Final Report - CSCE 606
Software Engineering
Aggieland Art Trail

1. Project Summary

The Aggieland Art Trail is a comprehensive web application designed to enhance the experience of exploring the vibrant art scene in Bryan-College Station. With multiple stakeholders in mind, including participants and administrators, the application provides a centralized platform for managing and enjoying the art trail.

For participants, the Aggieland Art Trail offers an intuitive interface that allows for easy navigation and discovery of various art attractions. Participants can explore the trail, learn about the artists and their exhibits, and engage with the artwork. As they visit different art attractions, participants have the opportunity to earn virtual stamps as a form of achievement and engagement. These stamps serve as a visual representation of their progress and active participation in the trail. By collecting stamps, participants can unlock virtual badges, symbolizing milestones or accomplishments reached during their art trail journey. The stamps and badge system adds an element of gamification and motivation for participants to explore more art attractions, fostering a deeper appreciation for the art scene in Bryan-College Station. The app also promotes community engagement by allowing users to share experiences on the Visual Art Society of BCS Facebook page. Real-time updates through blog posts on the website keep participants informed and inspired about the latest news, announcements, and activities, fostering their connection to the evolving art scene.

Administrators play a crucial role in managing the Aggieland Art Trail. They have exclusive access to the administrative view, which allows them to add, edit, and manage art pieces on the trail. This ensures that the information about the exhibits remains accurate and up to date. Administrators can also track participant engagement by monitoring the stamps collected by participants within specific time frames. This data provides valuable insights into the popularity and engagement levels of different art attractions along the trail. Additionally, administrators have access to website-wide settings, enabling them to configure the overall design and visual aspects of the application.

The web application aims to improve the overall experience for participants and administrators. It provides an interactive and user-friendly interface for participants to explore and engage with artwork, earn stamps, and share their experiences through the Visual Art Society of BCS Facebook page. Real-time updates through blog posts on the Aggieland Art Trail website keep participants informed and connected. Meanwhile, it offers a comprehensive platform for administrators to manage and optimize the trail. By fostering art appreciation, community engagement, and tourism, the application contributes to the growth and vibrancy of the art scene in Bryan-College Station.

2. User Stories Description

○ Interactive Map

Navigating the map

Points: 2 and Status: Completed
Enable participants to navigate and explore the locations of art pieces on a map.

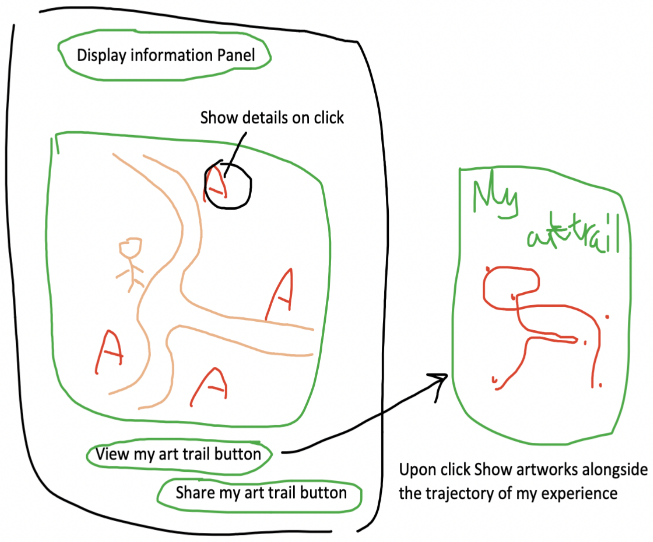

Display information Panel
Show details on click
A
A
A
A
View my art trail button
Share my art trail button
My
art trail
Upon click Show artworks alongside the trajectory of my experience

Figure 1: Mock UI for Interactive Map

Get current location

Points: 2 and Status: Completed
Display the current location of the participant, providing a convenient and interactive way to discover the art attractions.

Artwork around me

Points: 3 and Status: Completed
To access additional information, artist bios, and behind-the-scenes details, simply click on the marker corresponding to the art piece on the map.

Original: Ability to view only the art-pieces within a 20m radius.
Reason for Revision: Request for the user to have the freedom to explore any art piece.

Check-in to art piece

Points: 2 and Status: Completed
To check in at different exhibits along the trail and collect virtual "stamps" or points as a representation of participants journey. This signifies participants progress along the trail.

