DUBLIN INSTITUTE of TECHNOLOGY
Institiúid Teicneolaíochta Bhaile Átha Cliath

SessionOn
Final Year Project Report

DT282
BSc in Computer Science

Céin O'Rourke
Oisin Creanor

School of Computing
Dublin Institute of Technology

04/04/18

School of Computing
Scoil na Ríomhaireachta

Abstract

The objective of this project is to connect friends on a social level, directed more towards the weekend, when friends would be looking to go out and meet up with each other after a week of college or work etc. The idea of it being a mobile app that will enable a user to see their friend's location and then be able to see what they are doing and where. The application also calculates how far away friends are from one another and updates each time the user clicks on the name of the user they wish to see. The user simply signs in with email and password and they are registered on the firebase database and saved. This uses OAuth2 and aids the app in authentication and security.

The application will be designed and implemented for the android platform written in Java. The layout will be written in XML. The app will also include google maps activity in co-ordination with firebase using API key. The app will run on an android device and connect to firebase using either WI-FI or mobile data. The app will run more smoothly on firebase as it syncs all updates across all devices running the app. The main idea behind a user the app would be for example when the user goes out to his local pub for example and checks in saying what is on. Then using location, geo mapping and boundaries set by the user their friends will get a notification and be able to look where they are and see who is with them also, what event is happening. They can then choose to join the session or look in their radius to see who else is out closer to them. The app will include a messenger for quick conversation. The app will continuously process data to give live updates, so if people move etc. One's account can be public or private as they can choose. I will have it connected to an API that connects to a cloud to save locations and accounts. The problem I want to eradicate is the whole group chat scenario when someone tries to organise something, and it just falls through. Hopefully the outcome of this project will be a full up and running app that people will use to organise nights out between groups of friends or random parties.

Declaration

I hereby declare that the work described in this dissertation is, except where otherwise stated, entirely my own work and has not been submitted as an exercise for a degree at this or any other university.

Signed:

_____

Céin O'Rourke

04/04/18

Acknowledgements

I would like to thank my project supervisor Oisin Creaner for his guidance and support throughout the project.

I would also like to thank Jon McCarthy for his aid throughout the year and his guidance.

Table of Contents

1. Research

a. Background Research

This part of the report contains the background research conducted before any implementation or any design of the project. Before making any head way on SessionOn. Other projects existed were researched and learned to progress with the

project and see what other approaches were taken etc. There are a few that came up that used the same kind of technologies, SessionOn would thrive with if incorporated. As location is a big thing nowadays and many social apps incorporate it into their applications and the development of SessionOn will benefit from the research of these other applications.

As mentioned before there are multiple apps now that are using location in their projects and then there are others that are based solely on location. There are a few that used different technologies that needed to exist in the project. For example, Snapchat, Whatsapp, Tinder, Facebook, Glympse, Life360.

Snapchat

Snapchat is a big name now in the social network community and released a new feature called snapmap in June 2017. “Snapchat appeared to have copied social map startup Zenly for this feature, but now we’ve learned that Snap Inc has acquired Zenly for $250 million to $350 million in mostly cash and some stock. Snap will keep Zenly running independently, similar to how Facebook runs Instagram independently.” [1].
The feature lets you see where all your friends are and what they are doing by them sending their photos/videos to their public story. This feature is something SessionOn could use to benefit it. There are a lot of pros and cons to this feature of course.
“When it comes to Snap Map, young people aren’t scared about stalkers or strangers – their biggest worry is each other. The map can jeopardise relationships in a number of ways. People can see when their friends are hanging out without them, they can tell when someone has lied about setting off but is still at home, and – when checking at night – they can figure out who’s sleeping with whom. The feature even allows you to see the last time someone sent a Snap, meaning you can tell if they’ve been ignoring your messages.” [2]

Figure 1: Tweets under comment in snippet [2]

When asked some possible users about this, they said that they would prefer to have the option whether to broadcast their location publicly or with friends. The snapmap architecture is slickly designed as to get into it a user only has to open the app and then just pinch the screen to access it.

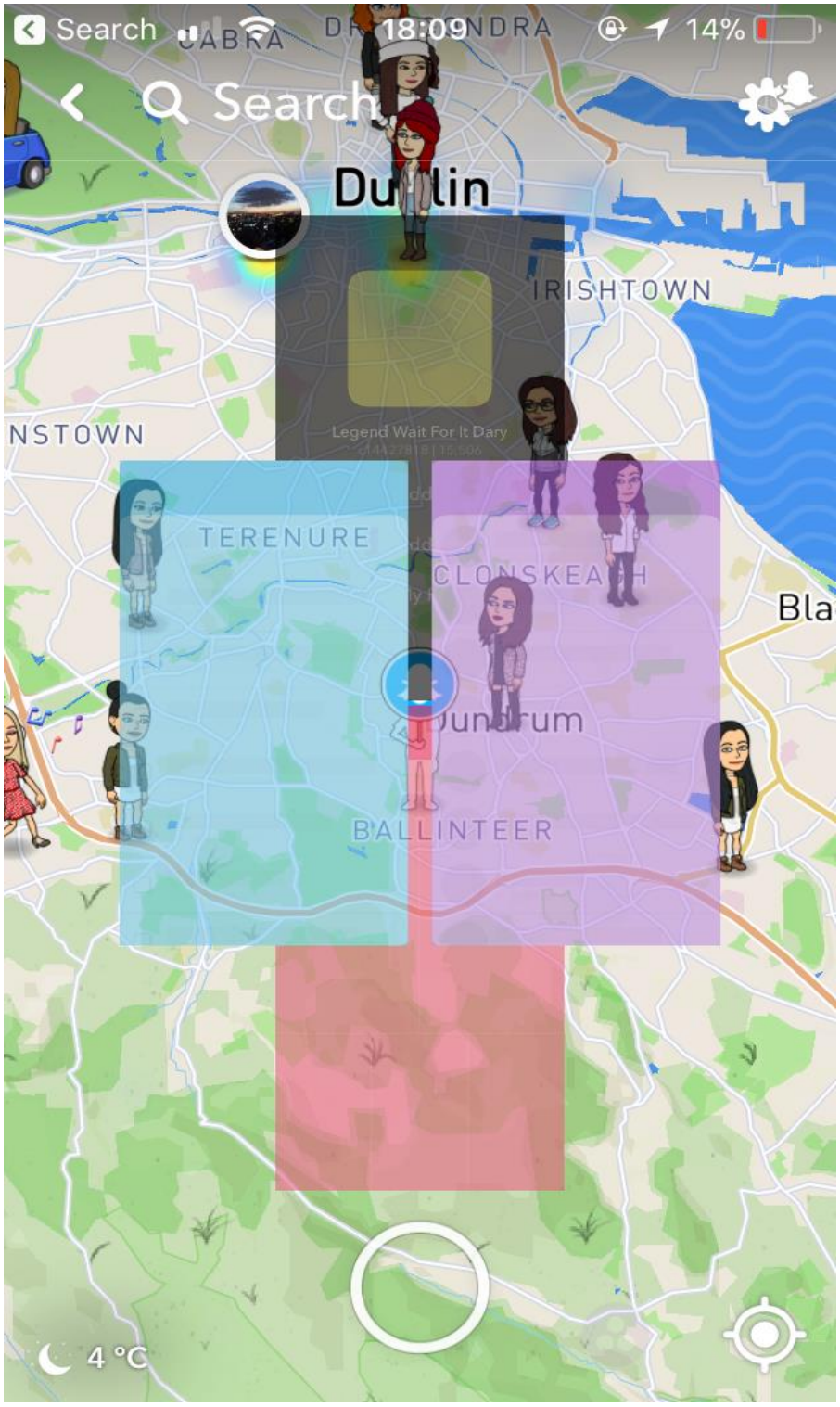

Figure 2: Snapmaps Access Screenshot

Tinder

Tinder is also a huge social dating app now which incorporates geo mapping and geo boundaries into its application.” The 2016 Editor's **Top Pick - Dating App** Award goes

to **Tinder**. 2016 was another big year for Tinder being the most popular dating app on iOS and Android. Its estimated value was pegged at $1.2 billion USD in 2016 and it has now more than 1.5 million paid subscribers" [3].

This is one of the features the project would flourish with when to included. And in Tinder a user can choose how far the radius is to match with other singles within that radius.

