Enhancements to the FOSSAsia Open Event Android App

Abstract:

Enhancing the FOSSAsia Open Event Template to add new functionalities like
Fixing app specific bugs and optimize app's performance and stabilization
Android Wear support
Enhancing the Map Activity
Adding social features in the app
Enhancing the general UI/UX of the app.

This project will be most beneficial for those people need an App for any event that they will be hosting but are forced to pay a said amount of Money to have it made from any indie developer.

Current status of the Project:

Though the FOSSAsia Open Event Android App is currently in a functional state yet there are many issues to be sorted out.
While most of the bugs that I found out were related to UI/UX, few of them are also critical to the functioning of the app itself.
The project also lacks some features that will be a lot helpful to the users.

Project Goals:

1. The aim of the project is to enable anyone with little or no knowledge of Android development to be able to create a working app effortlessly using Android Studio via replacing the placeholder URLs by the correct ones and replacing the app's description by modifying components in strings.xml.

2. Adding the ability for a user to compile the app **without using** Android Studio by adding a section in the app for the same that asks for the proper API URLs and icon file and generates an Android app.
   We can also do the same by using a script and creating a webapp out of it that does the same.

3. Adding support for Android Wear so as to make it possible for the user to get the notifications for the event he/she has bookmarked, allowing him/her to mark the attendance after the event has started, get the directions to the event and rate the evnt once it has concluded.

4. Ability for a user to mark attendance for a session, rate the session and also there will be a separate section in the app that will show the recent Google+ posts by the Organization that is hosting the event.

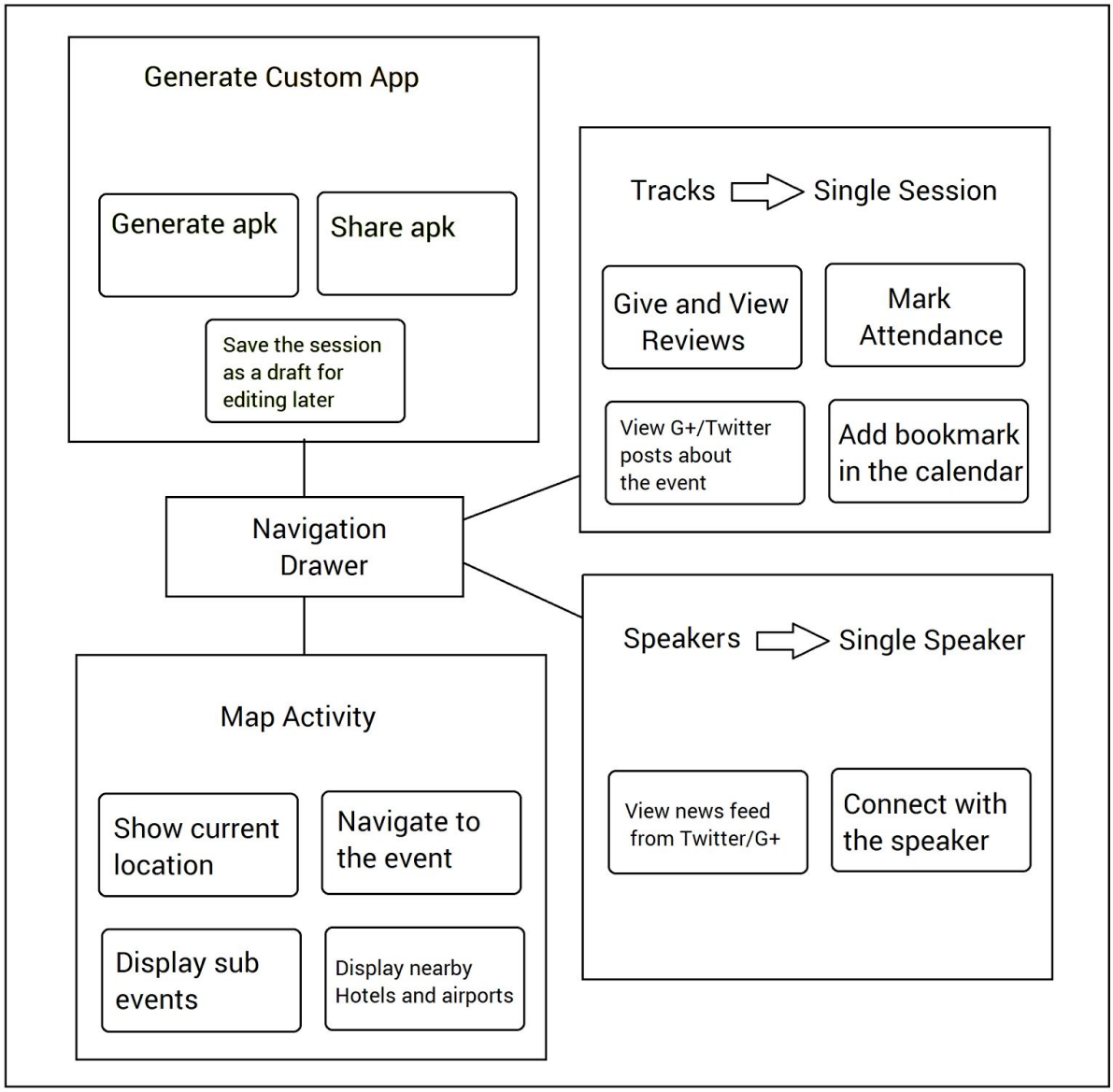
5. Enhancing functionalities of the Map Module in the App.

