ME Software Systems
Department of CSIS
BITS Pilani

WHEREABOUTS PROJECT REPORT V1.0

NOVEMBER 26, 2016

This document describes the overall process adopted while building the Whereabouts Software including "team and project management" details.

Contents

Revision History

Name	Verified by	Date	Reason For Changes	Version
Manpreet	Kartikeya	26-Nov-2016	Document creation	1.0

1. Introduction

1.1 Purpose

The purpose of this document is to present a detailed description project scheduling and team dynamics of the Android based mobile application Whereabouts. It will explain the project lifecycle along with a detailed description regarding the team dynamics. This document is intended for both the stakeholders and the developers of the system and will be proposed to Mr. Kuldeep Kumar, Assistant Professor, Department of Computer Science and Information Systems, Birla Institute of Technology and Science, Pilani Campus, Pilani - 333031, Rajasthan, India for its approval.

1.2 Scope

The Android based mobile application "Whereabouts" developed in this project provides a platform for instant messaging and location tracking to the users. The user is provided with a screen displaying the current location of his peers as markers on the map. These markers are updated in real time. Chat option is provided to the users on clicking any of the selected peer. A facility of scheduling a message (this feature will be referred to as "Drop Message" in the subsequent content of this document) is also provided to the user. The scheduling of message takes into account the location of selected peer and date and time as desired by the user. This will be explained in detail in later sections.

This application bring location tracking and messaging on a common platform. This can be used beneficially in numerous scenarios.

1.3 Definitions, acronyms and abbreviations

The acronyms are given below:-

Term	Definition
App	An application program (app or application for short) is a computer program designed to perform a group of coordinated functions, tasks, or activities for the benefit of the user.
Project	The project for development of Android based mobile application "Whereabouts" developed by M.A.S.K.
SRS	Software Requirements Specification document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document.
Peer	Once a user accepts the peer connection request from another user, the two users become peers on each other.
API	Application Programming Interface

1.4 Overview of Report

The project report for this system contains various sections describing various details about the project. Section 2 covers the motivation behind the project, in section 3 various activities associated with project management, section 4 covers various phases in which the project was developed, Section 5 contains the various components of the project. Section 6 and 7 describe about the technologies used and the outcomes associated with the project. Section 8 describes about the various types of challenges faced during the creation of the project and the learning received from this project and section 9 describes about the enhancement possible with respect to the project and the course for which this project is created.

2. Project Motivation

The various factors which motivated the development of this project were:-

1. Instant messaging and location tracking applications are quite popular nowadays, so to bring the two services on a common platform seemed to be attractive idea.
2. In day to day life, peers are often interested in locating and communicating with each other. Thus to provide an application serving both the interests, that could be used with a hand held smart phone sparked the idea to build this application

3. Project Phases

3.1 Requirement Analysis Phase

This phase consisted of gathering of requirements and features that would be proposed to be a part of the application. For identifying the requirements discussion with various groups of students of BITS, Pilani was conducted and the suggested features were analyzed for feasibility.
The requirement specification document has been attached in the attachment section [1] for reference.

3.2 Design Phase

This phase started with the analysis of the requirements to find out the core functionalities of the system which was developed. This phase included the design of various components of the system. Various design decisions were taken in this phase. The latest popular chat service platform, Firebase, was finalized as the platform to be used for building the application. In addition to this various Google APIs were explored and finalized for use in the implementation of the application.

A clear distribution of responsibilities was done after finalizing the design and analyzing the required skill sets of the team members. Abstract API's were prepared and pseudo-code was written for identified ADTs. The design document has been attached in the attachment section [2] for reference.

3.3 Development Phase

This phase started with the development of various modules identified during the design phase. The development of the Android application was done using the Android studio version 2.1.Regular discussions and meetings were held between the team members to resolve various implementation issues faced during development. The individual modules were tested by the developers and the module functionality was verified by another team member before integration testing.

3.4 Testing Phase

Unit testing was done by the developers. Unit testing and development phase were overlapped and internal audit by a peer team member was carried out for each module developed.

Integration test plan and system test plan were provided to the team members for the complete testing of the application.