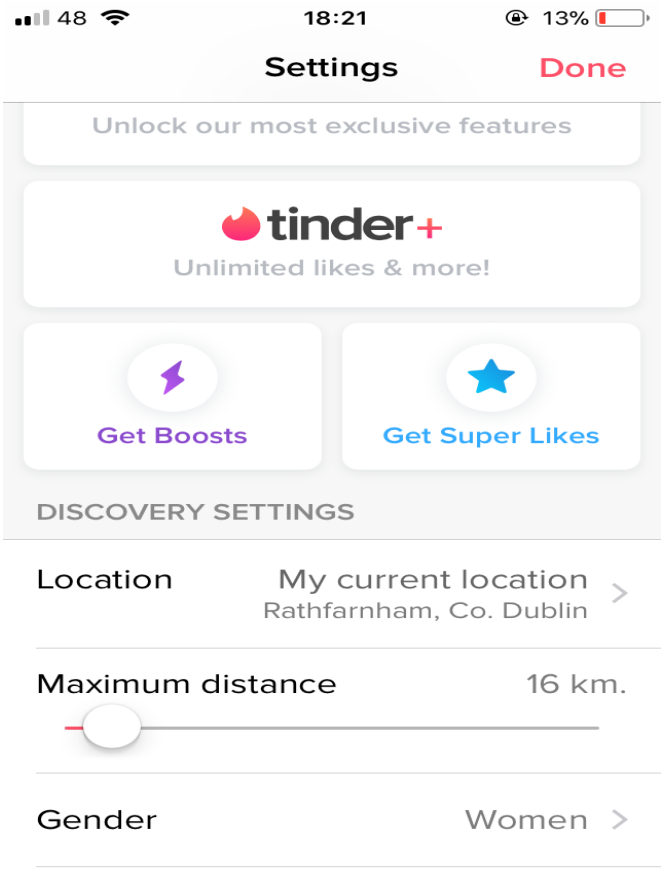

Figure 3: Tinder settings screenshot

Android has this built in using the Google API and has functions that can be used to inherit the ideas of this geo mapping and geo fencing.

int	GEOFENCE_TRANSITION_DWELL	The transition type indicating that the user enters and dwells in geofences for a given period of time.
int	GEOFENCE_TRANSITION_ENTER	The transition type indicating that the user enters the geofence(s).
int	GEOFENCE_TRANSITION_EXIT	The transition type indicating that the user exits the geofence(s).
long	NEVER_EXPIRE	Expiration value that indicates the geofence should never expire.

Figure 4: Functions screenshot [4]

b. Technologies Researched

Google services API

Google developer provides a lot of useful functions and a location API that you can go to developer page and get your personal API key and then include it into your code. This allows one to create a location tracking functionality into ones app. One can also use the features of Google maps which will allow one to include functionality that is included e.g. route planning, network provider, GPS provider, Google places.

Firebase

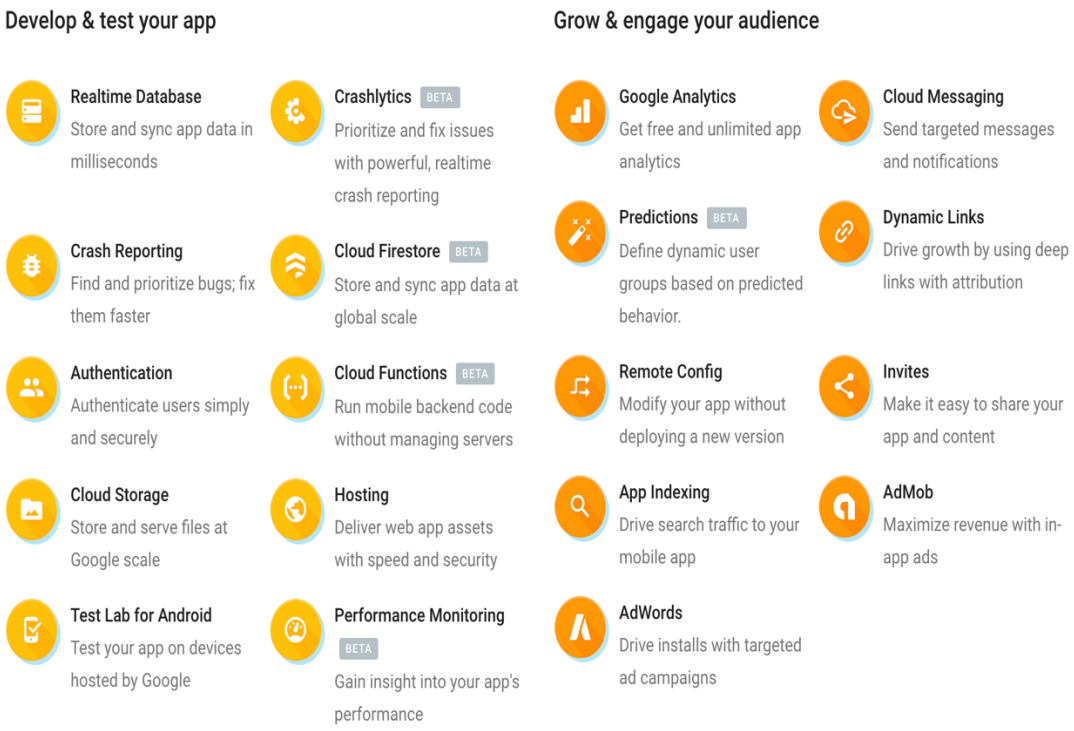

Figure 5: Abilities of Firebase [5]

Firebase is a backend service and incorporates a lot of features as shown above and if the project had said features included it will hopefully boost its efficiency and its performance.

“Firebase frees developers to focus crafting fantastic user experiences. You don’t need to manage servers. You don’t need to write APIs. Firebase is your server, your API and your datastore, all written so generically that you can modify it to suit most needs. Yeah, you’ll occasionally need to use other bits of the Google Cloud for your advanced applications. Firebase can’t be everything to everybody. But it gets pretty close.” [6]

Real-Time database: “When you connect your app to Firebase,

you’re not connecting through normal HTTP. You’re connecting through a WebSocket. WebSockets are much, much faster than HTTP. You don’t have to make individual WebSocket calls, because one socket connection is plenty. All of your data

syncs automagically through that single WebSocket as fast as your client's network can carry it.
Firebase sends you new data as soon as it's updated. When your client saves a change to the data, all connected clients receive the updated data almost instantly." [6]

If the project was to incorporate this feature it would better the real-time updates for the regular location update and data processing. Also, the project would run smoother if more users intend to use it. It will also aid the project as it will keep all clients connected and all updates would be instant.

File Storage: "Firebase Storage has its own system of security rules to protect your GCloud bucket from the masses, while granting detailed write privileges to your authenticated clients." [6]

Authentication: "Firebase auth has a built-in email/password authentication system. It also supports OAuth2 for Google, Facebook, Twitter and GitHub. We'll focus on email/password authentication for the most part. Firebase's OAuth2 system is well-documented and mostly copy/paste.
If you've ever written an authentication system, let's commiserate for a moment. Custom authentication is terrible. I will never write an auth system again for as long as I live. I fell in love with Firebase Auth at first sight, and the flame has never wavered. Sometimes I get frustrated. Sometimes we fight. But I never forget the cold, dark abyss of a custom auth system. I count my blessings.
Oh, and Firebase Auth integrates directly into Firebase Database, so you can use it to control access to your data. I'm writing this as if it's an afterthought. It's not. It's the second reason that you will love Firebase Auth." [6]

This feature will aid the project and reduce the number of bugs at runtime, rather than being coded from scratch. Also, if the project wants to have more available features like sign in with the users Facebook or Github etc. it can be added to the project quite quickly and easily from the developer page of the firebase website.

App platform features: “The Firebase team has integrated a bunch of new and existing Google products with Firebase. I don’t plan to cover these features in detail quite yet…

A bunch of these features apply to iOS and Android but not to web.