6. Improving the runtime performance of the app so as to make it consume less system resources by adding LRU caching while adding social media functionality in speaker details activity. #perfMatters
   This also includes proper testing, debugging and writing Unit Tests.

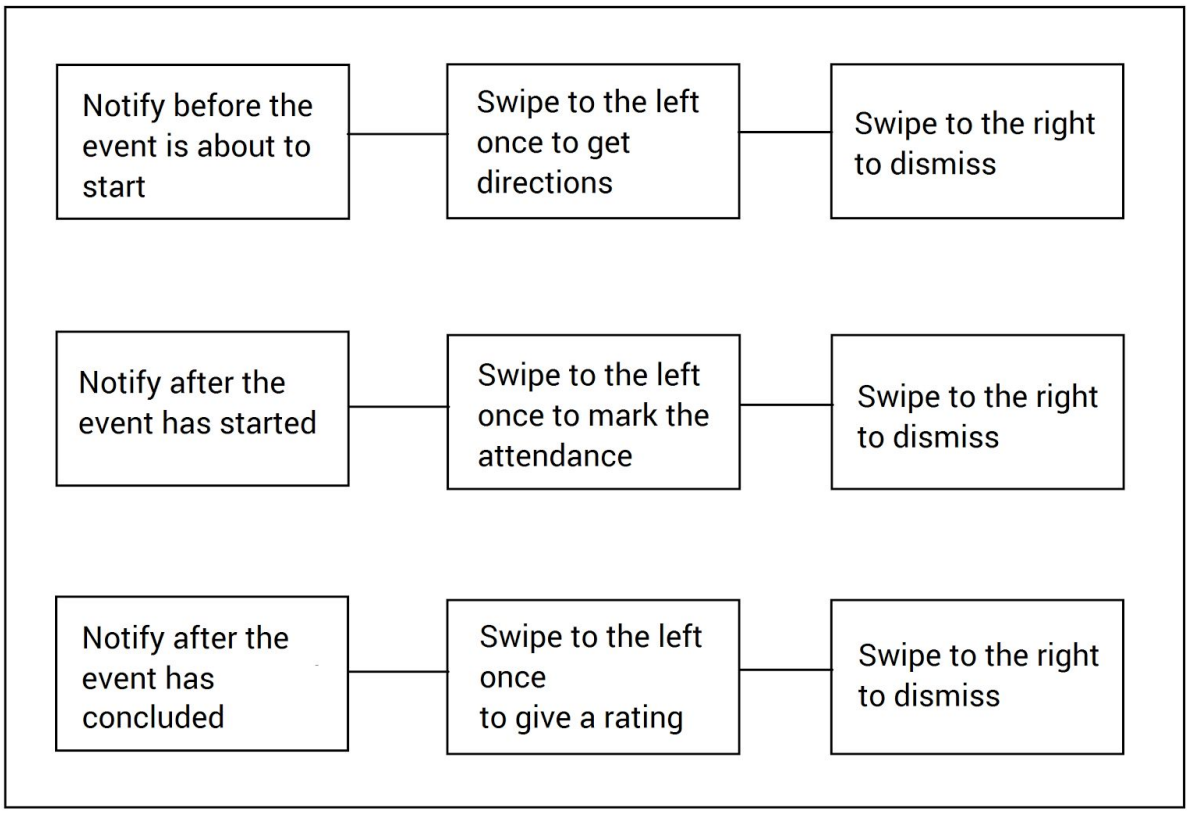
7. Adding a section that displays social features like the recent G+/Twitter posts by the organization and the speaker.
   Also adding a section that displays the photos from the event after it has concluded.

8. Improving the app's general UI/UX by making it optimized for tablets and redesign parts of app by following material guidelines

Wireframe for the final App:



Wireframe (for the modified components only) in the phone/tablet version of the app.



