**Final Report**



Caroline Krouse

Aaron Cesar

AJ Atkins

Megan Morton

Capstone Project, Spring 2020

The University of West Florida

April 24th, 2020

Capstone Systems Project CIS4595

Dr. Ownsnicki-Klewe

# 

# **Abstract**

In this report we will be summarizing the entirety of our project, Game Geo. This will include an overview of the final requirements of the project and how those final requirements compare with our initial requirements. Additionally, we will also review the final timeline of the project and how it compared to the initial timeline. We’ll go into detail about the project results and expectations, the project process, and the work that still needs to be done on the project.

Table of Contents

[Abstract 2](#_Toc38632754)

[1 Executive Summary 2](#_Toc38632755)

[2.1 Comparison with Initial Requirements 4](#_Toc38632756)

[3 Final Timeline 4](#_Toc38632757)

[3.1 Comparison with Initial Timeline 5](#_Toc38632758)

[4 Project Results Compared with Expectations 6](#_Toc38632759)

[5 Project Process Review 7](#_Toc38632760)

[6 Work to be done 8](#_Toc38632761)

[7 Acknowledgements 9](#_Toc38632762)

**Table of Figures**

[Figure 1: Final Timeline………………………………………………………………………………………….5](#_Toc38633196)

[Figure 2: Initial Timeline………………………………………………………………………………………..6](#_Toc38633197)

[Figure 3: Final Timeline………………………………………………………………………………………….6](#_Toc38633198)

[Figure 4: Pictionary Game………………………………………………………………………………………7](file:////Users/carolinekrouse/Downloads/The%20final%20FINAL%20REPORT.docx#_Toc38633199)

**1 Executive Summary**

Game Geo is a location-based game where users can play games like Pictionary with other users asynchronously. This means that users can play or create games at a specific location and other users can interact with the games at that location regardless of whether or not the first user is still there. Our goal was to create a minimum viable product (MVP). We defined our MVP as successful when we are able to track the user’s location and display it on a map, allow users to create a Pictionary game for other players at their current location, and allow players to interact with the Pictionary games other users have created. Our goals for the project was to be able to have a map that is centered on the user’s location, the ability for users to create games at their current location, implement a player profile, and allow users to create and play Pictionary games. The crucial requirements for Game Geo include a fully functioning map, a way to create and Pictionary games, and a database to store the images and related info about pictionary games.

We used an agile development approach for our project and completed it in five sprints. The project did not deviate too far from the anticipated project timeline. However, due to a combination of different complications, our timeline and goals were adjusted to accommodate the additional strain of not being able to have physical meetings and other complications associated with Covid-19. We were unable to implement a user profile system as a result of the complexity and time-consumption of making sure the geolocation portion of the game is operating as intended. There is still work that needs to be implemented including a player profile system allowing users to sign up and authenticate using their Google account, a visual statistics system that aggregates data from all of the games a user has played, and the addition of other games such as trivia and hangman. Despite the work that still needs to be implemented, we were able to successfully accomplish our goal of having a MVP by the end of the five sprints.

**2 Final Requirements**

One of the most important requirements for our final Game Geo release is the Map. The Game Geo map is what the application is primarily focused around. The Map displays the user’s location and a small area surrounding them. The Map provides a way for the user to see nearby game challenges and interact with them. Another feature that is required of the map is to provide a way for the user to create and place challenges of their own. We poured a lot of time into getting this requirement as polished as we could because the Map makes up so much of the user’s experience.

Another equally important requirement for the application is the Pictionary game. Having a functioning map and game is what makes the application Game Geo. The Pictionary game is also directly tied to the Map requirement because as a user creates a drawing, they are creating a Pictionary challenge that will appear on the Map. The most important feature of the Pictionary requirement is that the user is able to draw. In addition to the drawing feature we added the ability to choose different color brushes as well an eraser tool.

The Database is also crucial to the app working correctly. As a user creates a game challenge by drawing a picture the application must store this image and other related info in the database. The application periodically accesses the Database to populate the Map with game challenges. All of these requirements are major components of Game Geo and that is why they made it into the final requirements of the project.

## **2.1 Comparison with Initial Requirements**

Game Geo had several initial requirements that were never fully addressed. One of these requirements was a user profile. We wanted to include a user profile in the application to have a way to track a player's statistics, and to give them another way to interact with other players. Alongside a player profile were two requirements, user authentication and friends list. In order to sign in or create a user profile we wanted the user to have the ability to sign in with Google. Once signed in a player should have a friends list where they could share stats with friends and leave comments on game challenges.