- Remote Config
- Test Lab
- Crash
- Notifications
- Dynamic Links
- AdMob

“[6]

```
implementation 'com.google.firebase:firebase-core:12.0.1'
implementation 'com.google.firebase:firebase-auth:12.0.1'
implementation 'com.firebaseui:firebase-ui-auth:1.2.0'
implementation 'com.firebaseui:firebase-ui-database:1.2.0'
implementation 'com.android.support:appcompat-v7:26.1.0'
implementation 'com.google.firebase:firebase-database:12.0.1'
```

Figure 6: Adding firebase screenshot

Overall the project will benefit greatly in its security and also the effectiveness of the app like data processing (for updating user’s location). As seen in these snippets there is a lot that the project can benefit from by including Firebase into it and greatly improve its functionality.

c. Other Relevant Research Done

Life360

Life360 is a social network application that is built more about family life the idea of meeting up with family for things such as dinner or just checking in on one another. It uses the gps tracking and also messaging features that SessionOn would thrive with.

“With Life360’s geo-fencing feature, called “Places”, users can now designate specific places to receive alerts for. Whether it’s a school, mall, office, or home, now people can quickly be informed when someone has arrived at that destination.” [7]. This also uses the idea of family members checking in and letting them know they are home safe or at work etc.

d. Resultant Findings and Requirements

- Database
- GPS tracking
- Geo fencing
- Geo mapping
- Google API/Cloud

e. Bibliography

[1] available from: https://techcrunch.com/2017/06/21/snap-map/ Author: Josh Constine Accessed: 29/11/17

[2] available from: https://www.newstatesman.com/science-tech/social-media/2017/07/what-is-snapchat-snap-map Author: Amelia Tait Accessed: 16/09/17

[3] available from: https://www.datingsitesreviews.com/article.php?story=2016-dating-sites-reviews-choice-awards---dating-app Author: Admin Tom Accessed: 29/11/17

[4] available from: https://developers.google.com/android/reference/com/google/android/gms/location/Geofence Author: Google Accessed: 21/11/17

[5] available from: https://firebase.google.com/ Author: Google Accessed 21/11/17

[6] available from: https://howtofirebase.com/what-is-firebase-fcb8614ba442 Author: Chris Esplin accessed: 21/11/17

[7] available from: https://thenextweb.com/insider/2012/12/18/life360-to-hit-25-million-users-adds-geo-fencing-feature/ Accessed: 26/11/17 Author: Ken Yeung

2. Description of Solution

The solution unfortunately is not yet a fully functioning app where a user can create an account using email and password, then sign in with their account. Then show their location using Google maps and a create session button at the bottom of the screen as set out before. The user is able to see where their friends are or others who broadcast their session publicly. The application gives a list of people who are online on the session and can choose to join and broadcast their location to the other users. Rather than creating a session, the app shows where the user is sessioning and how far away he or she is. The user can then choose to join this list of sessioners and choose to meet up with them or stay put and see if anyone joins them where they are. The user's location is continually updated, so if the user moves it will be updated in the app using data processing. Unfortunately, no chat function so users can chat within the app I had trouble trying to add this function with the different versions of firebase and google play services. This is disappointing because a huge percentage of messages between friends nowadays is the, where are you? Question. "Our research shows that 6 to 8 text messages a day are dedicated to this question. Why not just open the Life360 map and answer it for yourself?" [1]. SessionOn would definitely flourish with this feature and could be an amazing social app. In the future I believe this can be achieved as SessionOn has sorted all versioning problems between Firebase and google play services.

3. Approach and Methodology

Iterative Design Methodology

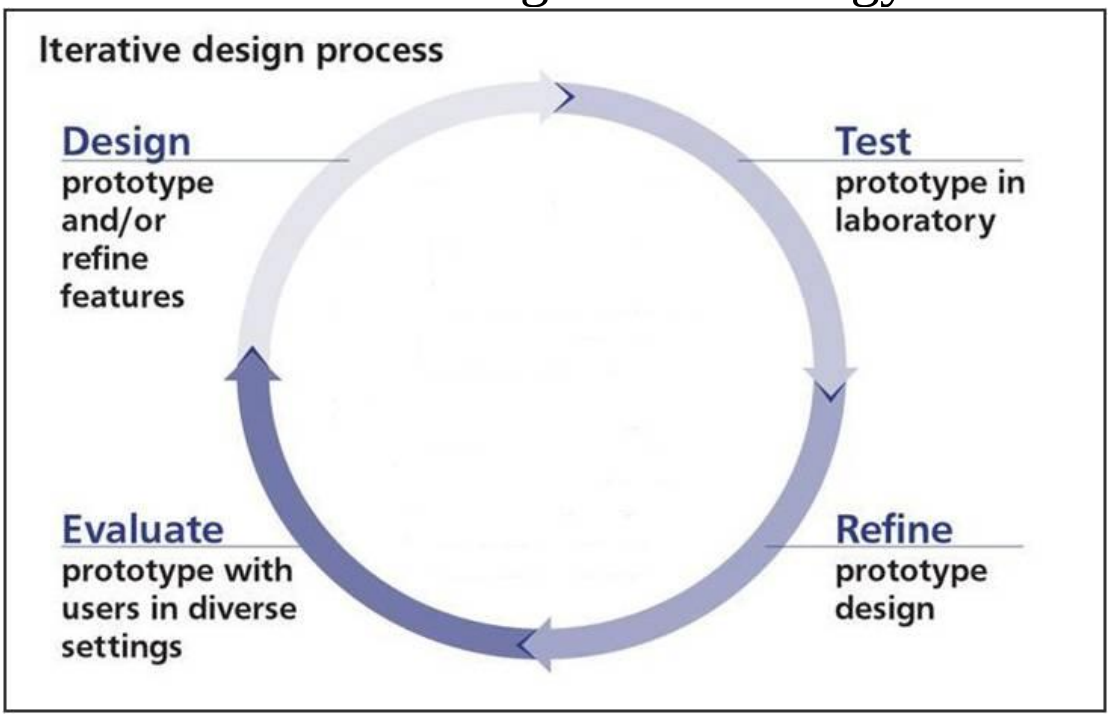

Figure 7: Iterative design[2]

This design methodology is the idea of going around the requirements as shown in the figure above multiple times design, prototyping/coding, testing, refining, evaluating. On completing one segment of the app then using that as the basis for the design of the next segment. Alistair Cockburn says "**Iterative** fundamentally means **re-do**. Iterative development helps you improve your **product."** [3]. If SessionOn uses this methodology it will aid it to be a better app for users. As there is only one person developing I feel that this methodology is the only way that the development if the software will be completed successfully and efficiently.

4. Design

a. Technical Architecture Diagram

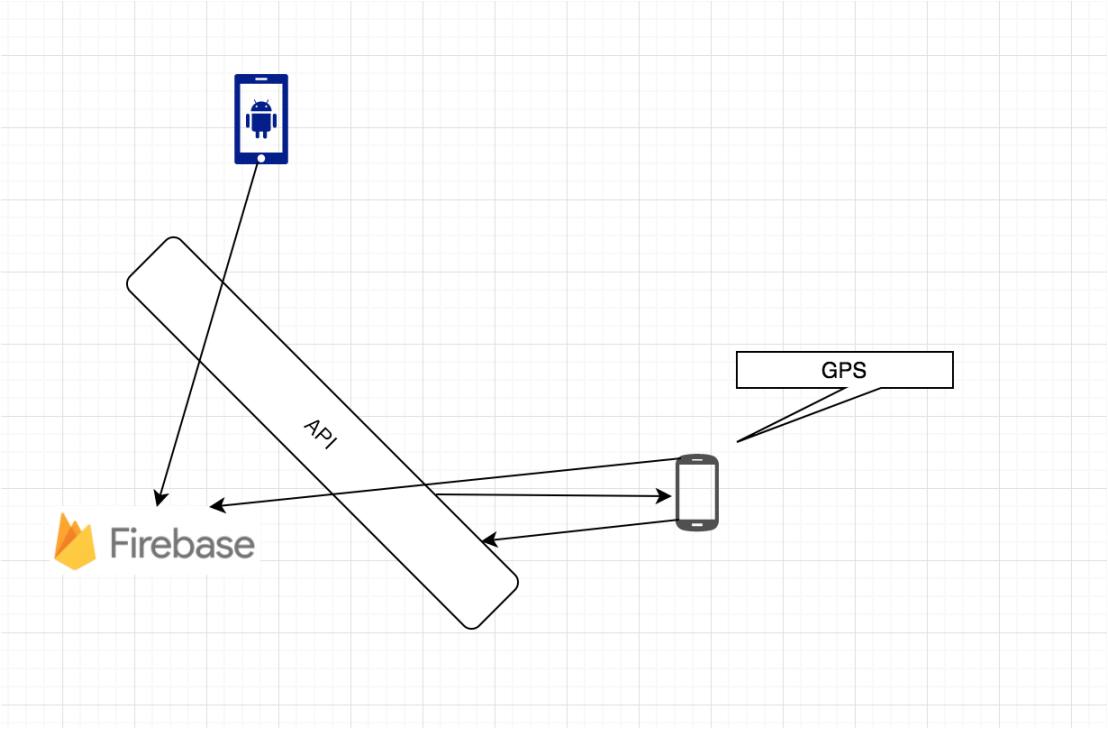

Figure 8: Components

As shown above the app connects to firebase through an API and then the GPS co-ordinates are sent to the database to track the user's location and connects to the API to continuously process and broadcast the live location of the user.