Wireframe for the Android Wear Module of the app

Implementation:

1. So as to make the app build simple for non-experienced users, I will be providing a good documentation along with a FAQ.
   I will also try to modify the existing code to be as user friendly as possible by using relatable variable names and commenting the code as much as possible.

2. If we don't want the user to go through the hassle of installing Android Studio, the repository for the open-event app will also have a sample apk which will have an option (enabled by default) to allow the user to generate a signed apk.
   This option can be hidden or shown by calling a function named `enableDevmode(bool)` with an appropriate parameter.
   Implementing this won't be that hard as it has already been done by BuildmLearning in their GSoC 2015 project, so I will be taking help from their codebase.
   The methods like the ones listed below will be used for generating the modified .apk.

   ```
   KeyStoreDetails("TestKeyStore.jks", keyPassword, aliasName, aliaspassword);
   //For generating the signing key
   SignerThread(getApplicationContext(), getOutPath(), keyStoreDetails,
   getIconPath(), getUrlNames()); //For generating the .apk file
   ```

   The app can also be created by creating a script that takes the API url as input and automatically generates the apk using the buildbot.
   This can further be realized as a web app that will grant the user to customize the app according to his need.

3. The wear module will be responsible for notifying the user before his bookmarked event is about to start.

   ```
   NotificationCompat.Builder builder = new
   NotificationCompat.Builder this);
   builder.setContentTitle("Title of the Track").setContentText("Details
   about track and
   time").setSmallIcon(R.drawable.ic_launcher).setContentIntent(PendingInten
   t); //For building the notification
   NotificationManagerCompat notificationManager =
   NotificationManagerCompat.from(this);
   notificationManager.notify(notificationId,notificationBuilder.build());
   //For displaying the notification on the wear device
   ```

   It will also provide the driving directions to the user to navigate to the event and prompt him/her to mark his attendance once the event has started.
   This will be most helpful for the organizers, in case they want to provide quick

refreshments for the attendees.

4. I will be adding the ability to mark the attendance and rate a session after it has concluded.
   Note: This idea is already in discussion my Manan and Mohit over here (https://github.com/fossasia/open-event-android/issues/167) , and I would like to help them implement this.
   Also a section can be added to the session details that shows the reviews given to the session by the attendees.
   These features will also be present on the Wear module, so that the user can swipe the notification to mark his attendance.

5. As of now, the Map Fragment only shows the location for the event, I will be enhancing it to show the current location of the user and allow navigation to the event from within the app itself, or add a button that will open the Google Map and allow the user to navigate to the event.
   The Map activity will also show the location of nearby Hotels and Airports.
   As there might be many sub events happening at once, I'll also add another map activity that will show the location of each event that is taking place in the Campus.
   The maps will also be cached for offline usage.
   Also for F Droid users, we can use a static image file instead of maps as they might not have an actual Map app installed on their device.

6. Currently, the App uses a service to notify the user for his bookmarked event.
   My plan is to give an option to the user to **store the reminder for the event in the inbuilt calendar** app instead of doing it by the service.Doing this will cause the app, not to hog the resources on lower end devices.
   The code will be similar to the following code snippet:

```
Calendar beginTime = Calendar.getInstance();
beginTime.set(getEventStart());
Calendar endTime = Calendar.getInstance();
endTime.set(getEventEnd());
Intent set = new
Intent(Intent.ACTION_INSERT).setData(shortEventDesc)..putExtra(CalendarContract.
EXTRA_EVENT_BEGIN_TIME).putExtra(CalendarContract.EXTRA_EVENT_END_TIME);
```

   LRU image caching can also be implemented if we are planning to integrate social features like recent posts from the organization and the speaker.

   This part will also include exception handling that happen/might happen in the app when there is a URL mismatch with the App and the Server.
   This can be easily done by adding try catch statements at the points where a Force

Close can happen and displaying a toast to the user indicating that something went wrong.
I also plan to write proper Unit Tests for the project as they'll help me in reducing the bugs while adding new features to the app.

7. For implementing section that shows recent posts by organization and speaker, we can use the Google+ api and display the posts in a RecyclerView.
   I was also thinking about adding a section in the app that will show the Pictures from the event similar to the fossasia website.
   The pics can be stored into a github repo and the app can get the json from that repository using the Github API by calling the endpoint "`/users/:username/repos`".
   I will further create a custom adapter that will be responsible for setting the images into a recyclerView from the URLs in the fetched JSON Object.
   I have already implemented something similar in my icon pack.

