**GameMate**

By: Anissa Braca and Daniel Vallejo

CSIS 4903: Capstone Project for Computer Science

Nova Southeastern University

Dr. Ajoy Kumar

April 28, 2024

**Table of Contents**

Chapter I. Introduction….….………...……………………………………………………………3

Chapter II. Review of the Literature..…………….……………………………………………….6

Chapter III. Methodology……………………………………………………………………..…..8

Chapter IV. Resuts…………………………………………………………………..………..….10

Chapter V. Conclusions………………………………………………………………...……..…10

Back Matter………………………………………………………………………………......…..12

**Chapter I. Introduction**

Relevance and Statement of the Problem

Video games, one of the most popular hobbies here in the United States. Video games have been around for a really long time, but online gaming became really popular since 1970s starting with Multi User Dungeon (MUD1) that was created in 1978. This game brought many gamers from all over the world to play and interact with each other. As games kept on developing, so did online gaming a place where players can come play and compete with each other. The biggest issue that caused was that you will go and play with random strangers online, have a game and never seen them again making the experience feel really lonely, but GameMate came to change that because now our mobile app gives the chance for players to join our app and match with people they have a similar interest with and can chat with each other in real time making them build stronger connections. GameMate was built using SwiftUI and Firebase; SwiftUI was used to build the whole front-end of the app, in this case the User Interface (UI) of the app and it was also, used to as a medium between the front-end and the back-end extracting all the information from the database, in this case firebase. Firebase was used as the database, and its composed of three different sections: Cloud Firestore which is a NoSQL database, Authentication which was used to authenticate the information was using and make sure it was correct, Storage which was used to stored the images of the users, and Security Rules to limit users in what they can do, such as, not allowing for other users to edit the profile of another user and issues like that one. Our app has different sections to it, it has a login page, registration page where the user will be able to fill out all his information and the games he has interests on, a profile page, match making page, and the chat page.

Barriers, Issues, and Limitations

Some of the barriers and issues that GameMate faced was that it was built on a total new language that the developers were not familiar with in this case SwiftUI and not so much experience with Firebase, but still familiar with it. The framework SwiftUI was never used by any of the developers before, so having to learn this new framework with a limited time made it an issue. The application had more potential that couldn’t be met by the time of the deadline because having to learn it and get familiarized with it in little time. Another issue was time management, having other classes and exams plus been student workers made the process a little hard, but the app got to be done with all the requirements met. Another issue, was learning how to use the security rules in Firebase, the developers struggle two weeks trying to figure out how to upload the pictures to the Firebase, at first it was thought that the issue had to do with SwiftUI and some issues with the logic of the code, but that was not the case at all, the issue was that the security rules were not set correctly in Firebase not allowing for the pictures to be uploaded. Some of the limitations GameMate faced as stated before was time management, there were times the developers were either working, or stuck with another assignment making it a little hard to make progress with the app. Aside from these issues, barriers, and limitation stated before there was not other issues.

Definition of Terms

Framework: Tools or languages use to develop applications or softwares.

SwiftUI: it’s a user interface toolkit to develop mobile applications for the Appstore of apple.

Firebase: It’s a real time firebase that allows to store and zync data at real time.

Cloud FireStore: it’s the NoSQL database that firebase uses.

Authentication: It’s a section of firebase that is where all the information of a user is store to allow to check if the information is correct or not.

Storage: It’s the act of saving information into the database in this case the images were saved into the storage of firebase.

Summary

One of the biggest issue most gamers faced in today’s time is not been able to have someone to play their favorite games with, making the whole experience of gaming feel really lonely and depressing. GameMate was a mobile app built to face this issue many gamers face today, by allowing them to create a profile and see the profile of other gamers where they can see their stats and their common interests. If they are able to match they will be able to chat with each other in real time and be able to get on the game at the same time and play together building stronger relationship and making the experience better and fun. GameMate was built using SwiftUI and Firebase. SwiftUI was used to build the whole front-end of the app, in this case the User Interface (UI) of the app and it was also, used to as a medium between the front-end and the back-end extracting all the information from the database, in this case firebase. Firebase was used as the database, and its composed of three different sections: Cloud Firestore which is a NoSQL database, Authentication which was used to authenticate the information was using and make sure it was correct, Storage which was used to stored the images of the users, and Security Rules to limit users in what they can do, such as, not allowing for other users to edit the profile of another user and issues like that one. Some of the issues faced by GameMate was learning the framework SwiftUI and expanding our experience with Firebase, such as, learning the security rules of Firebase; another limitation was managing the time. GameMate still became to be a great app that still has a lot of potential that will be incorporated into the app later on.