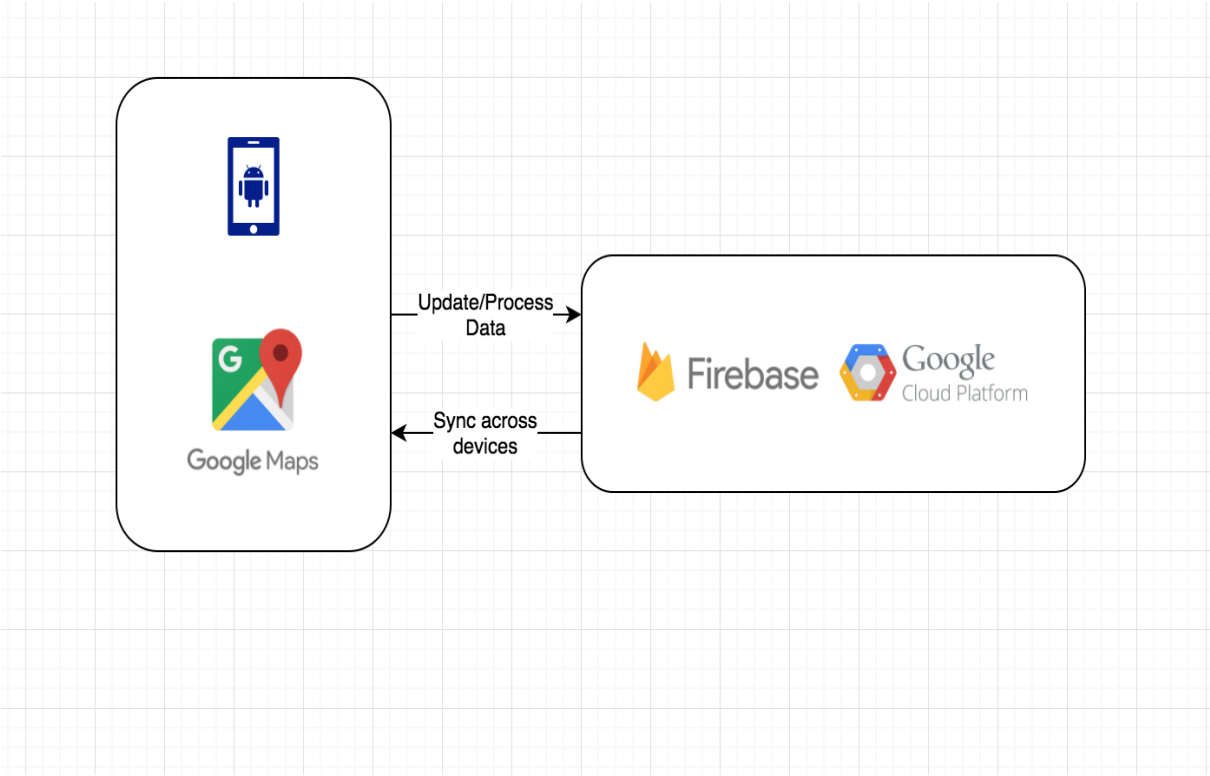

Figure 9: Technical architecture

As shown above the presentation layer has the app with google map activities and it connects to firebase and Google's cloud platform. The location and if added the cloud messaging that can be continually updated and then all changes will be synced across all devices that want to run the app. For example, if a user moves then their location will be updated and other users on other devices will see this updated.

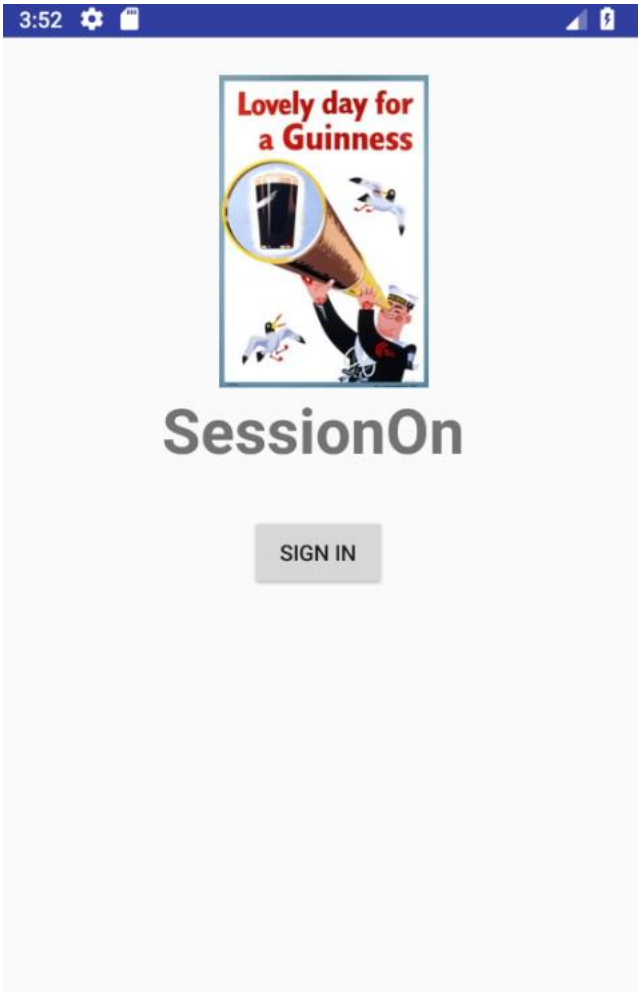

Figure 10: Sign in

The app has a simple sign in method using email and password that are saved on the database and remembered by the app. By using Firebase, the app definitely benefits from a security aspect as on can securely change their email or password if forgotten. The sign in method takes the user to different pages when entering email and password as shown in the figures below.