8. I will be enhancing almost entire UI of the app by following Google's Material Guidelines so as to make it look like a polished android App.
   I already have mockups of the final app ready, which I have posted below.
   This includes adding a tablet UI to the app consisting of a Master Detail flow.

Timeline:

GSoC is round about 12 week duration, with about 25 days of Community Bonding Period in addition.
I will be spending **20% time in fixing the bugs** left out in the current version of the app, adding wear module support and making the template ready for public use.

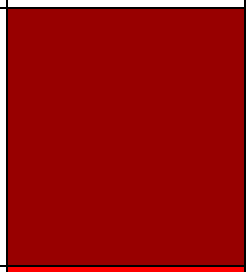
45% time on adding new features to the App.

15% of the time on enhancing the UI of the app by following material guidelines and remaining **20% time on testing the app**, preparing Wiki and FAQ for the template.

I will be needing a **break for about 9 days** in the month of May for my end Semester exams, which I will cover up by chipping in some extra hours daily after the exams.

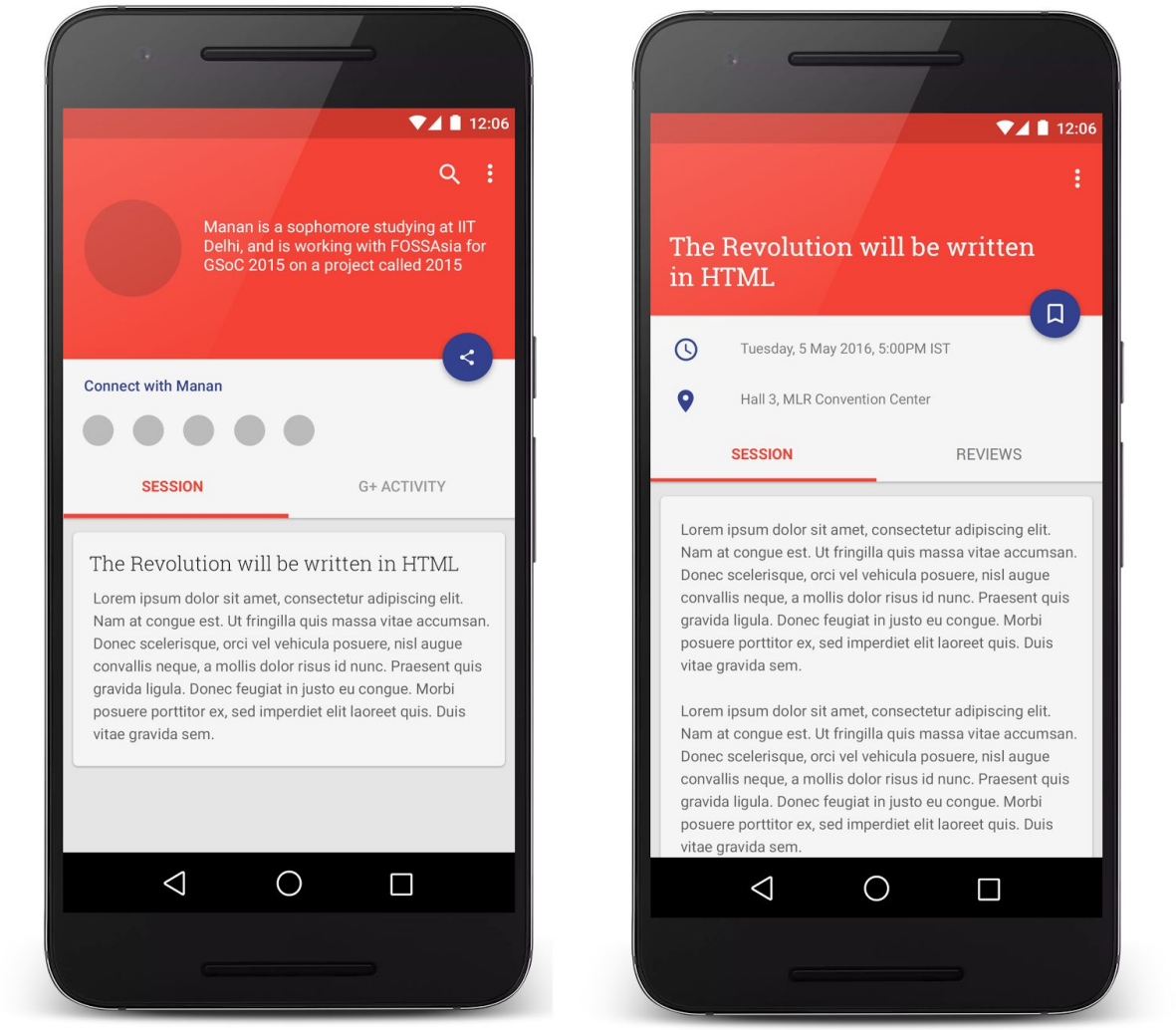
The detailed timeline is linked below:
Legend: Importance and Time Devoted: > > >

