation as to why an exception occurred and give them a chance to either input the correct answer or they will be taken back to the home page.

Security Requirements:

1. Login/Password System: Our system will have a login/password system to maintain the preferences, ratings, and reviews of our site's visitors and locals. This implementation will also require password confirmation upon creation. We will also ask the user for a security question that we will store along with the answer and if the user forgets their password, they will be able to retrieve it by providing the answer.
2. Encryption: The website will not be encrypted as there will be microtransactions or requirements for personal information, such as credit card numbers, will occur.

3. Security requirements: no personal data associated with the user, or their account should be stored. All relevant data retrieved via an API must only be used for the operation that the user has initiated; there will be no storage of any user-specific data.

Availability Requirements:

1. Times of Accessibility: the system will run 24 hours a day, 7 days a week unless for bug fixes or maintenance times. As long as the server is available the website will continue to run.

2. Maintenance: fixes or maintenance will be fixed as soon as possible and will be displayed in the event that the website is down.

High-level system architecture and database organization

Visual Studio Code is used to write the requisite files for the application. The files in question include HTML, CSS, JavaScript and PHP files. Bootstrap is the framework used to create the GUI and GUI functionality of the application and PHP that interfaces with MySQL serves as the basis for the backend. PHP feeds information to custom JavaScript functions to facilitate a more responsive end-user experience with the application. Additionally, jQuery is used in our JavaScript backend to make certain backend operations easier to implement, such as acquiring HTML DOM objects. Both frameworks used in this project, Bootstrap and jQuery, make use of the MIT license. All modern web browsers are supported including Chrome, Chromium, Firefox, Brave, Vivaldi, and Edge. A hierarchy of our software stack is listed here and serves as a representation of the high-level system architecture of the application:

- I. Graphical User Interface: browser_view
    - a. Bootstrap framework
- II. Generated HTML provided by:
    - a. JavaScript backend
        - i. Supplemented with data retrieved in PHP operations
        - ii. jQuery framework
    - b. PHP engine
        - i. Supplemented with data retrieved via MySQL operations
- III. MySQL Database
    - a. *user_accounts* database table
    - b. *images* database table

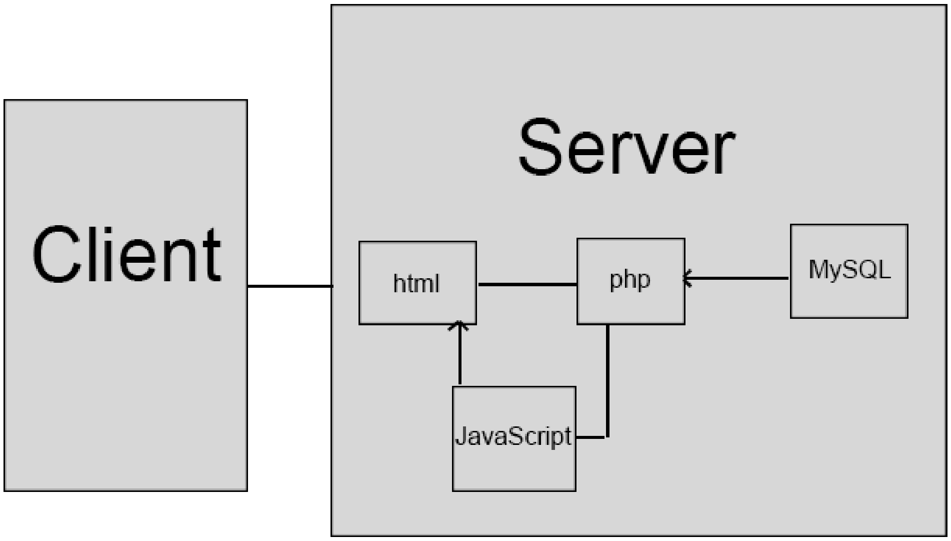
The MySQL database contains the *user_accounts* and *images* tables. These two tables fulfil the entirety of the application's data storage requirements including storing user credentials with salted and hashed passwords and storing saved image URLs, image hash identifiers, and user provided image tags. The following is a table serving as a representation of the aforementioned database tables:

user_accounts			
field	type	null	key
user_id	INT	NO	Primary
username	text	NO	
password	text	NO	
saved_img_hashes	text	Yes	
images			
field	type	null	key
image_hash	text	NO	Primary
filename	text	Yes	
saved_url	text	NO	
tags	text	NO	

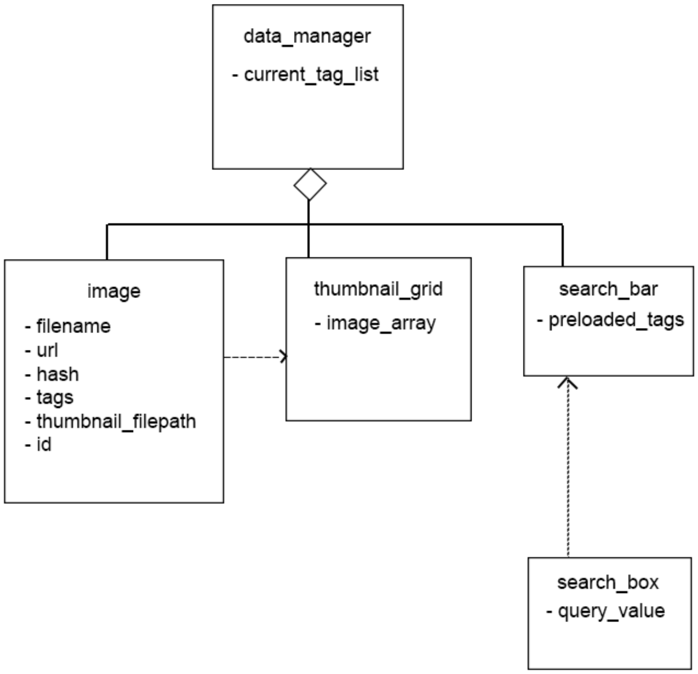
Items are predominantly retrieved using a combination of image hash and tag values.