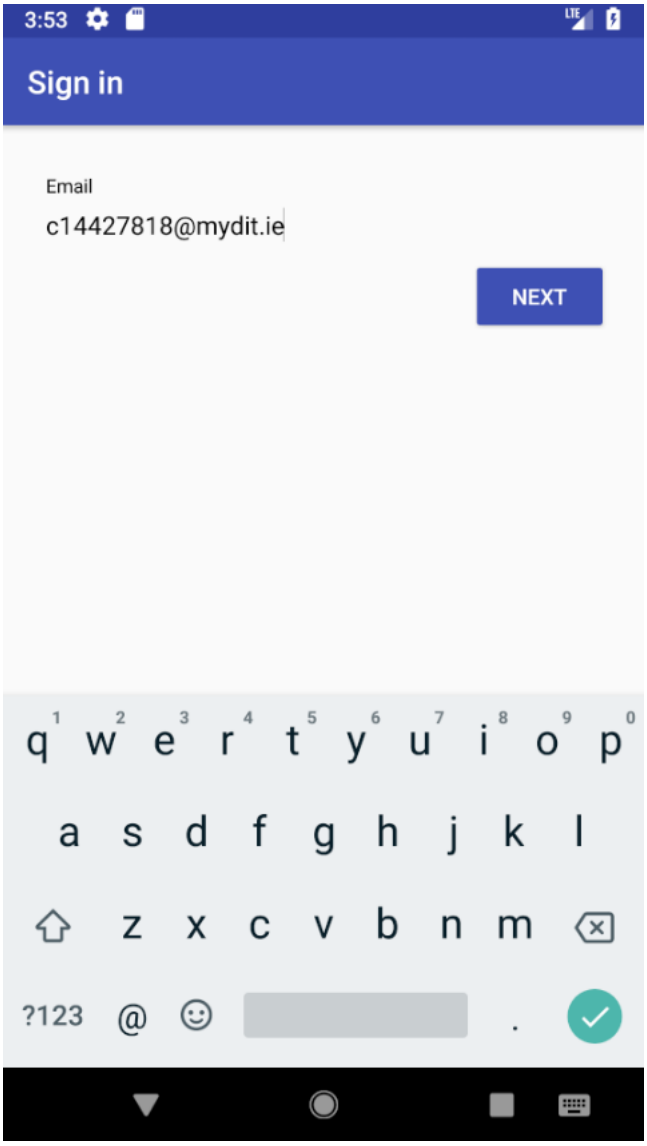

Figure 11: Email

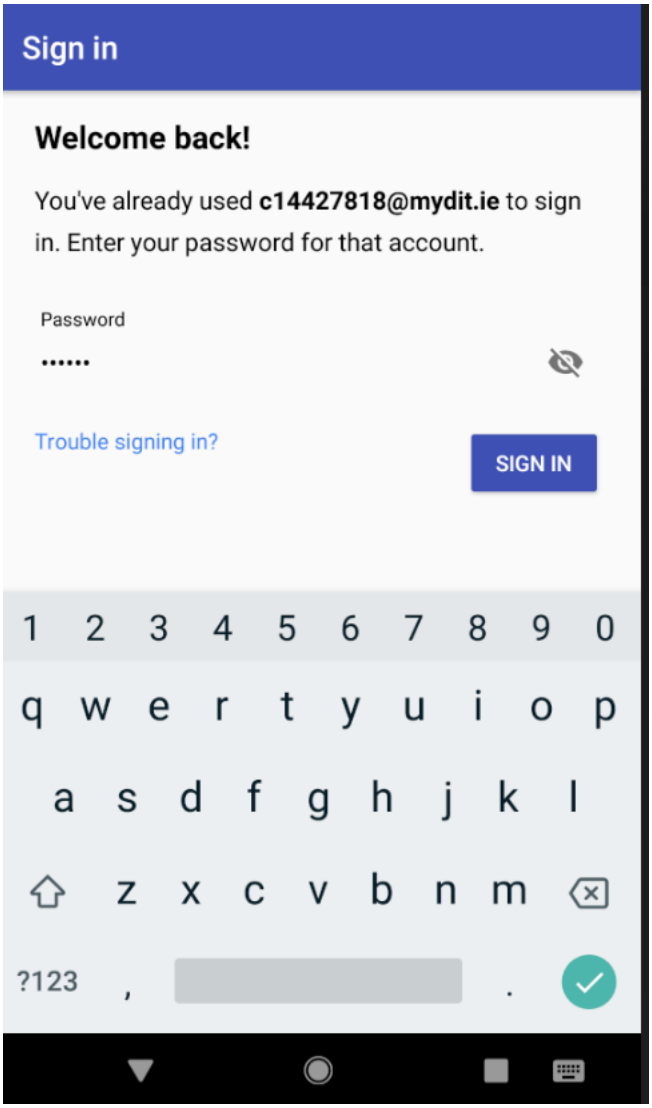

Figure 12: Password

As seen here the app remembers emails that have been used previously and are saved to the database.

b. Other Design Documents

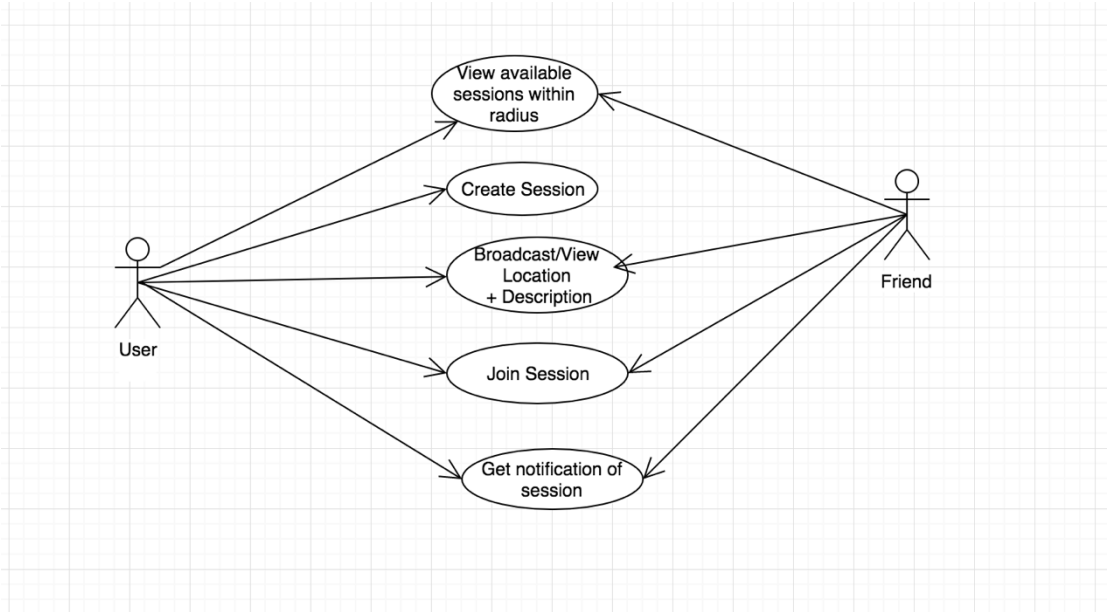

Figure 13: Use Case

Above is the Use Case diagram portraying some of the main features available to each user. The user is the one who creates a session be it a night out or a simple meet up with friends. Both the user and friends can see a location and description of what is taking place and who is there. When a session is created both will get a notification if they are in the boundaries of said person. The user location will be tracked and can create a new session if they have moved elsewhere and a new notification will be triggered.

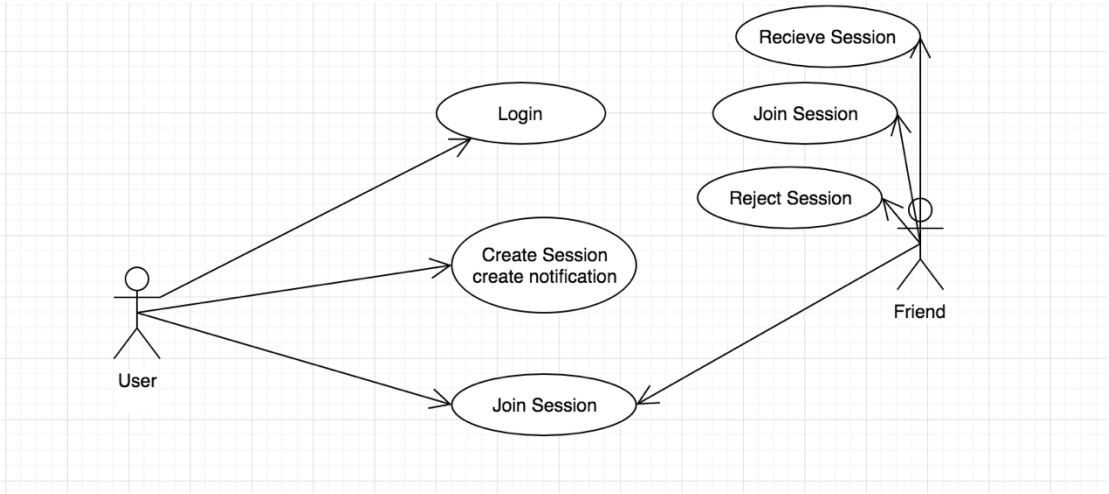

Figure 14: Use Case

Above shows the Use Case for the notification system and how a user can quickly create a session and notifications are automatically sent and location broadcasted. A

user can choose to accept the session invitation and join the group or simply reject it if they do not wish to go out.