Time Frame	Start Date	End Date	Task	
Community Bonding	22nd April	23rd May		
	23rd April	27th April	Requirement Gathering and Documentation. Allowing the user to be notified using the default calendar app instead of using a service.	
	27th April	1st May	UI Enhancement of the app - I.	
	1st May	10th May	Break for End Semester Exams.	
	11th May	22nd May	Improving Map Module to display all the sub events and nearby hotels and Airports. Start working on the Wear Module.	
Blog Posts			4 Blog posts on Starting my Journey for GSoC 2016, designing an app following material guidelines, implementing Map module in your app and creating a service.	

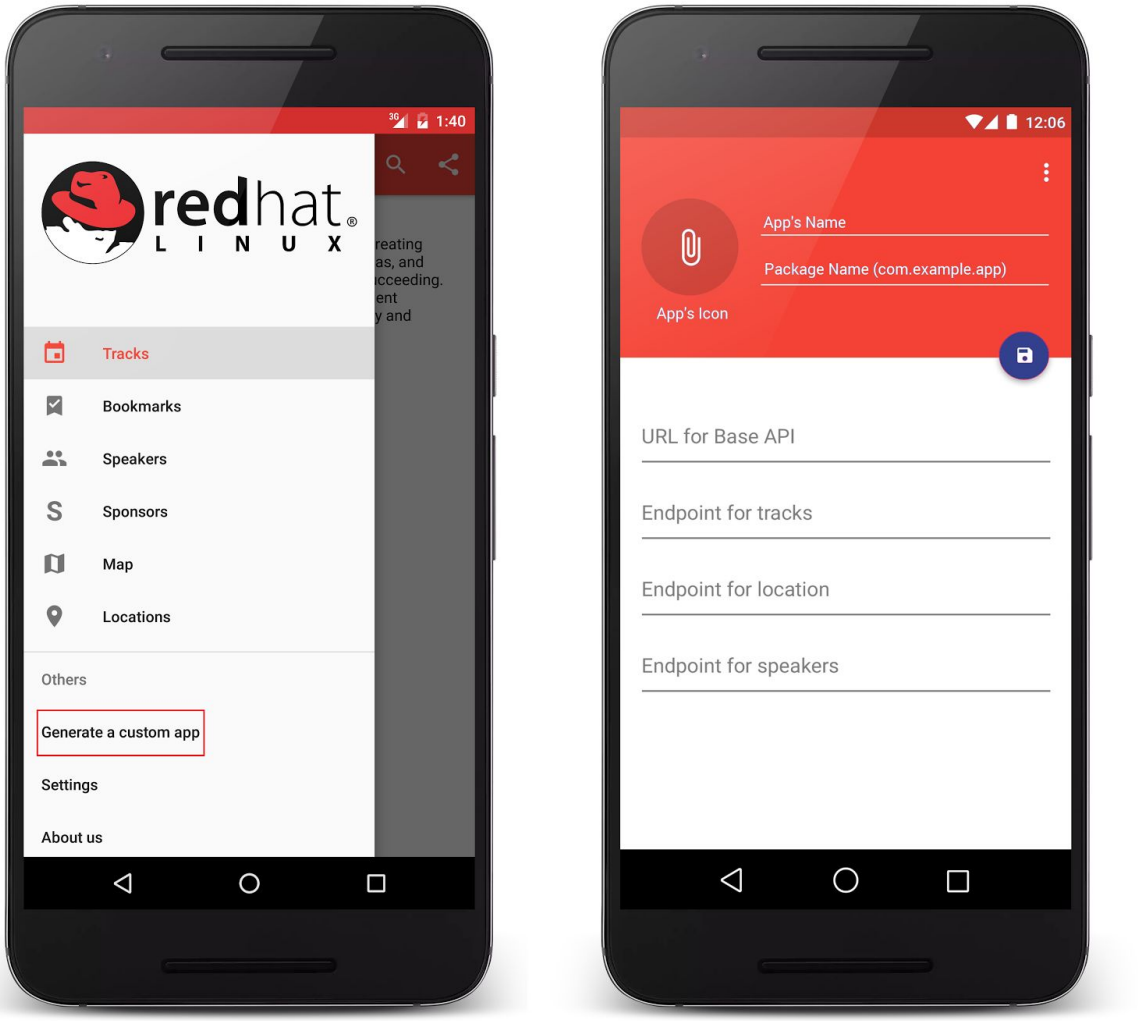
Mid Term Evaluation	23rd May	27th June		
	24th May	10th June	FInish creating the Wear Module by adding the option for marking the attendance, reviewing the event and getting the directions for the event.	
	11th June	22th June	Adding an option to display the images from the events after it has concluded.	
	22th June	27th June	UI Enhancement of the app - II including the support for Tablets.	
Blog Posts			3 Blog posts on how to add Android Wear support to your app, Making networking calls with Retrofit and Working with the Maps Activity in Android.	
End Result			The Template will be almost Production Ready by 27th June with all the basic functionalities included, the only things that would be remaining will be adding Social features and allowing the app to be compiled without using Android Studio.	