3.5 Release and Maintenance Phase

The release of the application is planned for user acceptance testing post the completion of integration and system testing. The application will be freely available for download from the given link-
https://drive.google.com/open?id=0B2lfhTC9ePF-eTZJUks3NDRYa2c

For any support related issues users can contact the team by mailing at-
mask.whereabouts.support@gmail.com

4. Project Components

4.1 Authentication Component

The user is provided with "login with Google" functionality on application startup for the first time. He/she then logs in with the application with the credentials used in his/her device for Google account. These details are stored in the application environment and the user need not sign in every time he/she starts the application.

On first time login, the user is prompted to provide his/her phone number. After getting the phone number, home screen of application is presented to the user. On subsequent application startup, the user sees the application home screen.

4.2 Map Rendering Component

This modules displays the current location of the users and their peers on the Google map API. The module fetches the current location of the user and stores/updates it into the firebase database. The stored location of the peers are retrieve and update its location on the map. A chat screen will be prompt if the pointer of any peer is tapped.

4.3 Instant Messaging Component

The user is displayed his chat screen with the peer, with previous conversation being displayed. The user types in the new message he/she wants to send to the peer and on clicking on send button the message would be received by the peer. In similar way, the messages sent by the peer will be displayed on the user chat screen.

4.4 Peer Management Component

This module provides the user the functionality of adding peer through sending peer request. This request details will be stored in a database and the other peer receives notification in his peer request section. If user accepts the request then name entry of each other will appear in peer list section.

4.5 Drop Message Component

This module provides the user with the functionality to schedule a message that will be delivered to a peer only if that peer happens to be around a particular place in a particular time duration- both the place and duration is provided by the user. These details will be stored in a database table and the message will be delivered to the peer in form a notification if he/she is present at the user specified location within the time interval mentioned by the user. If the selected peer is not present at the mentioned location within specified time interval, the message will not be delivered to the peer and its lifetime will be timed out-such messages will be removed from the database and will never be received by the peer.

4.6 Relevant Documentation

The various documents which are part of this project could be accessed in the attachment section of this document.

5. Technology and Techniques Incorporated

The languages and OS used in this application are listed below:

1. **Java:** Java is a programming language expressly designed for use in the distributed environment of the Internet. Java can be used to create complete applications that may run on a single computer or be

distributed among servers and clients in a network. It can also be used to build a small application module or applet for use as part of a Web page.

2. **Android:** Android is an open source and Linux-based operating system for mobile devices such as smartphones and tablet computers. Android was developed by the Open Handset Alliance, led by Google, and other companies.

3. **SQL:** is a special-purpose programming language designed for managing data held in a relational database management system (RDBMS), or for stream processing in a relational data stream management system (RDSMS).

The technologies and frameworks used are:

1. **Google's Firebase Cloud Messaging:** Firebase Cloud Messaging (FCM) is a cross-platform messaging solution that lets you reliably deliver messages at no cost. Using FCM, you can notify a client app that new email or other data is available to sync. You can send notification messages to drive user re-engagement and retention. For use cases such as instant messaging, a message can transfer a payload of up to 4KB to a client app.

   An FCM implementation includes an app server that interacts with FCM via HTTP or XMPP protocol, and a client app. You can compose and send messages using the app server or the Notifications console.

   Firebase Notifications is built on Firebase Cloud Messaging and shares the same FCM SDK for client development. For testing or for sending marketing or engagement messages with powerful built-in targeting and analytics, you can use Notifications. For deployments with more complex messaging requirements, FCM is the right choice.

2. **Google's Firebase Real Time Database:** Store and sync data with our NoSQL cloud database. Data is synced across all clients in real time, and remains available when your app goes offline.

   The Firebase Real Time Database is a cloud-hosted database. Data is stored as JSON and synchronized in real time to every connected client. When you build cross-platform apps with our iOS, Android, and JavaScript SDKs, all of your clients share one Real Time Database instance and automatically receive updates with the newest data.

3. **Google's Firebase Authentication:** Most apps need to know the identity of a user. Knowing a user's identity allows an app to securely save user data in the cloud and provide the same personalized experience across all of the user's devices.

4. **SQLite Database:** SQLite is a relational database management system contained in a C programming library. In contrast to many other database management systems, SQLite is not a client–server database engine. Rather, it is embedded into the end program.