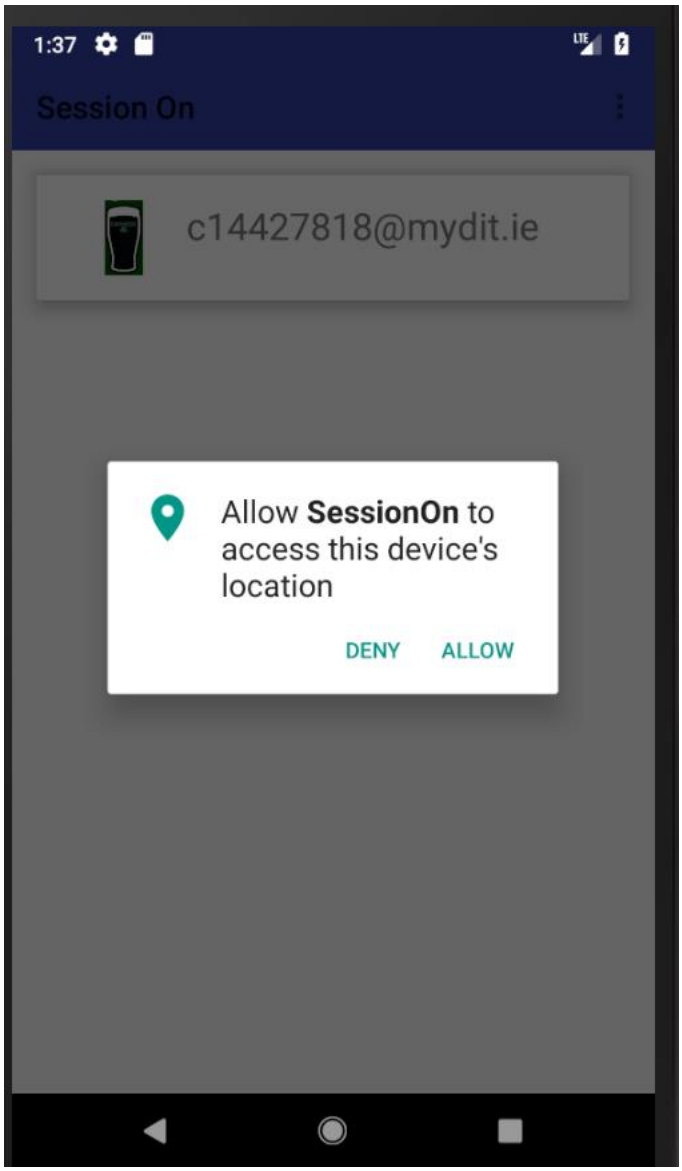

Figure 15: SessionOn screenshot

In the figure above a user can see who is online and allow the app to access the user's location. The User can then see other people who are online and by clicking their name can see where they are on the map and how far away they are from them.

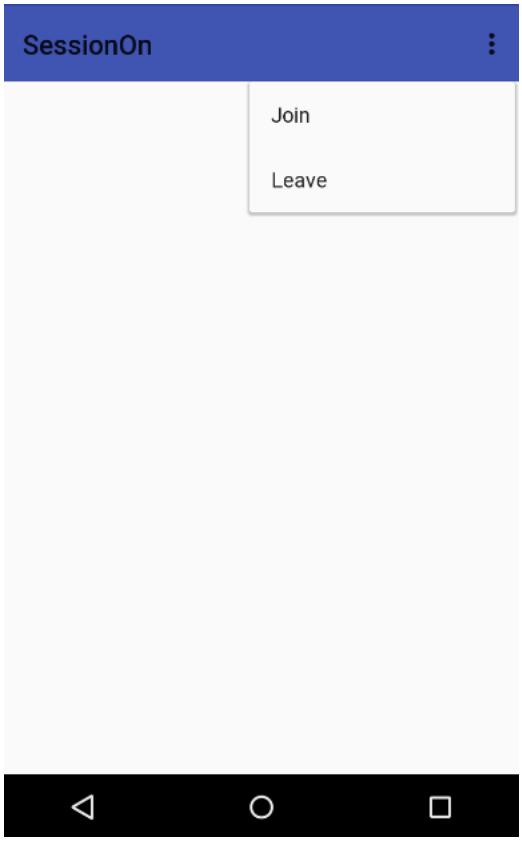

Figure 16: Join/Leave screenshot

As shown above this is the join or leave function. When a user signs in they join this kind of feed to see who else is online and can then simply click on the user and see where they are and then find out what they are doing. They can then leave this wall if they are not doing anything but can still see who else is online and see where they are.

```
            viewHolder.itemClickListener = new ItemClickListener() {
                @Override
                public void onClick(View view, int position) {
                    if (!model.getEmail().equals(FirebaseAuth.getInstance().getCurrentUser().getEmail())){

                        Intent map = new Intent( packageContext: UserOnline.this,SessionTracking.class);
                        map.putExtra( name: "email",model.getEmail());
                        map.putExtra( name: "lat",mLastLocation.getLatitude());
                        map.putExtra( name: "lng",mLastLocation.getLongitude());
                        startActivity(map);
                    }
                }
            };

        }
    };
    adapter.notifyDataSetChanged();
    UserOnline.setAdapter(adapter);
}
```

Figure 17: Code screenshot

As shown above and previously mentioned this is how SessionOn shows the location of the user that is clicked on. This enables the user to pick and choose which person he wants to see what is up to. SessionOn benefits from this in the sense that if the person only wants to go out with a certain person they have the means to do so.

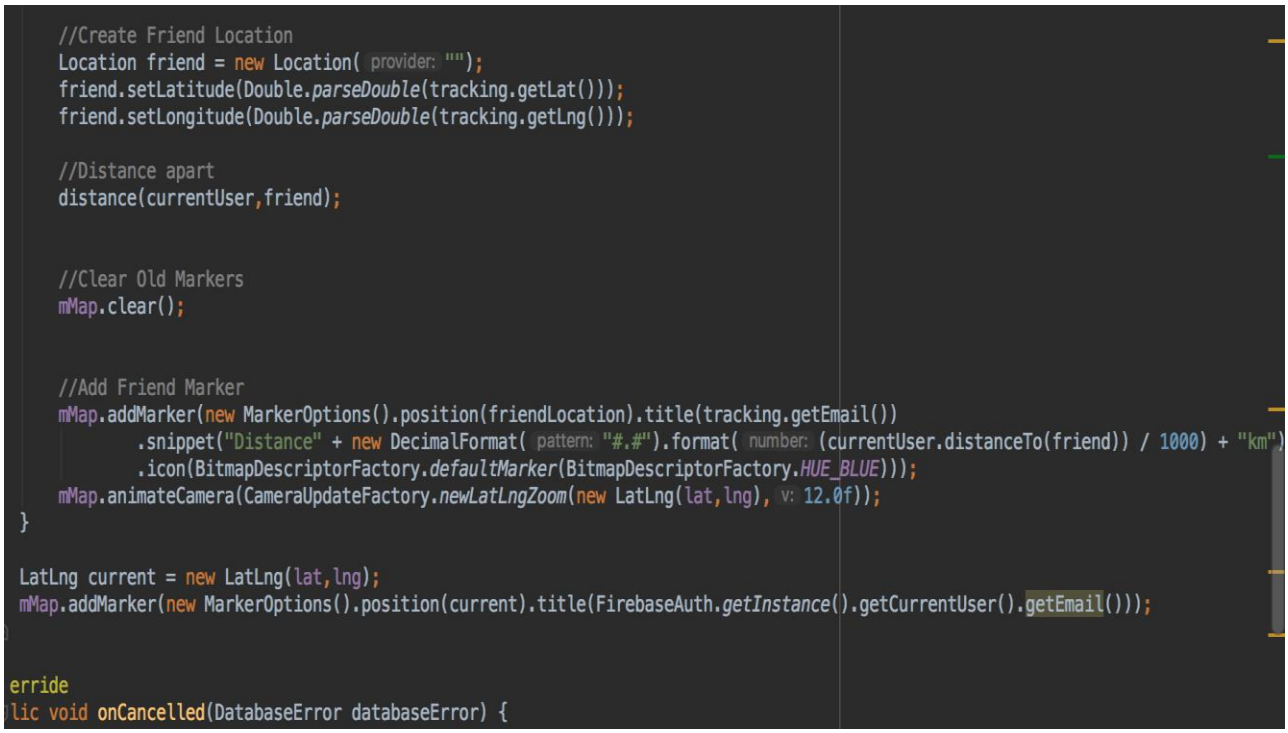

Figure 18: Code screenshot

In this code screenshot, you can see that this is where SessionOn creates the friend or friends' location and then calculates hoe far the user is away from said friend or friends.

```
private double distance(Location currentUser, Location friend) {

    double theta = currentUser.getLongitude() - friend.getLongitude();
    double dist = Math.sin(deg2rad(currentUser.getLatitude()))
            *Math.sin(deg2rad(friend.getLatitude()))
            *Math.cos(deg2rad(currentUser.getLatitude()))
            *Math.cos(deg2rad(friend.getLatitude()))
            *Math.cos(deg2rad(theta));

    dist = Math.acos(dist);
    dist = rad2deg(dist);
    dist = dist * 60 * 1.1515;
    return (dist);


}

private double rad2deg(double rad) {
    return (rad * 180 / Math.PI);
}

private double deg2rad(double deg) {
    return (deg * Math.PI / 180.0);
}
```

Figure 19: Code Screenshot

As shown above. In this code snippet. This is the algorithm used by SessionOn to calculate how far away the user is from his or her friends. It is them passed up the function above to give the distance in km.