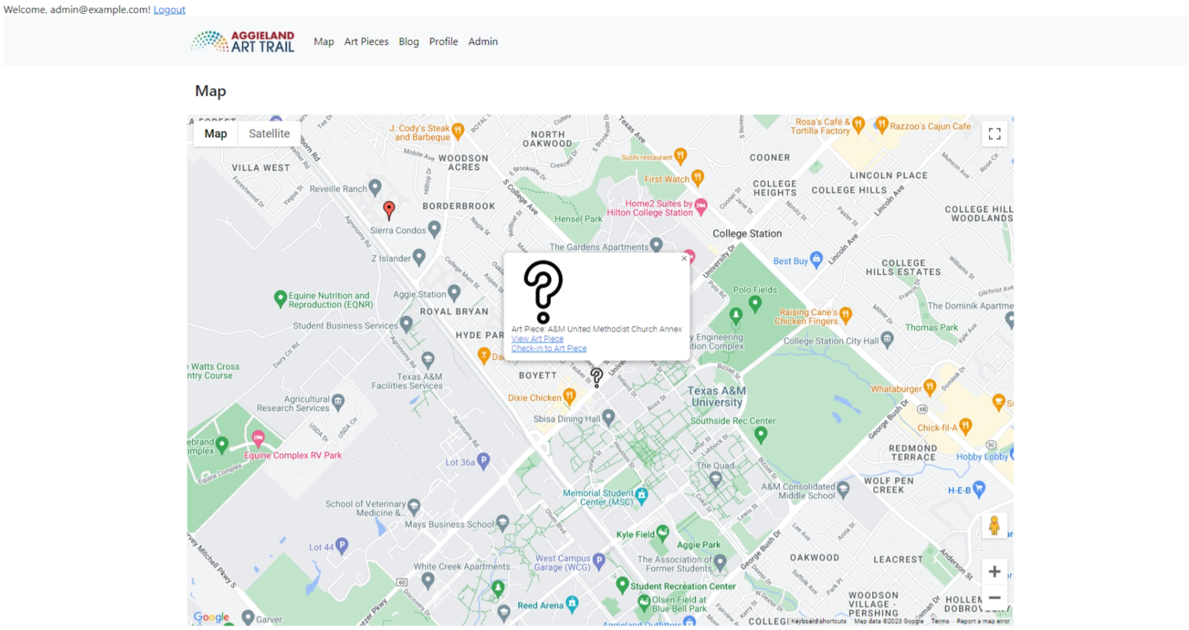

Welcome, admin@example.com! Logout

Figure 2: Interactive map

Authentication System

Login (Authenticate user)

Points: 3 and Status: Completed
A registered user should be able to login to the website with their login credentials.