   Unlike client–server database management systems, the SQLite engine has no standalone processes with which the application program communicates. Instead, the SQLite library is linked in and thus becomes an integral part of the application program. The library can also be called dynamically. The application program uses SQLite's functionality through simple function calls, which reduce latency in database access: function calls within a single process are more efficient than inter-process communication. SQLite stores the entire database (definitions, tables, indices, and the data itself) as a single cross-platform file on a host machine. It implements this simple design by locking

the entire database file during writing. SQLite read operations can be multitasked, though writes can only be performed sequentially.

5. **Maps API from Google:** By using the Google Maps API, it is possible to embed Google Maps site into an external website, on to which site specific data can be overlaid. Although initially only a JavaScript API, the Maps API was expanded to include an API for Adobe Flash applications (but this has been deprecated), a service for retrieving static map images, and web services for performing geocoding, generating driving directions, and obtaining elevation profiles. Over 1,000,000 websites use the Google Maps API, making it the most heavily used web application development API.

6. **Geofence API from Google:** Google Play Services is a proprietary background service and API package for Android devices. When first introduced in 2012, it provided simple access to the Google+ APIs and OAuth 2.0, but has since then expanded to cover a large variety of Google's services, allowing applications to easily communicate with the services through common means, being internally referred to as simply GMS.

    The Location APIs abstract away specifics about the location technologies, providing Geofencing APIs for scheduling specific actions upon the user entering or leaving specific geographic boundaries, Fused Location Provider for acquiring location information with as reduced power usage as possible and activity recognition for allowing applications to adapt to the current action of the user (e.g. cycling, walking, etc.).

6. Project Outcomes

6.1 Final Product of the Project

The final expected product an Android based mobile application – "Whereabouts" will be ready for release and use after the completion of testing. The final product is expected to be in sync with requirements with which it is made.

6.2 Deliverables Agreed versus Deliverables Delivered

The status of agreed deliverables is mentioned in the below table-

S.No	Deliverable name	Status
1	SRS	Delivered
2	Design	Delivered
3	Code	Ready-To be delivered
4	Test Plan	Ready-To be delivered
5	Whereabouts APK	Ready-To be delivered
6	Developer Manual	Ready-To be delivered
7	User Manual	Ready-To be delivered

Table 1: List of deliverables

6.3 Project Estimation versus Actual Execution

The project was estimated to be completed within a time frame of 87 days with working hours of 3 hours per day.

The distribution of effort estimated during the project planning is as mentioned in the below table-

S.No	Phase Name	Duration	Estimated Start date	Estimated End date	Actual Start date	Actual End Date
1	Requirement Analysis / Requirement Gathering	17	29-Aug-16	15-Sep-16	1-Sep-16	21-Sep-16
2	Design	15	16-Sep-16	30-Sep-16	25-Sep-16	26-Oct-16
3	Implementation	20	1-Oct-16	21-Oct-16	28-Oct-16	25-Nov-16
4	Testing	35	15-Oct-16	15-Nov-16	7-Nov-16	30-Nov-16 (expected)

Table 2: Phases of project development for Whereabouts application
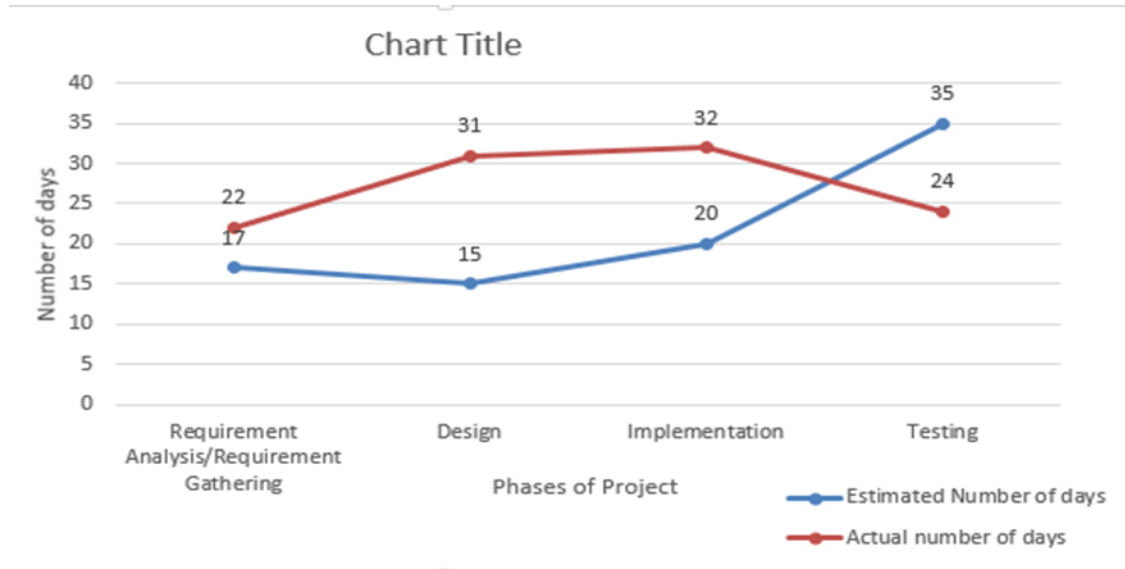