5. Prototyping and Development

The prototyping and development I have completed is incorporating firebase into my application for a real-time database and for secure login authorisation. I have also added in google maps activity so that when a user logs and their location is shown in app and I have added in to sign in another user and show how far they are from one another. Also, in another prototype, I have added in that they can use their network provider or GPS if network fails, as shown below.

```
locationManager.requestLocationUpdates(LocationManager.NETWORK_PROVIDER, 0, 0, new LocationListener() {
    @Override
    public void onLocationChanged(Location location) {
        //Get Lat
        double latitude = location.getLatitude();
        //Get Long
        double longitude = location.getLongitude();

        //Instantiate class: LatLong
        LatLng latLng = new LatLng(latitude, longitude);

        //Instantiate class: Geocoder
        Geocoder geocoder = new Geocoder(getApplicationContext());
        try{
            List<Address> addressList = geocoder.getFromLocation(latitude, longitude, 1);
            String str = addressList.get(0).getLocality() + ",";
            str += addressList.get(0).getCountryName();
            mMap.addMarker(new MarkerOptions().position(latLng).title(str));
            mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(latLng, 10.2f));
        }catch (IOException e){
            e.printStackTrace();
        }

    }
```

Figure 20: Network code screenshot

As shown above this code function can use the network provider the user has and get the co-ordinates of the user and show their location in the app. Also added for learning purposes is the country where the user is.

```
else if (locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER)){

    locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 0, new LocationListener() {
        @Override
        public void onLocationChanged(Location location) {

            //Get Lat
            double latitude = location.getLatitude();
            //Get Long
            double longitude = location.getLongitude();

            //Instantiate class: LatLong
            LatLng latLng = new LatLng(latitude, longitude);

            //Instantiate class: Geocoder
            Geocoder geocoder = new Geocoder(getApplicationContext());
            try{
                List<Address> addressList = geocoder.getFromLocation(latitude, longitude, 1);
                String str = addressList.get(0).getLocality() + ",";
                str += addressList.get(0).getCountryName();
                mMap.addMarker(new MarkerOptions().position(latLng).title(str));
                mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(latLng, 10.2f));
            }catch (IOException e){
                e.printStackTrace();
            }

        }
```

Figure 21: GPS code screenshot

As shown above the same with the network provider if that fails the code then goes to the GPS provider to get the users location and show it in app.

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    btnLogin = (Button)findViewById(R.id.btnSignIn);
    btnLogin.setOnClickListener((v) → {

            startActivityForResult(
                    AuthUI.getInstance().createSignInIntentBuilder().setAllowNewEmailAccounts(true).build(), LOGIN_PERMISSION
            );

    });
}

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {

    if(requestCode == LOGIN_PERMISSION){
        startNewActivity(resultCode,data);
    }
}