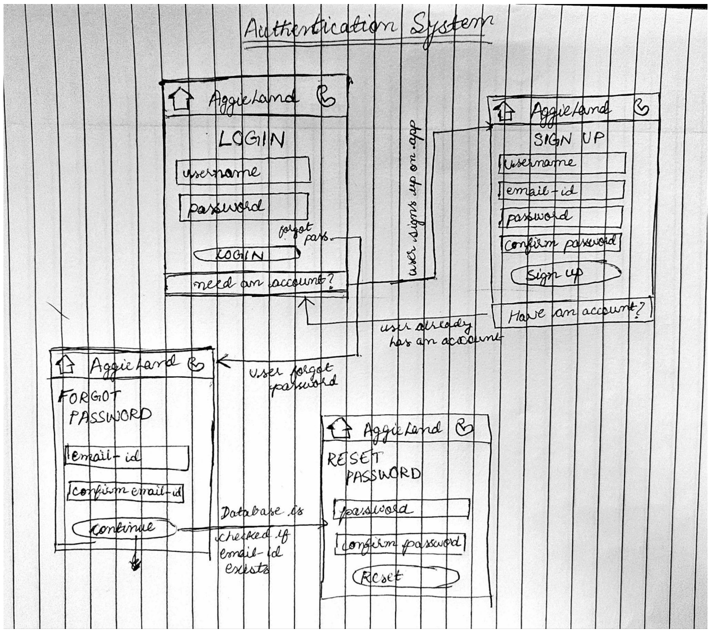


Figure 3: Mock UI for Authentication System

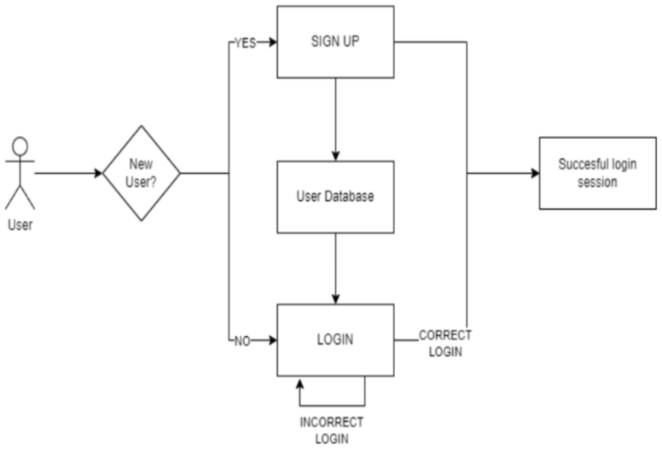


Figure 4: Control Flow for Authentication System

Sign up (Create account)

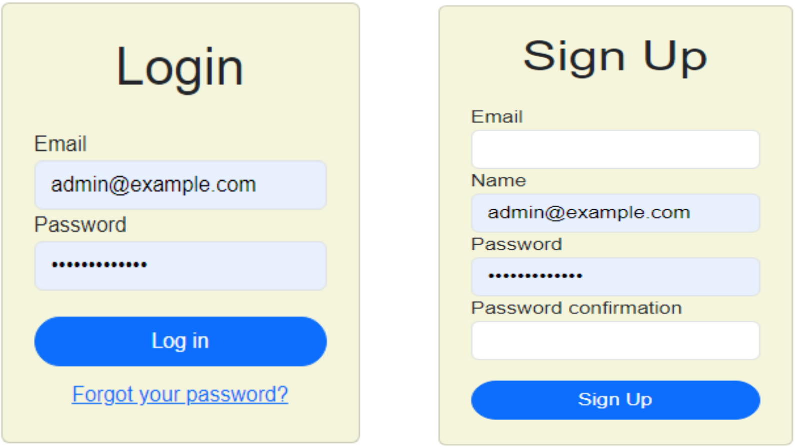
Points: 3 and Status: Completed
Enables users both participants and administrators, to create accounts by providing essential credentials such as email addresses and passwords.

Reset password

Points: 2 and Status: Completed
Offer an option to reset their password. Users can initiate the password reset process by selecting the "Forgot Password" option. They will then receive an email containing instructions and a secure link to reset their password.

Login session

Points: 2 and Status: Completed
To remember the users login session, eliminating the need to log in every time they visit the site. This aims to enhance convenience and streamline the user experience by allowing them to remain logged in across multiple visits without having to re-enter their credentials repeatedly.



Email: admin@example.com
Password: •••••••••••••

Email:
Name: admin@example.com
Password: •••••••••••••
Password confirmation:

Figure 5: Authentication System

Admin System

Create, Update and Delete art pieces

Points: 3 and Status: Completed
The administrator should have the capability to perform CRUD operations on the art pieces in the directory. This includes creating new art pieces by uploading images and providing relevant information, editing any details associated with the art piece, and deleting art pieces as needed.

Engagement Tracking

Points: 2 and Status: Completed
The administrator must be able to view the number of stamps that users have collected across a particular time frame. This data provides valuable insights into the popularity and engagement levels of different art attractions along the trail.

Admin Panel

Points: 2 and Status: Completed
The administrator must be able to give administration access to the website to the other users.

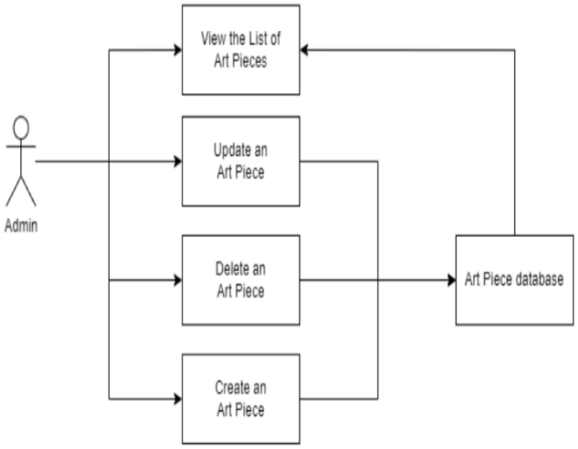
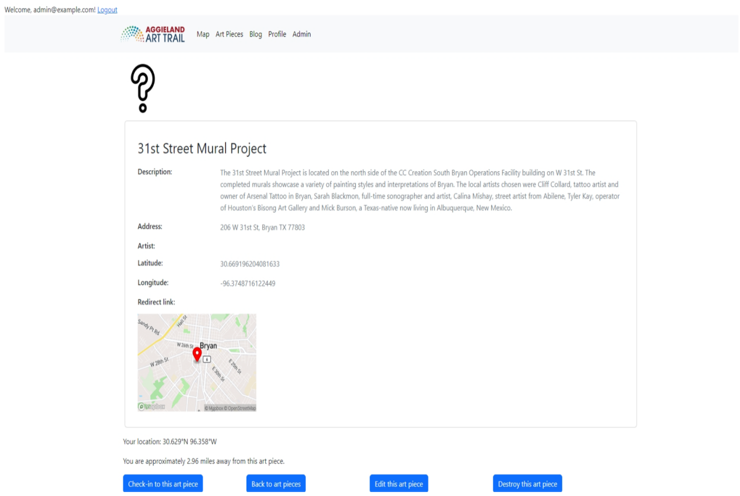