Final Evaluation	27th June	29th August		
	28th June	14th July	Adding a section to show most recent G+/Twitter posts by the organization and the speaker and allowing the user to rate the event.	
	10th July	28th July	Adding the ability to compile the app without using Android Studio by allowing the app to be compiled using both the sample app provided and a script that will be later realized into a web app.	
	28th July	7th August	Writing Unit Tests and Final UI tuning of the app.	
	8h August	20th August	Thorough Testing, Exception Handling and distributing the app among beta testers (if allowed).	
	20h August	27th August	Preparing Documentations, Wiki and FAQs and a Webcast on the FInal Product.	
Blog Posts			4 Blog posts on Implementing Master Detail flow in your app, signing and exporting an android app from your phone, What is Material Design and why do you need it, Need of testing & debugging your app.	
End Result			The Template will be fully ready including Good Documentation and ability to generate custom .apk via the provided sample app and a Webapp.	

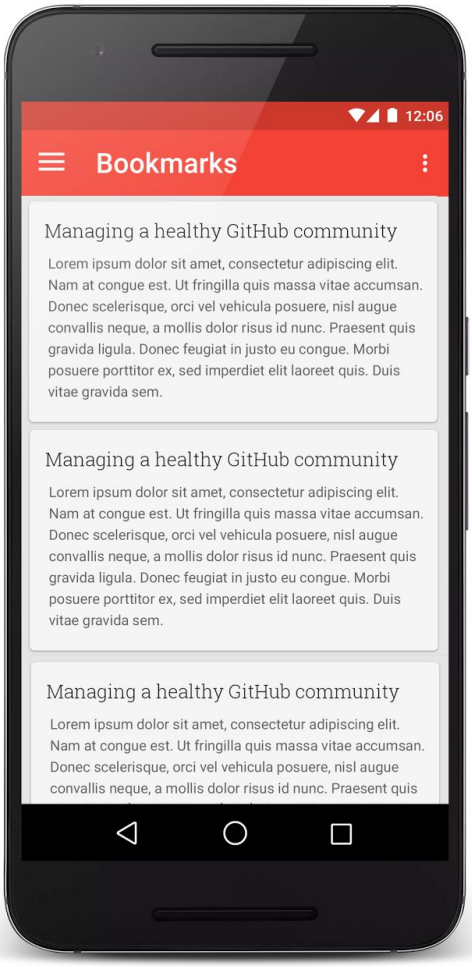
UI UX Mockups:



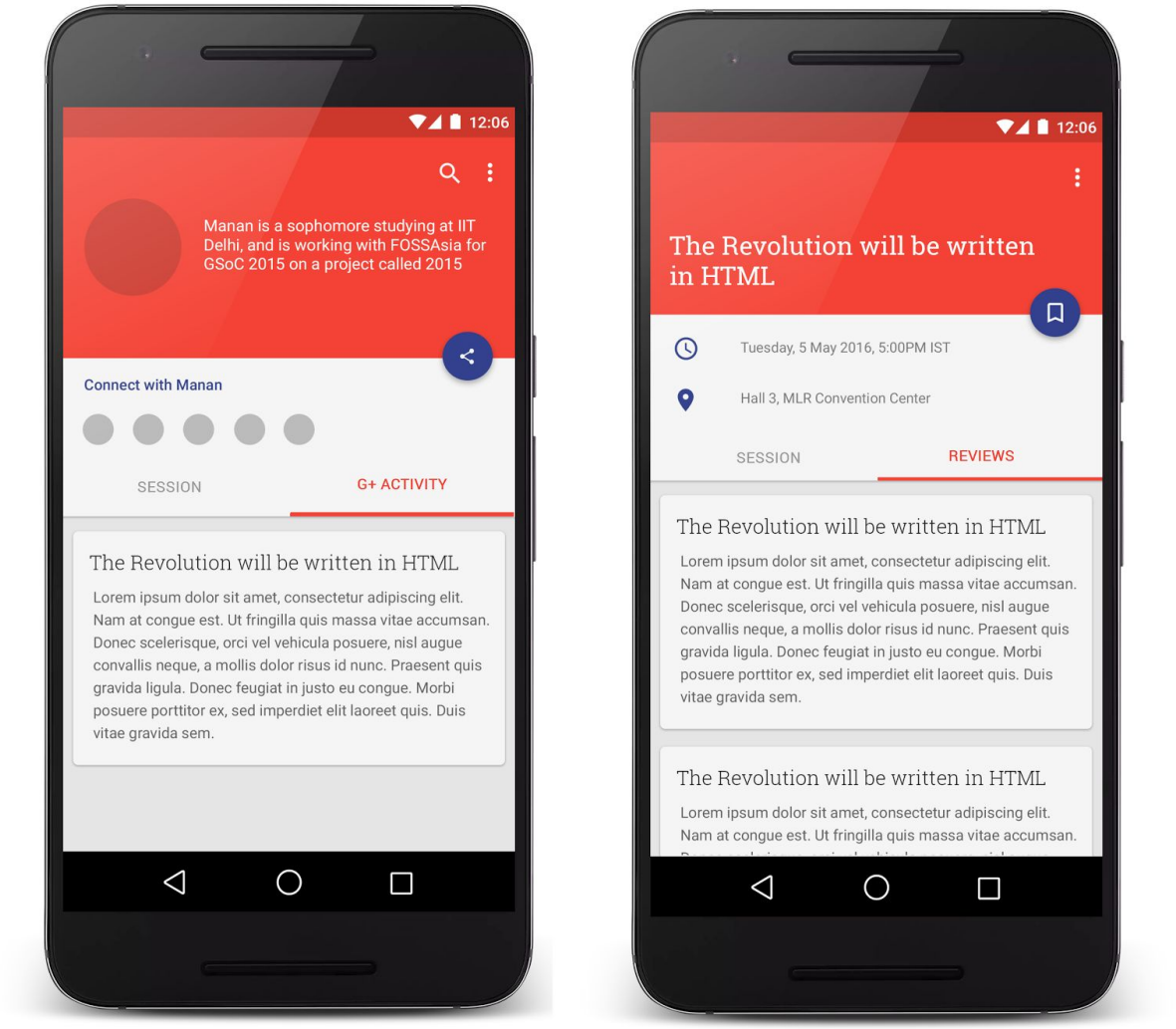
Redesigned SpeakerDetails Activity (Left)
Redesigned SessionDetails Activity (Right)



Generating the custom app from the provided Sample apk



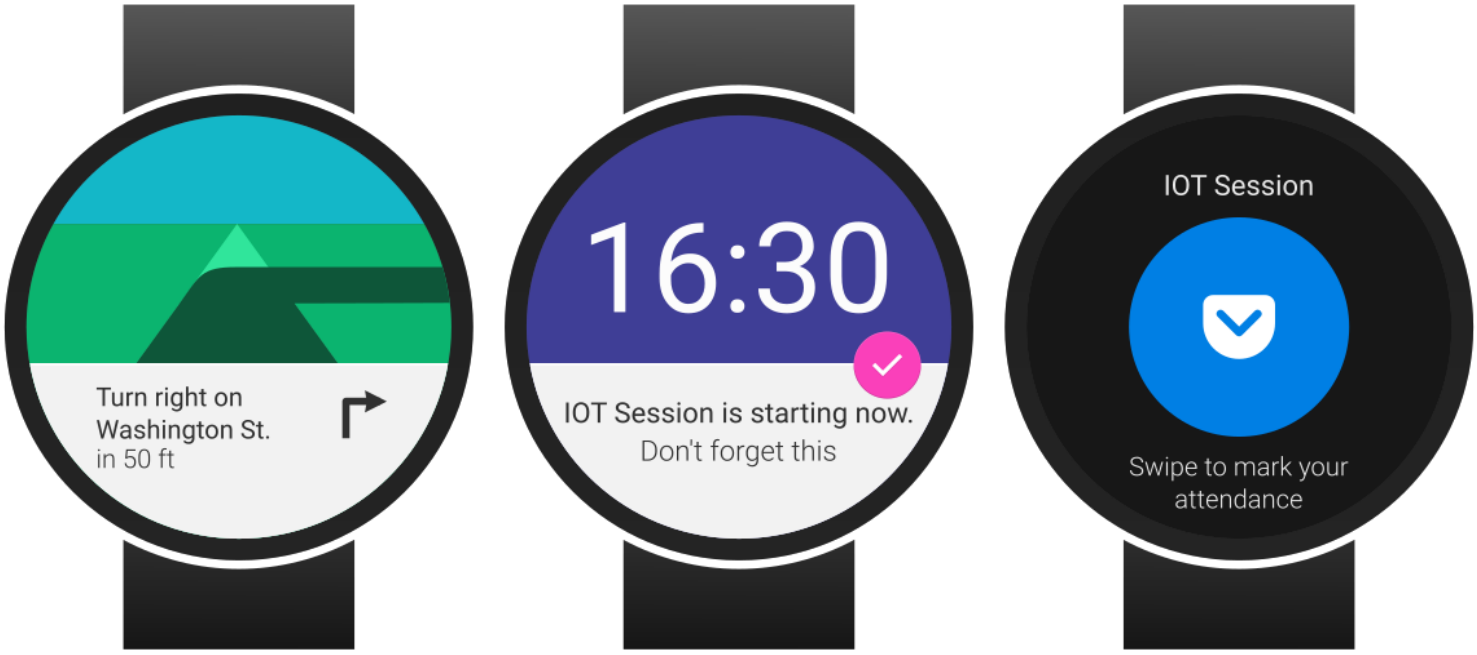
Using better colors and fonts for the CardViews throughout in the app (Left)
Displaying individual events too in the mapActivity (Right)