Tying into the user profiles, we had initially planned for game challenges to award points or increase a player’s score. We had hoped to give more points the faster a user could complete a challenge. These scores or points would be tracked on their profile. Additionally, these stats and times would be recorded for all players in a region such as by state or zip code. The stats would then be displayed in leaderboards showing which state or zip code was better at Pictionary.

Another initial requirement that never got introduced into the application was different challenges. Game Geo was initially supposed to have two types of game challenges, Pictionary and Trivia. We had planned on developing two games with the hope of introducing more games later. Unfortunately, we were only able to produce a Pictionary game.

# **3 Final Timeline**

By the time our project reached completion, we had all of our MVP requirements completed. Our last sprint was probably one of the toughest because we had all these” final touches” we wanted to complete before submission.

Prior to Spring Break, our team met every Tuesday, like clockwork. Post Spring Break, we met on a Friday, a Sunday, a Thursday, and only *once* on a Tuesday. It was extremely difficult to meet weekly, so we relied on Discord for most of our communication.

Our team held a few working sessions, but it was nearly impossible for us to get everyone involved at the same time. We stopped relying as much on our tasks management system, Trello, and started simply posting small updates via Discord.

A close up of text on a white background

Description automatically generated

Figure 1: Final Timeline

## **3.1 Comparison with Initial Timeline**

Compared to our initial timeline, our sprints ended up extending until right before our final submission was due, giving us an extended (three weeks instead of two) **sprint 5**.

Taking into account the transition period from in-person to online courses, our sprint goals changed, and some tasks took longer than we initially anticipated it would take us to complete.

At some point our methodology was changing for some tasks. We anticipated our authentication additions would be done in an agile method like the rest of our project, but it fell into a more Waterfall approach leaving us waiting for the final product to work on the additional security features. This caused our Initial timeline to extend sprint 5, as mentioned above, and not complete some of the features we initially wanted to.