Figure 6: Control Flow for CRUD operations on Art Pieces



Welcome, admin@example.com! Logout

31st Street Mural Project

Description: The 31st Street Mural Project is located on the north side of the CC Creation South Bryan Operations Facility building on W 31st St. The completed murals showcase a variety of painting styles and interpretations of Bryan. The local artists chosen were Cliff Collard, tattoo artist and owner of Arsenal Tattoo in Bryan, Sarah Blackmon, full-time sonographer and artist, Calina Mishay, street artist from Abilene, Tyler Kay, operator of Houston's Bisong Art Gallery and Mick Burson, a Texas-native now living in Albuquerque, New Mexico.

Address: 206 W 31st St, Bryan TX 77803

Artist:

Latitude: 30.669196204081633

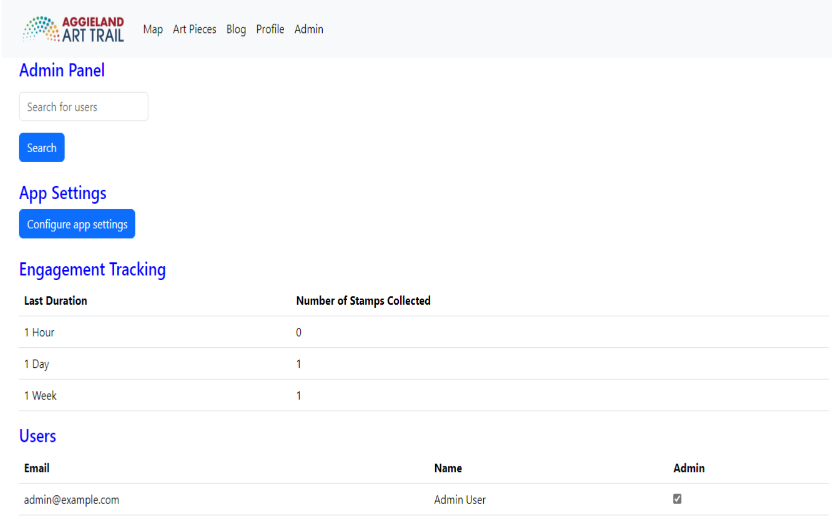
Longitude: -96.3748716122449

Redirect link:

Your location: 30.629°N 96.358°W

You are approximately 2.96 miles away from this art piece.

Figure 7: CRUD operations on Art Piece



Admin Panel

App Settings

Engagement Tracking

Last Duration	Number of Stamps Collected
1 Hour	0
1 Day	1
1 Week	1

Users

Email	Name	Admin
admin@example.com	Admin User	☑

Figure 8: Admin Panel

Badges and Stamps

View stamps and badges

Points: 3 and Status: Completed
Dashboard that allows the participants to view their collected stamps and badges. It also gives a preview of the stamps that are yet to be collected (grayed out).