**Chapter II. Review of the Literature**

The background Literature

In this case, the whole idea of GameMate was to create an app where gamers would be able to create a profile and match with other gamers according to their stats and game preferences. The concept of making an app where users can match according to their preferences has a broad and interdisciplinary literature, including fields, such as, computer science, psychology, and social media studies. It includes computer science because without SwiftUI and Firebase the app would have not been able to be built, but also the concepts that were learned in computer science were the ones put into practice when building this app. It also, includes psychology because in this case users will be using their brains and tactics when matching with other users making sure they pick the best teammate for the game they decide to play. They will have to decide whether they want to pick a user to play a game for fun, or to take the game serious and compete against other players. Lastly, social media studies because to some extent this will be a social media app where the users will need to learn how to use it and get the most they can of GameMate.

What is Known and Unknown about the Topic

In this case, what is known about the topic is user profiling and data analysis, as it needs to be used in order to be able to allow for the users to match, also, all the data that is saved from when they create their profiles is saved in the database. Many users might not have played the games that are listed in GameMate, but when the users were writing their stats sometimes they would put their favorite games, since that is data we can analyze from the database it can later on be added into the app, so people can play and match over those favorite games of them. Another thing that is known, is the algorithms used to match the profiles, in this case they were match based on two preferences: the gender and game preference. The user when creating a profile can pick to match with a male, female, or any depending on their preferences, and they can also match according to game preferences, such as, picking Call of Duty or any other game. Once a user picks a game the algorithm will show users that also pick those games.

What is not known, is the long-term engagement users will have with the app or how gamers might interact in the app as they are allowed to chat and say what they desire. Also, people change with their life and some people play for a living, but other ones not so knowing whether users will stay always in touch to play every time is unknown. Privacy and security concerns, because right now GameMate is secure, but is not sure whether when the app is deployed there will be ways to hack the application unless tested farther. Cultural and global differences might also impact the app, as some users might not communicate as another user would like him too or different hours too as when some people might be sleeping is when the other one might be able to connect to play.

**Chapter III. Methodology:**

Methods, Procedures, and Resources Used

In this case the methods and procedures used were the ones of the methodology Agile. This was the best way to tackle GameMate and make it to what it is today. It was the best way to tackle GameMate because it allowed us to know what could be delivered and what it could not be delivered. For example, in the beginning the developers wanted to integrate the APIs of the games that were going to be part of GameMate as it would allow to show real time stats, after emailing the companies for their APIs and not receiving any response it was clear that feature would not be able to be part of GameMate. Doing weekly standups also, help the developers to start on the application early and submit each part of the project each time there was a weekly standup. It was also, a good way to see how the application was coming out and to be able to tell whether the application was good or not. The resources used was google to sometimes come with solutions to the problems we would face and get stuck on, another resource was professor Kumar in giving us ideas on how to make the application better. If Waterfall or any other methodology was used for this project the project would have not been done by the deadline as the developers would have waited last minute to start or other complications might have arise, but been able to deliver new parts of the project each week help a lot, using the methodology of Agile a great decision. The frameworks used for the application was SwiftUI and Firebase. SwiftUI been used for the front-end of the application and firebase used as the back-end of the application.

Reliability and Validity

In this case, GameMate is a really reliable app as it passed every test that was thrown at it, and the application runs smooth with not issues. GameMate did faced a few reliable issues that were fixed throughout the process of building the application. In this case, as stated before that it has validity as it’s a project that met the requirements of the capstone project, but it also has great potential to make it far once deployed, due to the fact, that so many gamers feel lonely when playing games alone, so this will help them find their future partner to play games and have a great time. Also, the algorithms that was built for the matching for storing the data is well studied and established. The frameworks used were also great with a lot of potential to build other types of apps.