private void startNewActivity(int resultCode, Intent data) {

    if(resultCode == RESULT_OK){

        Intent intent = new Intent( packageContext: MainActivity.this,UserOnline.class);
        startActivity(intent);
        finish();
    }
    else{
```

Figure 22: Code Screenshot

In the figure above this shows the sign in feature of SessionOn that works with the firebase authentication. As shown below.

Identifier	Providers	Created	Signed In	User UID ↑
ceino1995@gmail.com	✉	21 Nov 2017	13 Apr 2018	Ntjwd7zCpNVo7obBXgOtXKwDIHJ2
c14427818@mydit.ie	✉	21 Nov 2017	13 Apr 2018	YEwBL2BFlPMB7CxwfTfJ8LjuQFP2

Rows per page: 50 ▾ 1-2 of 2 < >

Figure 23: Authentication screenshot

As you can see these are the emails used to login to the app and are password protected to get into the application. This really gives the app a good security and will aid in the prevention of hacking etc.

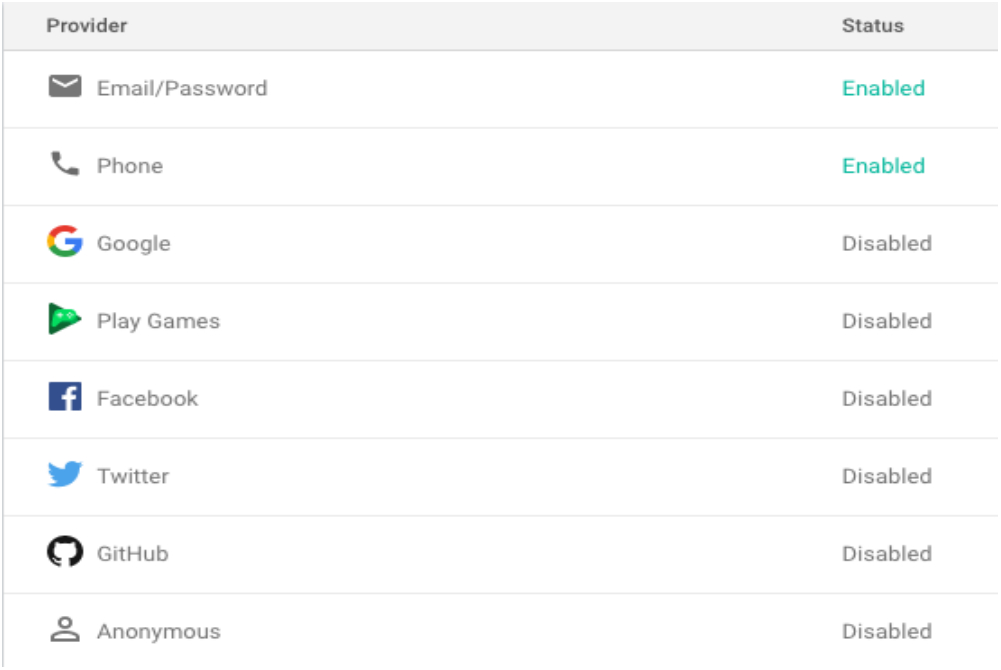

Figure 24: Firebase sign in methods screenshot

As seen above Firebase provides multiple methods of signing in that can be enabled and disabled at the click of a button. SessionOn uses the email and password as it is more traditional and then also provides the means to reset either with an email sent with a link to change the email or password to whatever the user desires it to be. As shown below.

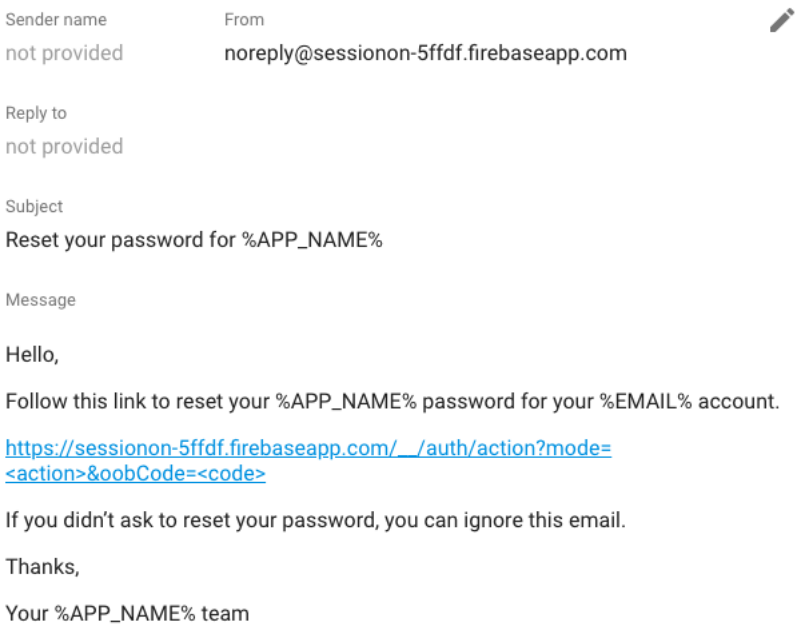

Figure 25: Password verification screenshot

6. Testing

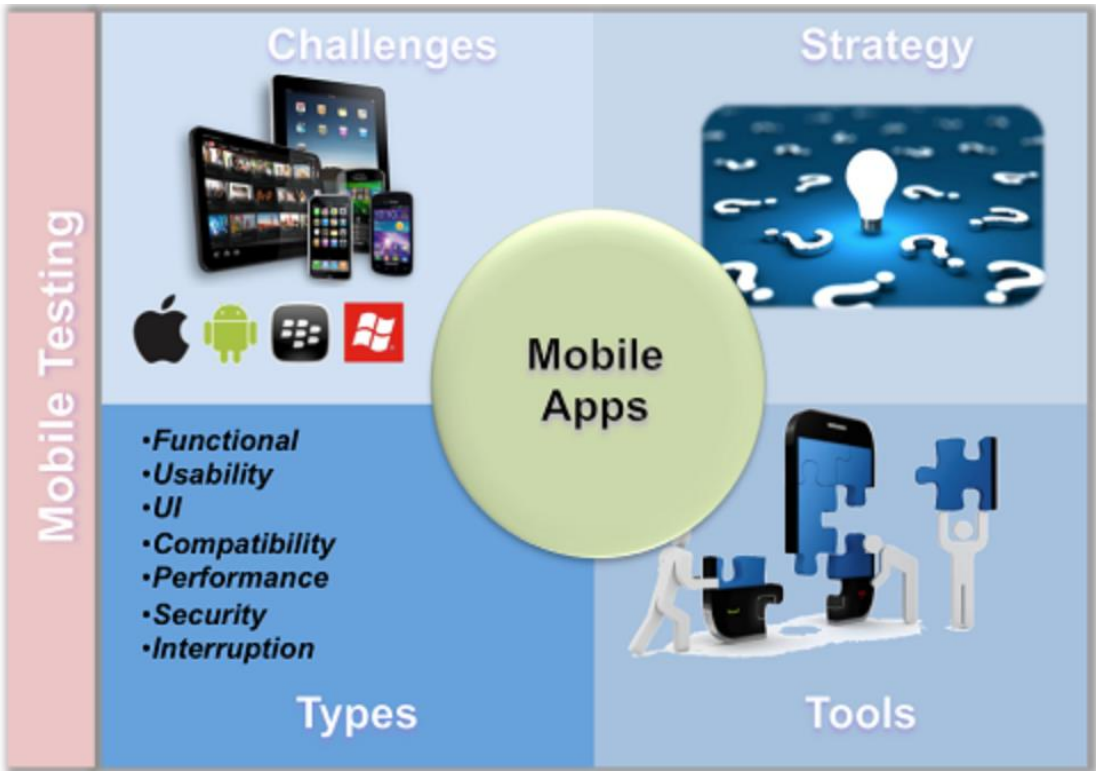

Figure 26: App testing challenges [4]

So far, I have just been testing the app by myself. Seeing if a user can log in and does the app broadcast the location correctly. Today building app is not just about writing a good code, the success of app is largely driven by user experience. A successful app should have an aesthetically pleasing UI and should deliver best user experience on all devices and various form factors, of course there are other important factors to be taken into consideration as well. Outstanding testing strategy is the only way to make your mobile app ready for business. "[4]. Hopefully soon I will be able to do some usability testing getting a questionnaire done to and getting users to complete it as they test the app.

I have still been testing the app alone and have make progress as I ran 2 emulators and signed into different accounts and joined and left the online status and checked the whereabouts of the users who were online. The app does work differently to what I set it out to be, but it does hit the goal of taking away the organisation of trying to bring friends together. The idea that someone can just go out sign into the app and see if

anyone else is online and around their area and then the user has the choice to meet up with them or not.

7. Issues and Risks

One of the issues I feel that I will have is making enough platform considerations, like making the app executable on IOS and windows phones. This could be a huge downfall for the app narrowing my clients down to ones that only have android phones. “While any app developer will find themselves forced to confront this question, often they fail to comprehensively dissect each option at their disposal. Some developers base their platform choice on stereotypes. It’s not uncommon to see someone develop for iOS just because they believe monetization would be easier on the App Store. While this may be true in some cases, it’s not universal. Additionally, there are many other considerations that must go into platform selection.” [5]. Android phones are more popular on a global scale but IOS is still extremely popular. Only being able to run on one could be a serious downfall for my application.

Also, another issue I found that was quite tedious is the google play services updating throughout the creation of the project. Trying to get the right versions to work with each other was quite annoying. Also, the way SessionOn is coded was to suit an older version of firebase and google play services and trying to update the code that was already written was a time consuming and extremely strenuous procedure.

Another issue that SessionOn encountered was the emulator for android mobile devices was quite buggy and seems to crash, even though there is no problem with the code itself. Also, sometimes the IDE was a little buggy and when trying to add firebase, The IDE added the wrong version or the version that was not the latest.

Another issue I feel that could be a big downfall is in relation to the notifications. Unfortunately, I was not able to incorporate a notifications feature into the app and I do think that this is a downfall fall as the user has to use the app to check if a friend is on the move rather than getting a notification on screen to keep them up to date.

Back tracking onto the versioning problem, when I started the development of this app the version was different to what it is now and people that do not have their phone up to date for any number of reasons will not be able to use this app. This was also a problem then trying to test the app on certain emulators as I had to create new ones with the latest update to try and run the app which resulted in crashed and the app not running at all and malfunctioning.

8. Plan and Future Work

My plan for future work is to hopefully get the application fully functional and up onto the play store and see if it takes off. I still believe there is work to be done on it and although it is not exactly as I had portrayed or hoped it to be I still believe it is close to what was the use off the app idea to begin with. Now that I have all the firebase and google play services up to date and working together without problem, I believe Firebase could add on some cool features to the application. The main one being the cloud messaging. If that can be incorporated into the app and then used, I think that SessionOn could definitely thrive as a social app. Also, SessionOn could go further and take on the route of maybe adding a profile with a wall so users could update photos and then maybe give others a reason to track and join their sessions. In relation to the messaging aspect I feel that this could be hugely beneficial to the app in the sense that they can see where the person is and then quickly shoot them a message and find out what they are up to and if there are any future plans. Also, if the person moves and the forgets to text saying that they are moving, all the friend has to do is check where they are again and send another message to see what the plan is.

9. Conclusions

The experience I have gained from doing this project is second to none. I can honestly say that I have never faced a challenged like this before and even though sometimes trying to figure certain parts out and failing along the way was tedious, I can still say that I have enjoyed the overall experience and that it will aid me in a future career. I

have never worked with the android platform before or done any kind of application development to this extent and have found it to be invaluable to me. From trying to find an idea that would benefit people on a social level to actually developing something that with a bit more work could be a popular social app. Although I did not achieve the full idea and changed it long the way, I still feel that it could work well as a social app and be useful to people of all ages. Building the app from the foundation up have really been an eye opener to how long development can actually take and is not something that is done overnight. Also researching the new technologies and trying to get them to work together to make the app more stable. Overall, I feel more confident in my knowledge of the android platform and more importantly, my skills in the Java language were certainly strengthened. Also, I believe that it will aid me in future technical interviews if given some coding to do. I believe it will help me towards solving any algorithms thrown at me. Although SessionOn is not exactly what I set out to do I feel that it has hit certain goals and taken on a new role as instead of trying to create a session it gives the users the ability to see where others are and find out what they are doing and decide whether to turn it into something big or just some small and social. I think that with a little more work and the addition of a few more features it could be something quite useful.

10. References:

[1] available from: https://www.life360.com/how_it_works/ Author: Life360 Accessed: 29/11/17

[2] available from: http://sites.path.org/rhtech/files/2012/07/Iterative-design-process.jpg accessed: 26/11/17

[3] available from: http://alistair.cockburn.us/incremental+versus+iterative+development accessed: 26/11/17 Author: Alistair Cockburn

[4] available from: http://www.hummingwave.com/web/mobile-testing.htm accessed: 26/11/17 Author: Sandeep Koul

[5] available from: https://www.entrepreneur.com/article/238849 accessed: 26/11/17 Author: Himanshu Sareen