Figure 1: Graph of project effort estimation v/s actual effort

7. Learning through This Project

7.1 Problems Incurred During Course of Project

Learning and implementing the technology used for implementation of the project was a challenge as all the team members were relatively new to the domain. Android Studio usage and Firebase API usage were

a new and exciting challenge. Team collaboration and division of work required some brainstorming efforts as well.

7.2 Alternative approach for Whereabouts given another chance

We would stick to our approach of using Firebase platform for the development of this project. The only change will be in the estimation of time that modules require for development and testing so that the tasks could be completed well within estimated times. The time required to create such a project in future will take much less time as the team is much well gelled with each other and the technology and requirements are also well understood.

7.3 Technical Experience Gained from this Project

The team has gain–ed expertise in following skills-

1. Android application development
2. Firebase real time database usage
3. Cloud messaging
4. Technical documentation

Also each member gained an insight of various phases of software development as each of the team member was involved in various activities in each phase (the same can be observed in the table given below)-

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15	A16	A17	A18	A19	A20	A21	A22
Aakash	*				*	*				*						*	*	*		*		
Kartikeya	*	*	*					*	*				*						*	*		*
Manpreet	*				*		*				*			*	*		*			*	*	
Shubham	*	*		*				*			*	*		*						*	*	*

Fig-Development plan

Activities:
A1 - UI Discussion & Design
A2 - Design & Documentation of Authentication Component
A3 - Development of Authentication Component
A4 - Testing of Authentication Component
A5 - Design & Documentation of Map Rendering Component
A6 - Development of Map Rendering Component
A7 - Testing of Map Rendering Component
A8 - Design & Documentation of Instant Messaging Component
A9 - Development of Instant Messaging Component
A10 - Testing of Instant Messaging Component
A11 - Design & Documentation of Peer Management Component
A12 - Development of Peer Management Component

A13 - Testing of Peer Management Component
A14 - Design & Documentation of Drop Message Component
A15 - Development of Drop Message Component
A16 - Testing of Drop Message Component
A17 - Design & Documentation of Track me Component
A18 - Development of Track me Component
A19 - Testing of Track me Component
A20 - Integration Testing
A21 - System testing
A22 - Product Documentation

7.4 Team Building Experience Gained from this Project

The project worked wonders in bringing out the best in each of the team members. Each of the team members demonstrated the qualities of leadership, time management, quick learning ability and helping each other to reach a feasible solution to all the problems at hand. The members learned to own their work and take responsibility of assigned tasks and made their best efforts to get the tasks completed.

7.5 Management lessons learnt

The team members learnt the importance of time management. Clear division of work and responsibilities led to a well-defined action plan for each member. Also it was observed that working together as a team helps in broadening the perspective of looking at a problem and may lead to simple and elegant solutions to tough looking problem. One of the most important lesson learnt is that an environment of harmony and bonding within team mates is essential to have a great workplace.

8. Future Enhancements

8.1 Enhancement With Respect To Product

The later releases would see the addition of feature "Track me" that could not be implemented due to time constraints. Details of this feature could be found in the design document attached in the following section. Based on user feedback new features may be added in subsequent releases.

8.2 Enhancement With Respect To Course

The course can be improved by having more interactions with the mentor and active tracking of progress of the project. Also feasibility analysis of the project must be done more rigorously in consultation with the mentor and team members.

Attachments:-

Sl. No	Document Type	Attached Document
1	SRS of Whereabouts	SSG562_G02_RS.pdf
2	Design of Whereabouts	SSG562_G02_DS.pdf

Table 3: Reference Documents