Section displaying the latest Google+ posts by the Speaker (left)

Section displaying reviews given to a particular event by the attendees (Right)

A better Splash Screen



Mockups for the Wear Module
Center: Notification before the event.
Right: Mark the attendance.
Left: Navigate to the event.

My Contributions:

My contributions to FOSSAsia's open-event-android repository are listed below.

Merged Pull Requests : https://goo.gl/AIkAph
Open Pull Requests : https://goo.gl/MFx3Rg
Issues reported (3 fixed) : https://goo.gl/GShOrm
Ideas suggested : https://goo.gl/vHMG14

I have also gone through the whole source code of the app and I believe that I have well understood the working of the API as well, so I believe that I will be able to work comfortably during the summer without much difficulties.

As I have already implemented some of the features that I have listed above in my own apps, I believe it won't be a difficult job implementing them here as well.

My Ideas for getting more developers on board during my summer project:

Since I am a part of the Google Developer Group New Delhi, we hold bi weekly sessions in which we promote OSS and teach students to work on technologies like FIrebase, Android etc. Now, I can use this platform and after the session, we can have a brainstorming session with the attendees on what are the things that they would love to see in such an app.
I will be doing the same for the web app and the IOS app too.
I will be encouraging them to open issues in the repo regarding the existing bugs/features and ask them to try to fix them and send a PR for the same.

I also run Study Jams in my college in which I teach Juniors, the basics of Android App development.
There are currently 45 students in the current batch, so the same thing can be done with them too.
This way, we would not only get more devs on board, but it will also help students in exploring the Open Source World as well.

How much time I will be able to contribute to the project:

My timezone is GMT + 5:30 and I will be able to devote 30-35 hours per week to the project. Apart from this, I'll also make a weekly blog post and a medium post highlighting the developments of my project.
I will be devoting my full time to this project and I do not have any other Summer internship or Job.

About me:

I am a 3rd year undergraduate student pursuing my degree in Electronics and Communication engineering from GGSIP University Dwarka.

I want to work with FOSSAsia on this project for the summer as I think that I have required skills so as to bring this project to its completion.
I had already started started to work on an open sourced Tech Fest app template for colleges in the last year.
It's partially done and the link can be found here. (https://github.com/the-dagger/Tech-Fest)

I am an Android Developer and have a total of 5 apps on the Google Play store.(https://goo.gl/wl3loD)
I am also an avid Open Source lover and the source code to most of my apps can be easily found on my GitHub profile. (https://github.com/the-dagger)

Apart from this, I also have received the Scholarship for Android Developer nanodegree provided by Google and Tata Trust for the students in India and have made 2 projects under it as of now.
The projects done are:
Popular movies (https://github.com/the-dagger/Popular-Movies)
Personal Portifolio App (https://github.com/the-dagger/App_Portifolio)

I can assure you that if I get selected to work with FOSSAsia this summer, I definitely will try my level best to make this project successful and will love to continue working with FOSSAsia's other projects even after the summer.
I would love to mentor in any Educational programs that FOSSAsia is a part of and I am already participating in various meetups through the GDG platform so doing the same with FOSSAsia will be no problem at all for me.
Further, I would really love to attend the FOSSAsia Summit in 2017. (Well, who wouldn't :p)

Also for some reasons, if I am not selected this year even then I'll try to contribute to this and other projects as much as possible and retry again next year.

Looking forward to working with you.
Harshit Dwivedi.