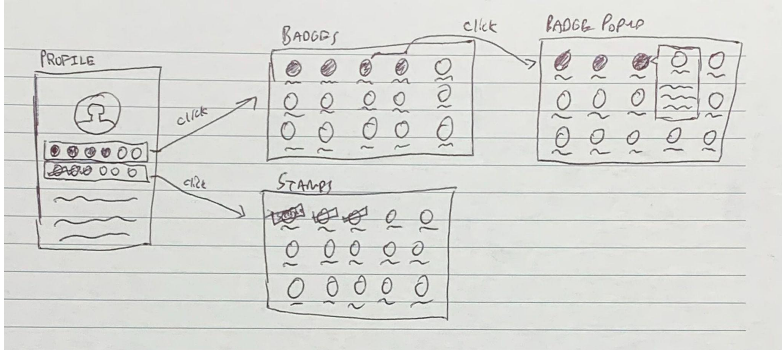
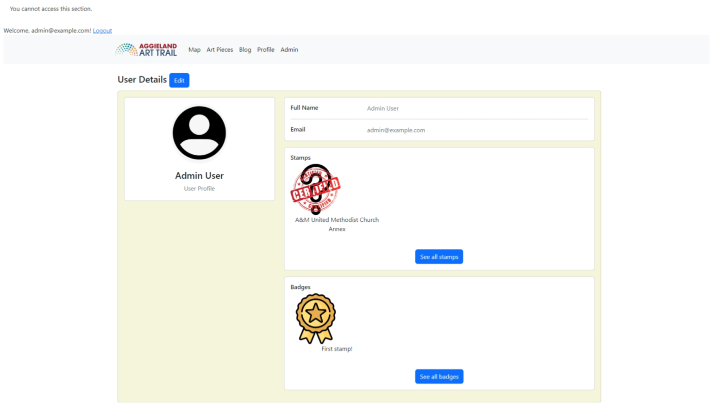


Figure 9: Mock UI for Stamps and Badges


You cannot access this section.

Welcome, admin@example.com! Logout

User Details

Admin User
User Profile

Full Name Admin User

Email admin@example.com

Stamps

A&M United Methodist Church Annex

Badges

First stamp!

Figure 10: Dashboard for Stamps and Badges

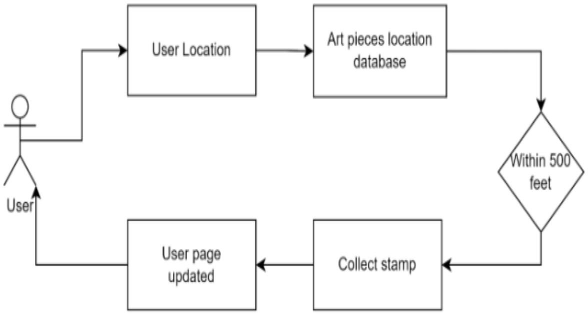
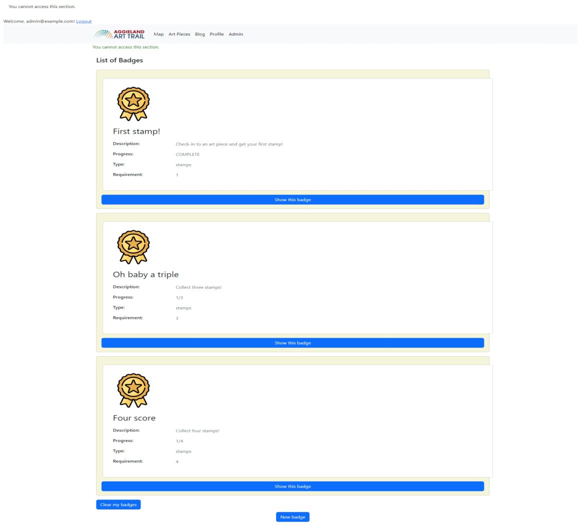


Figure 11: Control Flow for Check-In at Art Piece

Badge progress

Points: 3 and Status: Completed

Dashboard that presents the participants progress indicating the tasks they must complete to earn badges. This will offer participants a comprehensive view of their journey, providing motivation and guidance to fulfill the requirements and attain badges as meaningful achievements along the art trail.



You cannot access this section.

Welcome, admin@example.com! Logout

You cannot access this section.

List of Badges

First stamp!

Description:	Check-in to an art piece and get your first stamp!
Progress:	COMPLETE
Type:	stamps
Requirement:	1

Oh baby a triple

Description:	Collect three stamps!
Progress:	1/3
Type:	stamps
Requirement:	3

Four score

Description:	Collect four stamps!
Progress:	1/4
Type:	stamps
Requirement:	4

Figure 11: Dashboard for Badges Progress

Blog

Post to the blog

Points: 2 and Status: Completed
The administrator has the ability to publish news and announcements on the blog, ensuring that all users can view the updates. This feature enables the administrator to effectively communicate important information, upcoming events to the community.