Summary

In summary, the methodology that was used to build GameMate was Agile. Agile helped us build this project in a faster and more productive way, allowing us to make all the changes of the project while building it at the same time. Agile pushed the developer to have deliverables that needed to be delivered every Friday of the week. Agile also, ensure that the developers would build a reliable app, as it needed to pass every test that was tested on the application, one of the issues the developers noticed is that a user could edit the profile of another user, but that was tested later on and fixed and the issue never happened again as it was fixed. GameMate does have validity as it met the requirements of capstone project, but it can also change parts of a gamer’s life as they won’t have to play alone no more and will be able to meet new friends to play with and build strong connections. The resources used was SwiftUI, Firebase, Google, and ideas suggested by professor Kumar.

**Chapter IV. Results**

Data Analysis, Findings, Summary

In this case, the results showed that application was good and exceeded the expectations of many people including the developers. The application worked when registering users, it saved their information that would later be used to match their information with other users. Most of the users that used the app ended enjoying as it has a friendly non-complicated user interface. The algorithm used to match the preferences worked, as every time a user registered only users with the same preferences would show up in the match view. The algorithm used to match the users was based on the gender that they were interested in and the game preferences. It is expected for this type of application to have a community engagement and growth as many gamers play alone and this will change their experience for the best. GameMate it’s also going to keep on growing and improving as the developers are planning to incorporate for users to be able to link their social medias or discords in case they rather text in those other places, add another section where users will be able to upload videos or photos of a great play they do in the game. For example, a user is playing Call of Duty and they get 10 kills in a row they will be able to clip that play and upload it to that section, so the users will be able to see that and catch their interest in that play. Overall the app came with great results exceeding the expectations of everyone.

**Chapter V. Conclusion**

Implications and Recommendations

In conclusion, many gamers now days play all alone in their consoles or PCs making the experience feel really lonely and depressing and not getting their best experience out of their favorite games. GameMate was built to fight this issue that many gamers face every time they get on their favorite games. GameMate was built using SwiftUI and Firebase; SwiftUI was used to build the whole front-end of the app, in this case the User Interface (UI) of the app and it was also, used to as a medium between the front-end and the back-end extracting all the information from the database, in this case firebase. Firebase was used as the database, and its composed of three different sections: Cloud Firestore which is a NoSQL database, Authentication which was used to authenticate the information was using and make sure it was correct, Storage which was used to stored the images of the users, and Security Rules to limit users in what they can do, such as, not allowing for other users to edit the profile of another user and issues like that one. Two implications were faced when building this project one was learning and getting experience with SwiftUI and Firebase, the second implication was the time management. One of the implications that came with learning SwiftUI and getting experience with Firebase was uploading the pictures of the users when they were registering, and been able to edit another’s user profile from a different account. The recommendations that will be added for the future is for users to be able to link their social medias or discords in case they rather text in those other places, add another section where users will be able to upload videos or photos of a great play they do in the game. For example, a user is playing Call of Duty and they get 10 kills in a row they will be able to clip that play and upload it to that section, so the users will be able to see that and catch their interest in that play. Also, to add more games as of right now there is only the 7 most popular games out there.

**Back Matter**

References

Freddolino, P. P., & Blaschke, C. M. (2014). Therapeutic applications of online gaming. In *Internet-Delivered Therapeutic Interventions in Human Services* (pp. 316-339). Routledge.

Douglas, S., Tanin, E., Harwood, A., & Karunasekera, S. (2005, December). Enabling massively multi-player online gaming applications on a p2p architecture. In *Proceedings of the IEEE international conference on information and automation* (pp. 7-12).

Barker, C. (2020). *Learn SwiftUI: An introductory guide to creating intuitive cross-platform user interfaces using Swift 5*. Packt Publishing Ltd.

Varma, J., & Varma, J. (2019). What Is SwiftUI. *SwiftUI for Absolute Beginners: Program Controls and Views for iPhone, iPad, and Mac Apps*, 1-8.

Moroney, L., & Moroney, L. (2017). An Introduction to Firebase. *The Definitive Guide to Firebase: Build Android Apps on Google's Mobile Platform*, 1-24.

SwiftUI, L., by Building, F., & De Carli, S. Build Mobile Apps with SwiftUI and Firebase.

Venzano Rodriguez, E. (2022). Web And Mobile Application Portfolio.

Appendixes

A1. ChatViewModelA screen shot of a computer

Description automatically generated

A.2 MatchViewModelA screenshot of a computer

Description automatically generated

A.3 SwipeViewModelA screenshot of a computer

Description automatically generated