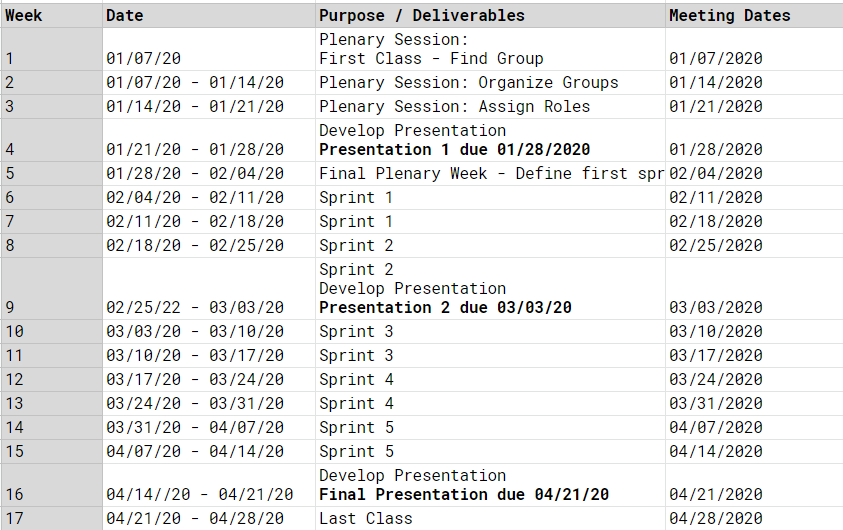
**

Figure 2: Initial Timeline

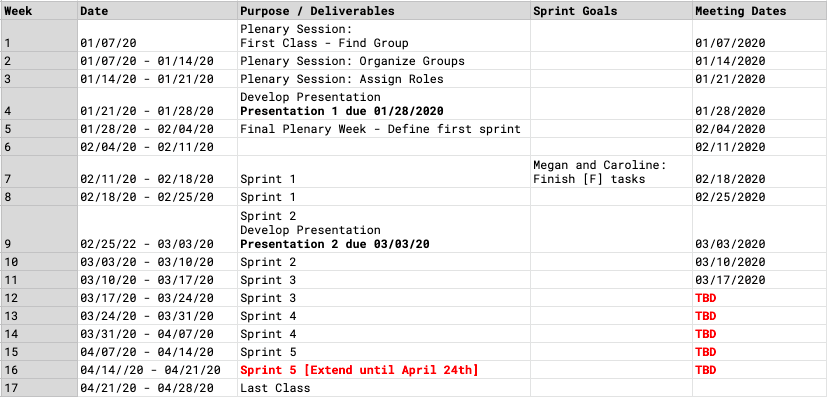
******

Figure 3: Final Timeline

# **4 Project Results Compared with Expectations**

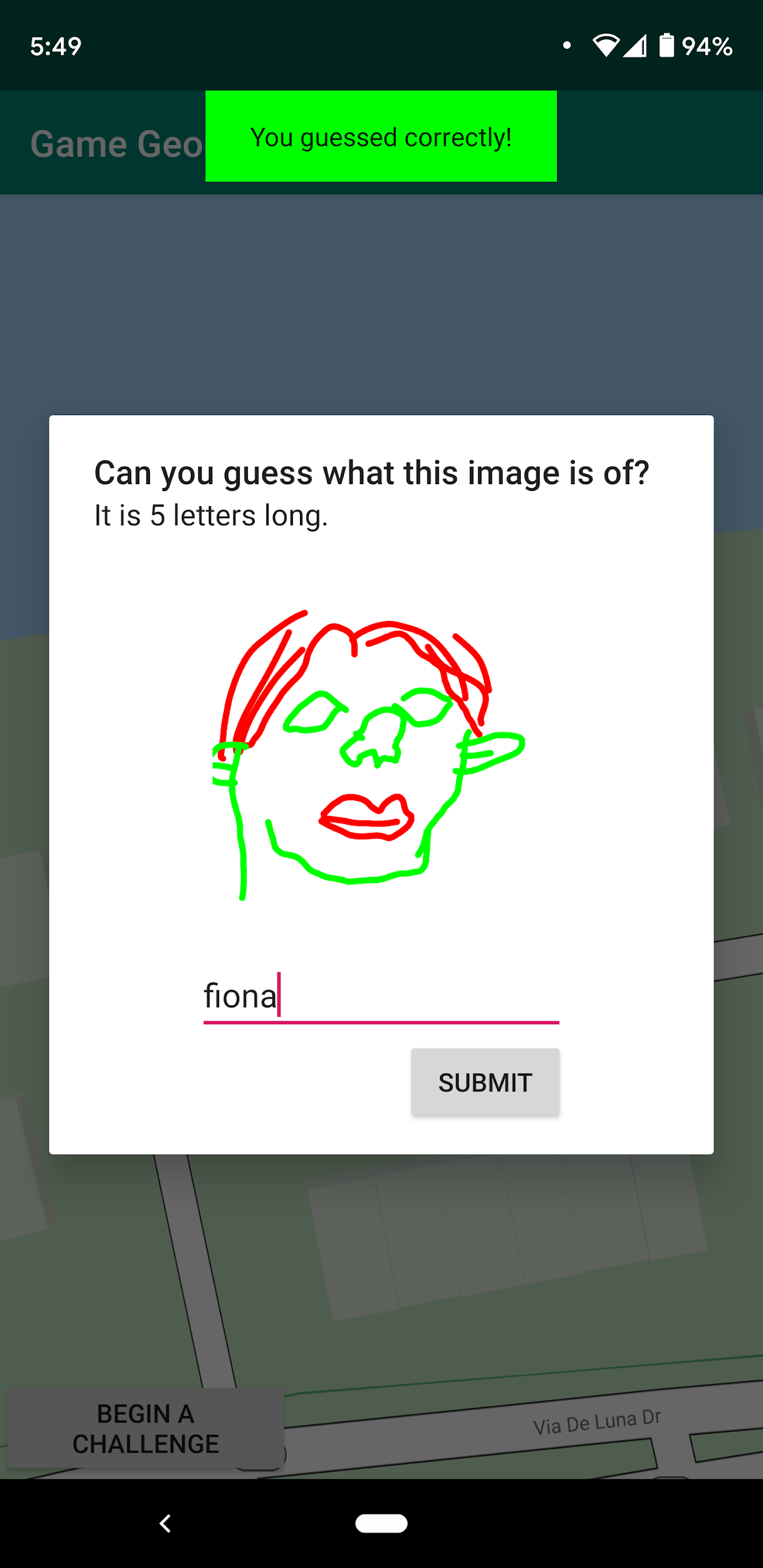
From the start of our project, the expectation was that we would make a geolocation-based game with a single game and then potentially build more games if we had enough time. In its current iteration, we were able to create a working version of Pictionary that users can play with others. The application is hooked up to a cloud hosted database and users can leave challenges that other players can view. We were not, however, able to develop any games besides Pictionary.

Figure 4: Pictionary Game

*Figure SEQ Figure \\* ARABIC 4: Pictionary*

Due to the amount of time it took to develop a geolocation-based Pictionary game, we were unable to create user profiles, and this has slightly stifled the progress of the Pictionary game. If we had player profiles, then we would send users notifications when someone had successfully answered a challenge they had left. Currently, when a user answers a Pictionary challenge they are notified if they were correct or incorrect, however, the user who created the image is not notified.

Overall, we were able to complete a significant amount of our project, but just fell short of being able to complete all the features to make the Pictionary game complete. For the Pictionary game to be complete, we would need to have the user who created the image be notified when someone guesses their picture correctly, and both the user who drew the image and the guesser would be awarded points.