Connect to Facebook group

Points: 2 and Status: Completed
A dedicated view with a link to the Visual Art Society of BCS Facebook page. This integration serves as a connection to the community and allows for the display of relevant information from Facebook, providing users with easy access to updates and fostering engagement between the website and the Facebook page.

Original: The client expressed a desire for a feature similar to a user-generated blog page, allowing users to upload posts and interact with them through likes and comments
Reason for Revision: After some consideration of the potential challenges of maintaining decorum and moderating explicit content on the blog, as well as the existence of an established community on Facebook, the client decided against implementing this feature on the website.

Groups

The client initially desired to provide users with the ability to join specific groups, where each group would have its own badges and activities. However, due to a lack of clarity on how this functionality should behave, the client made the decision to close this requirement and not pursue its implementation at this time.

3. Team Roles

Iteration	Product Manager	Scrum Master	Developers
0	Ashwin Raja	Hector Cardenas	Pratheek B R, Patrick Zhong, Srijith, Hanrui Chen, Hector Cardenas, Ashwin Raja
1	Ashwin Raja	Hector Cardenas	Pratheek B R, Patrick Zhong, Srijith, Hanrui Chen, Hector Cardenas, Ashwin Raja
2	Ashwin Raja	Hector Cardenas	Pratheek B R, Patrick Zhong, Srijith, Hanrui Chen, Hector Cardenas, Ashwin Raja
3	Ashwin Raja	Hector Cardenas	Pratheek B R, Patrick Zhong, Srijith, Hanrui Chen, Hector Cardenas, Ashwin Raja
4	Ashwin Raja	Hector Cardenas	Pratheek B R, Patrick Zhong, Srijith, Hanrui Chen, Hector Cardenas, Ashwin Raja
5	Ashwin Raja	Hector Cardenas	Pratheek B R, Patrick Zhong, Srijith, Hanrui Chen, Hector Cardenas, Ashwin Raja

4. Scrum Iteration Summary

○ **Iteration 0 (**Story Points: 0**)**

1) Gather the requirements from the client and features that have the highest priority.

2) Understood the requirements and got ourselves familiarized with the SCRUM process

3) Created Lo-Fi UI mockup for the Interactive Map, Authentication System, Blog, Stamps and Badges.

4) Delved into exploring Ruby and Rails to understand the implementation process for the desired functionality.

○ **Iteration 1 (**Story Point: 24**)**

1) Created the authentication system.

2) Implemented the admin capabilities of CRUD on art pieces.

3) Displaying the current location of the user on the webpage

- **Iteration 2 (**Story Point: 15**)**

  1) Used the Google Maps API to display the map and locations of all the art pieces and current location using markers.

  2) Enhanced the authentication system with reset password functionality.

  3) Implemented the upload image functionality for the art pieces for the admin.

  4) Added certain styling templates and CSS to the website.

  5) Bug fixes.

- **Iteration 3 (**Story Point: 12**)**

  1) Created the admin panel that allows the assignment of admin privileges to other users.

  2) Blocked access to the CRUD operations of art pieces for non-admin users.

  3) Created a user profile page where they can view their stamps and badges.

  4) Implemented the stamps feature. User could now checkin at the art pieces.

  5) Bug fixes.

- **Iteration 4 (**Story Point: 8**)**

  1) Implemented the Blog feature

  2) Enhanced the checkin functionality by adding a proximity check

  3) Shifted from CSS to Bootstrap to keep the styling cohesive throughout the website.

  4) Enhanced the stamps feature by including the concept of badges (milestoning)

  5) Bug fixes.

- **Iteration 5 (**Story Point: 12**)**

  1) Switched from PostgreSQL on Heroku to AWS because of dyno refresh error.

  2) Introduced the engagement tracking functionality to the admin dashboard.

  3) Enhanced the user profile page by adding the functionality of editing username, clearing badges and stamps.

  4) Created a new button on each artpiece that would link to the existing website to display more information about it.

  5) Bug fixes.

5. Individual Contribution

Team Member	Points
Ashwin Raja	14
Hanrui Chen	14
Hector Cardenas	14
Patrick Zhong	14
Pratheek B R	14
Srijith	14

6. Customer Meeting Summary

- **Iteration 0 (Sept 13, 12:00AM over Zoom):**

  During our first meeting, we discussed the client's project blueprint. They expressed the need for a platform where the community can come together and share their experiences exploring various art pieces. It is crucial to prioritize the implementation of social media sharing and stamping features to facilitate this interaction. Additionally, the integration of a map functionality was identified as a fundamental utility for the platform. We discussed how we can leverage our skill sets to effectively execute the proposed plan.

