DALHOUSIE UNIVERSITY

CSCI 5708: MOBILE COMPUTING

PROJECT REPORT

CONTENT

1. Title Page

The title we decided to go with for our product was ‘DalSocial’. We chose this particular name for the target audience we were pushing our application into. As the name suggests, the target audience was clearly the students and professors of Dalhousie University. The fairly good contribution of this report was done by the following members of the group:

Name	Banner ID
Siddharth Kharwar	B00897211
Hrishi Patel	B00911902
Parvish Gajjar	B00912090
Faiza Umatiya	B00899642
Shravya Reddy Gennepally	B00911193

2. Product

Our group aimed on building a social media application that facilitates various means of socializing with other like-minded people through features like Social, Events etc. The users of this application would be able to connect to other people who share similar interests. Our application differs from other applications as we focus on matching people who have a similar sense of perspectives towards the world. The list of functionalities that were initially planned and the functionality that was actually implemented is mentioned as below:

Feature Category	Feature Name	Completed
Minimum Functionality	Basic User Management	Yes
	Events Module	Yes
	Social	Yes
Expected Functionality	Reset Password	Yes
	Searching in Events	Yes
	Notifications	No
Bonus Functionality	QR Code Share	Yes
	Payment Gateway	No
	Group	No

The app performs well with some minor bugs. The User Management Module works well with no bugs found till date. The Social feature has a minor bug where already matched user might pop up for a user in certain cases. The Event module works well with a minor bug that delays the UI to update according to the data that is fetched.

We feel there were 2 features in the application that was something special for us. The Card Swiping to meet new people was something that was challenging for us as we all were new to Android XML designing. Other than that, the QR Code friending module was quite special as it allowed user to directly add user with the help of a QR Code with little to no time.

There could have been improvements to the application if the time permitted. We feel we could have improved the UI a little with colors that were more attractive to the user. We

decided to go with the default Material 3 colors without testing new color palette. Apart from that there were areas where we could have handled error more efficiently.
The product we decided to build got a little ambitious as one of the team members dropped the course and we could not implement all of the features that were assigned to him. Moreover, we encountered certain errors that delayed the development as we all were new to Kotlin programming and there were not many resources available pertaining to those errors.

3. Lessons

Some team members were new to Kotlin and Android Development. Some already had made Android apps before, but without proper implementation of design patterns or MVC pattern. Firebase was also new to many of us.

One of the biggest lessons is about teamwork. Start early, don't over commit. Make sure every team member is contributing equally from the start and check with team members regularly. Not all team members are comfortable to discuss issues with other team members.

A continuous integration pipeline would have been helpful along with test driven development. We had issues frequently where someone would merge their changes without testing and someone else's code will break.

4. Issues

Asynchronous Programming: Not necessarily an issue, but it was bit confusing at first for some of the team members. It's a bit different with Kotlin. It isn't as simple as using keywords like async and await as we did it in other programming languages like Java. We mostly relied on callback functions to achieve asynchronous programming.

Social Feature's Matchmaking logic: This feature matches people (like Tinder swiping feature) based on their interests. The logic was not very easy to build and required quite some time to build.

Android Studio and Setup issues: We all know Android Studio is a memory hog. Some of its features frequently stopped working. Features like code autocomplete for UI design (XML) did not work with the latest version of Kotlin Core.

5. Satisfaction

Overall the teams been satisfied with how our application turned out to be, the knowledge we gathered throughout the process of building this project as a team has been nothing sort of phenomenal which each of us would be remembering through our lives and this experience also showed us how a language is not a barrier to building any new project and developer could easily learn new languages/framework and build something beautiful.

While each of us worked on our distributed feature of the application, we kept integrating with the help of 'git' which really gave us the real time idea of how the application was shaping/turning out to be. Through this methodology each of us were able to check in each other's feature and see if anyone's stuck anywhere and needs help to fast-track the process of learning and development.

6. Task Distribution

Throughout the development of the project, we have adopted a strategy to provide each person with the equal work distribution, be it building the new features, documenting, or the presentation. The five features core components of our application were distributed among five people. While, the more experienced folks in Kotlin took the lead with developing the code base for other to start working on their respective feature.

Hrishi Patel – User Management, Profile/Settings, QR Code (Additional Functionality), Social
Siddharth Kharwar – Events Feature
Parvish Gajjar – Chat Messaging Feature
Faiza – Social
Shravya – Group MeetUps Feature

Each of us had fair share of learning while developing this feature as the students who were new had to understand the core functionality of building the UI Frames i.e. the .xml files, interaction with the database, and the finally controller where the actions take place when user interacts with the system.

7. Conclusion

To conclude, it's always going to feel amazing to look back at this term with what we as a team were able to achieve in such small amount of time. In the last two weeks of the developing process we were really happy with how the team was working efficiently, developing their own feature while also helping each other. Learnings from this project while implementing the features, QR Code Sharing, Real-time messaging will help us in our careers to further go on to build something even better and wonderful.

8. REFERENCES

[1] Android. 2022. What is Android? | Android. [online] Available at: <https://www.android.com/intl/en_ca/what-is-android/> [Accessed 9 October 2022].

[2] Firebase. 2022. Firebase. [online] Available at: <https://firebase.google.com/> [Accessed 9 October 2022].

[3] Figma. 2022. Figma: the collaborative interface design tool.. [online] Available at: <https://www.figma.com/> [Accessed 9 October 2022].

[4] draw.io. 2022. draw.io – Diagrams for Confluence and Jira - draw.io. [online] Available at: <https://drawio-app.com/> [Accessed 9 October 2022].

[5] “Balsamiq. Rapid, effective and fun wireframing software,” *Balsamiq.com*. [Online]. Available: https://balsamiq.com/. [Accessed: 31-Oct-2022].

[6] “Meetup - We are what we do,” *Meetup*. [Online]. Available: https://www.meetup.com/. [Accessed: 09-Oct-2022].