If we had completed the Pictionary game, we hoped to develop a comment system where users could leave comments on pictures others had drawn. Additionally, we would have begun working on the trivia game. These features were not expected to be completed, however, and were additional features we would have completed if we had the time.

Overall, our team is pleased with the progress of our application, as it is in a working state and can be used to play games with other players. The core of our application is its integration with Google Maps as well as drawing and viewing challenges, and all of these functionalities are in place. Additionally, we plan to work on the project further to complete some of the missing features and, eventually expand on it to create multiple games.

# **5 Project Process Review**

Our team used the agile approach to development, in which we developed our application in five sprints. At the start of each sprint, we met as a group and discussed what needed to be done on the project. Additionally, throughout each sprint we would frequently have meetings where we talked as we developed the project. While developing, we worked on the system in vertical slices, that is we developed each layer of the application stack during the same sprints, rather than developing each layer separately.

To keep track of what needed to be done as well as what was in progress, our team used Trello. At the beginning of our first sprint, we created a list of user stories that needed to be accomplished and worked through them throughout the semester. This list did go through some major changes, however, as we did not have the foresight to determine every task that needed to be completed for our project. Trello was useful to make sure that we didn’t miss anything and to ensure that all members of the team could always have something to work on.

For the agile process, our meetings were a crucial component in ensuring that our sprint goals would be set and worked through. These meetings were held in person every Tuesday for approximately 2-3 hours. Once classes became remote, however, we began having to hold the meetings online via Discord. While these meetings were sufficient, they were not as effective as the in-person meetings. Being able to work together in person allowed us to easily look over each other's shoulders and collaborate on the code as well as Trello. Through Discord, the closest alternative we had was screen sharing, however, this was not a perfect substitute to in person meetings.

Throughout the course of development, we did not have a project manager that assigned tasks to members. Instead, each member was required to choose a task to work on from Trello and develop that task. Due to the nature of our project being that many systems required other systems to be in place first, however, we had some minor difficulties with this. For example, being able to view other players' pins required that the database be set up, however our team faced some difficulties hooking the application up to the database. Due to these difficulties, the team came together to get the database working, which took time away from other parts of development. If we had developed in horizontal slices, this would have helped alleviate this issue, however, we worked on vertical slices and therefore were working in parallel on the layers of the system.

All in all, the agile approach to development was a good choice for our team because it allowed us to rapidly develop our system. If we had gone with a traditional approach in which we developed in horizontal slices, we would have risked only completing one layer of our system and not having a complete system that was functional. Additionally, the agile approach to development allowed us to further define tasks as we went along, which was beneficial since we did not have clear cut requirements from the start.

# **6 Work to be done**

Some work to be done for Game Geo that wasn’t able to be completed was things such as player profiles. We wanted to have the ability for users to authenticate and login through Google. Once logged in users could customize and view their profile where they would have things like a profile picture and a score. Awarding points to increase a user’s score would create an incentive to keep playing. Additionally, users could give points to other players' creations if they liked them.

Another feature we wanted to add was a friends list so users can compare their score with their friends. We felt adding a friends list would help get more players on the app as users tell their friends about it. We also wanted to add another layer of interactivity by providing the ability for users to leave comments on challenges.

Another part of Game Geo is that it was supposed to be a game suite of location-based games. We had the intentions of having two or even three games but only got around to implementing the Pictionary game. We had ideas for other games such as Trivia and Hangman.

Additionally, we have yet to implement a visual representation of gameplay statistics. If we had finished all other work that needed to be completed, we planned to allow users to view location-based statistics, such as what city in the United States successfully answered the most trivia questions. This could also lead to things such as highlighted Pictionary images that got the most amount of points.

Finally, some features of our system should be optimized were we to publish the app. There is some level of optimization with location services, but we need to further optimize how data is being pulled from the database. Currently, all pins are being pulled from the database when the user loads the application, however, we plan to alter the application so that only nearby pins are pulled down.

# **7 Acknowledgements**

We would like to acknowledge some sources that helped us immensely while developing Game Geo. One source in particular that helped us was Google’s Google Maps Documentation. We found many answers to our problems using their documentation. Another great source of information for developing in an Android environment was the textbook *Android Programming: The Big Nerd Ranch Guide.* For setting up the database and authentication we got a lot of insight from a guide written by Ashraff Hathibelagal.

<https://developers.google.com/maps/documentation>

<https://www.bignerdranch.com/books/android-programming-the-big-nerd-ranch-guide/>

<https://code.tutsplus.com/tutorials/how-to-use-mongodb-stitch-in-android-apps--cms-31877>