- **Iteration 1 (Sept 23, 11:30AM over Zoom) :**

  The customer feedback received after the demonstration was largely positive. They expressed satisfaction, specifically highlighting the method used to specify the location of new art pieces. During the meeting, we engaged in a productive discussion where we clarified the planned features and addressed any long-term questions or concerns raised by the customer.

- **Iteration 2 (Oct 11, 11:30AM over Zoom):**

  During the demonstration, we showcased the implementation of the login and sign-up flows. The customer expressed satisfaction with the level of detail and functionality presented. Additionally, we demonstrated the feature that displays location-based art pieces near the user. By setting a radius perimeter, we showcased how the app can accurately display art pins representing nearby locations. While the customer was generally pleased with the flow, they requested minor changes in the representation of pin pictures. We noted their feedback for further refinement.

- **Iteration 3 (Oct 18, 11:00AM over Zoom):**

  During the demonstration, we showcased the Info window functionality that pops up when a specific art piece is clicked. The customer appreciated this feature and suggested adding a button in the Info window to redirect users to their website for further information if desired. They were pleased with the ease of use when uploading images to the database for each art piece.

  Additionally, the customer requested that we populate the database with all the art piece data to provide them with a better understanding of the final product's appearance. They also recommended exploring ways to display the activity level of other users or art pieces within the app, allowing users to gauge the engagement and popularity of different pieces. We took note of their suggestions for further consideration and development.

- **Iteration 4 (Nov 1, 12:00AM over Zoom):**

  During the presentation, we showcased the updated website with added styling and CSS. The client expressed great satisfaction with the design, as it aligned with the color scheme of their existing website and provided a user-friendly navigation experience. We incorporated a navigation bar to enhance convenience and included specific details accessible only to admins.

  Furthermore, we presented the new user profile page, where users can view all the stamps they have collected, and demonstrated the stamp collection process. The client was impressed with the progress we had made thus far.

  During our discussion, we explored the future of the project and inquired about any additional features or changes the client wished to incorporate. Additionally, we had initially worked on a blog page; however, the client had a change of heart and requested us to integrate their Facebook page as a feed within the app. We made a note of these modifications and will proceed accordingly.

- **Iteration 5 :**

  During the presentation, we showcased the updated website to the client. They had encountered some issues with the signup and art pieces pages, so we promptly addressed those bugs and restored the working functionality of the affected links. Additionally, they requested to have their account elevated to administrator status, which we promptly resolved, granting them full access to the website and the ability to make changes to the list of art pieces as needed. To enhance the image handling capabilities, we migrated the entire image backend to an Amazon S3 bucket. We demonstrated the changes we made to the edit page, allowing the client to independently upload images and icons for art pieces, badges, and stamps. We engaged in thorough discussions to ensure that we had covered all the features they desired from our development efforts.

  Overall, we made significant progress in addressing the client's concerns, resolving technical issues, and empowering them with greater control over the website's content and functionality.

7. BDD/TDD Process

Our team used the Behavioral Driven Development (BDD) technique during the early stages of the Aggieland Art Trail project. We gained a thorough understanding of the requirements and objectives for the software program during our initial discussion with the client. Based on this first discussion, we generated a general list of user stories and made future revisions based on the client's requirements. We used the Test-Driven Development (TDD) technique throughout the development process to build the code in an organized manner and ensure that all of the expected features were implemented. We generated test cases for each unique functionality using frameworks such as Rspec and Cucumber before developing the code. This method enabled us to build a solid code foundation by systematically passing these tests, allowing us to implement varied situations and improve over time.

However, we did encounter challenges when it came to testing JavaScript. Initially, we were unaware that Cucumber did not support JavaScript, leading to failing tests. We experimented with Cucumber for JavaScript (cucumber.js) and deprecated Jasmine for Ruby, but neither provided a satisfactory solution. Eventually, we resolved the issue by separating the JavaScript code from the Ruby codebase and leveraging Jasmine for testing. This trial-and-error process taught us the importance of early separation and effective tooling in JavaScript testing.

8. Configuration Management Approach

In total, we had 24 branches for the project. However, we encountered one duplicate branch that needed to be resolved. Additionally, after pushing the initial user profile branch, it did not interact correctly with other branches when merged into the main branch. As a result, we created another user profile branch to address this issue.