High-Level UML diagrams

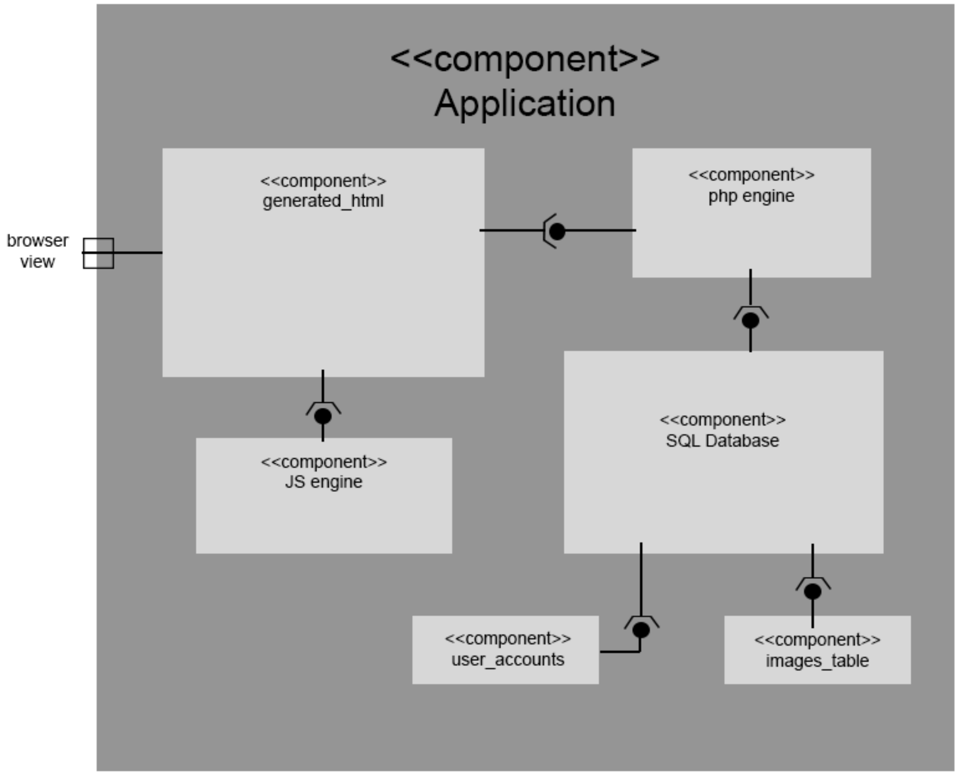
Deployment Diagram:



Class Diagram



Component Diagram



Identify actual key risks for your project at this time

- Skills Risks

With many of us coming from CS backgrounds and not necessarily focused on web design/ graphic design, we relied heavily on our minimal knowledge of these skills to support us within this project. Having to go in-depth into these skills, we took time to look over various codes and frameworks to give us an idea of the best possible solution we can handle and implement. Ultimately, we decided that Bootstrap and HTML were sufficient for us to re-learn and further develop to complete a project that would satisfy the requirements and expectations of this class.

- Legal/Content risks

Taking a closer look at our project, we looked into the logistics of developing a site that would be a great asset to social media users worldwide, but we bit off more than we could chew. With access to multiple highly secured social media sites, we wanted to bridge the gap between these platforms. However, we never considered the mass security breaches and the high effort it would take to produce a project in such little time. Thus, Overview will now focus on becoming an image storage and finder tool, which will allow us to still have the utility to cater to social media consumers and produce a product that is more realistic in terms of technique and time. This was a massive deal for our team because we needed to focus on a product that reflects our strengths and the learnings within this class.

- Schedule Risks

Also, we decided it was best to step back and redefine our product from the foundation due to conflicting time schedules and family emergencies. Also, we know that meetings are supposed to be daily; however, our team decided that we host daily check-ins and have team meetings on dedicated days so that we can see the progression of our individual tasks and have serious days where we consult on how we move forward within this project. This has been a more effective strategy that doesn’t compromise our previous work schedules to benefit the team overall.

- Teamwork Risks

As for teamwork, we need to have a stronger focus on the project as a whole; however, with this most recent Milestone (Milestone 3), we were able to find our struggles and what set us back previously in milestones and adapt to prevent them again. We ran things down the wire, and not having people come through to work or discuss project details initially took a toll on us as a team. With a more organized schedule and mid-term examinations behind us, we promised to come together as a team and help each other along the way rather than dedicate tasks and expect each other to complete a submission at the last minute.