To effectively tackle the development tasks, we divided the app into subtasks: map, stamp, user, and admin profiles. Each group member selected a task and created their own branch to work on. Once a task was completed, we conducted thorough testing before merging it into the main branch. Some branches were created specifically for testing purposes. In certain cases, test code was written by a different person, leading to the creation of separate test branches. Additionally, test branches were created to ensure that changes were confined to specific files under a particular directory within a branch, thereby enhancing clarity. To facilitate task management, we assigned tasks on Wednesdays and held regular discussions on Mondays to track progress and ensure task completion. Once a branch contained completed functionality, it would be merged into the main branch, contributing to the overall progress of the project.

9. Issues with Tools/Deployment

We initiated the deployment process using PostgreSQL on Heroku. Initially, everything worked well when there were no images in the system. However, after a few iterations, the uploaded images started disappearing from the database. We identified the cause of this issue as the dyno restarting at certain intervals. To address this problem, we decided to migrate to AWS. We leveraged S3 buckets to store the images, ensuring their persistence. During the project deployment verification, the other team identified a versioning discrepancy between the local system and the one deployed on Heroku. We promptly investigated and resolved this Ruby versioning issue.

Another challenge we encountered was related to AWS, but not directly caused by it. For a considerable period, our map functions worked flawlessly locally but failed to function correctly on Heroku. After reviewing the backlog, we eventually discovered that the problem stemmed from Rails assets not being parsed correctly in JavaScript. To rectify this, we resolved to upload all assets or files related to JavaScript to an AWS web service, ensuring their proper functioning. Throughout the deployment process, we encountered and successfully addressed these issues, ensuring the stability and functionality of our application.

10. Tools & Gems Used

During the app development process, an assortment of gem files were utilized to incorporate various features.

1) bcrypt: Secures the passwords of users
2) bootstrap: Simplifies the UI design process by providing predefined classes for easy and responsive layout options.
3) sqlite3: Database engine used in the development environment
4) pg: Postgres is the database engine that is used when deployed to Heroku. Postgres is the default database that Heroku prefers.
5) capybara: Integration testing tool for rack based web applications. It simulates how a user would interact with a website
6) cucumber-rails: Testing tool that helps with behavior-driven development
7) rspec-rails: this is a testing tool that helps with Test-driven development and unit testing
8) faker: Helps in creating fake names, addresses etc. used during the testing process.
9) factory_bot_rails: Creates objects for testing without directly calling the constructor
10) simplecov: Finding the code covered by the BDD and TDD tests, which will make sure that all the code that is being written is tested.
11) geocoder: Converts a given address to latitude and longitude.
12) Jasmine: Testing framework that is used to test the javascript code.
13) aws-sdk-s3: Integrates the Amazon Web Services S3 bucket to rails to make uploading images using the website possible.

Additionally, we have used a couple of APIs:

1) Google Maps API: Displays an interactive map, through which we can view the locations of the user and the art pieces
2) Facebook API: Integrates the visual art society's facebook page with the blog page present in the app.

11. Repository contents and deployment process

Our repository is organized into three main categories based on the Model-View-Controller (MVC) architecture of Ruby, which governs the application's logic, content rendering, and user connection. Each category corresponds to specific use cases, named accordingly. Additionally, we have GemFiles for imports, representing a component of Ruby's architecture. Cucumber testing files are also included to test our features. The documentation folder contains all five iteration reports. The repository also contains the app's code and comprehensive instructions for manual installation, testing, and setup from scratch.

We do not use Docker and would require the user to set up the app manually, we have provided detailed instructions that one can follow to do so. To completely set up the app from scratch, the user will have to set up their own Amazon account to create a S3 bucket, which is responsible for holding all the images that we use throughout all the pages of the app. To get started, clone the repo and follow the instructions to create a new AWS S3 bucket. If you want to continue to use the S3 bucket that we created, contact one of us for the master.key file. The next step would be to run the app on a local machine using the usual array of rails commands, starting from bundle install to rails server. The detailed steps are given in the readme file on GitHub. To run all the testcases, simply run a migration on the test environment followed by 'bundle exec cucumber' and 'bundle exec rspec'. The final step would be to host the website on Heroku. All the steps to do so in the readme file on the GitHub repository. After successfully completing all these steps you would have a fresh app running on Heroku.

12. Important Link

Project management page: GitHub Project Management
Github repo: GitHub Project Repository
Heroku deployment: Aggieland Art Trail
Presentation and Demo: https://youtu.be/qMU_K_B7FYI