1 Introduction

1.1 Overview (Executive Summary)

Our project is the design and creation of a mobile application which combines a mobile phone and a smartwatch in order to allow communication between group members by way of a server. The phone application will allow group members to fully send and receive messages amongst themselves. The smartwatch will have a more simplistic design to fit its technological limitations. This kind of service is becoming increasingly popular as people try to find ways to remotely communicate and perform meetings. Its availability to anyone allows for a large amount of potential growth. The project will be completed over the course of the next month.

1.2 Definitions and Acronyms

UI - User Interface
IDE - Integrated Development Environment
QA - Quality Assurance

2 Management Structure

2.1 Project Lifecycle

Requirements

- Allow members of the group to communicate
- Communication between a smartwatch and mobile phone
- Mass replies via phone
- Quick replies from templates via smartwatch
- Users will be allowed to group themselves from the phone
- Users will be notified of messages

Design

- Keep the Smartwatch UI simplified

- Phone will be the main form of user communication

Implementation

- June 9: Implementation begins
    - basic functionality will be added first
        - UI for both the phone and watch
        - Phone Replies, connect the phone to the watch
        - Connection to the server
- Add watch based user responses
- After initial implementation is done, add variety to phone capabilities

Verification

- June 23: latest to begin QA and testing
- Testing phase will end on June 25

Maintenance

- Maintenance will be at a minimal level

2.2 Roles and Responsibilities

Role	Responsibility
Project Manager	Anastasiya Cotton
Planning and Tracking Lead	Steven Mitchell
Requirements Lead	Michael Dunn
Design Lead	Kurt Bonatz
Implementation Lead	Kurt Bonatz
Quality Assurance Lead	Tyler Bell
Development Engineers	Tyler Bell
QA Engineers	Michael Dunn

2.3 Communication

We will be using Slack for communication. This will allow us to get in contact with the group anytime on and off campus. We will have meetings in class.

3 Risk Management

3.1 Risk Identification

Risk	Probability	Severity	Description
Deadlines	Medium	High	Deadlines for certain features are not met potentially setting the entire project back.
Communication	Medium	Medium	Communication will be key in this team project, lack of or delayed communication could negatively impact development.
Failure of testing devices	Low	Medium	If the watch we use for testing fails at some point during the implementation or testing phase, whether from hardware failure or potentially being lost, we would need to acquire a replacement product.
Ability to meet in person	Low	Low	If some group members are unable to attend vital meetings, it may slow down the entire development process
Watch integration	Low	High	There is a chance our module implementation could fail to properly integrate with the watch device.

Note: Probability and Severity can be indicated in levels i.e. Low, Medium, High.

3.2 Mitigation Plan

- Deadlines
    - All members will be held to hard deadlines, with work updates per week to make sure that deadlines can be met. If a deadline cannot be met by a single person, assistance will be provided by other group members. Group members will alert others if the deadline cannot be fulfilled.

- Communication
    - We will be utilizing Slack and Google in order to maintain communication alongside after class meetings if necessary.
- Failure of testing Device
    - If the watch fails, we will likely be using an emulator until we can either find a replacement or, if necessary, the end of the project.
- Ability to meet in person
    - Meetings will be preplanned in according to the schedules of the group members. These will be likely limited to after class sessions.
- Watch integration
    - Consistent testing of our module should alert us of any critical errors within a reasonable amount of time to correct them.

4 Planning and Control

4.1 Milestones

May 19, 2015: Project kickoff meeting
June 2, 2015: Complete requirements
June 4, 2015: Complete design
June 9, 2015: Complete test plan
June 23, 2015: Complete testing
June 25, 2015: Project Delivery

4.2 Work Breakdown Structure

1. Requirements
    a. Create requirements
        i. Detail all requirements and sub requirements
        ii. Complete when all project parameters have been elaborated, but subject to change
2. UI Design
    a. Create mockups
        i. Design all possible interfaces of the smartphone application and smartwatch module
        ii. Complete when all features can be accessed from the UI
    b. Perform UI Testing
3. Implementation
    a. Smartphone App
        i. Authentication

1. User should be able to login and logout of the application

ii. Send messages
1. User should be able to type and send a message to another user

iii. Receive messages
1. User should be able to view messages sent to them from other users

iv. Store messages
1. User should be able to view old messages upon closing and re-opening an application

v. Alert when inactive
1. User should receive some notification of received message while app is closed

vi. Group chats
1. Create
    a. User should be able to create a group
2. Join
    a. User should be able to join a group
3. Leave
    a. User should be able to leave a group
4. Switch
    a. User should be able to switch groups they are viewing

vii. Quick Reply
1. User should be able to customize the quick replies that appear on the smartwatch module

b. Smartwatch Module

i. Notifications
1. User should be notified on watch when message received to smartphone app

ii. Read recent messages
1. User should be able to read recent messages from users or groups

iii. Quick reply
1. User should be able to reply to a message by selecting from a list of pre-written responses

iv. Select chat/group
1. User should be able to select which chat or group messages they are viewing

4. Testing

a. Thorough testing of all smartphone features

i. All smartphone features should be tested for bugs, efficient implementation, and integration with other features (including smartwatch features)

b. Thorough testing of all smartwatch features

i. All smartwatch features should be tested for bugs, efficient implementation, and integration with other features (including smartphone features)

4.3 Schedule

Team project is expected to take 1 month and 2 weeks till launch of the application. The following is a high level schedule of some significant milestones for this project:

May 12, 2015: Initiate Project
May 19, 2015: Project kickoff meeting
June 2, 2015: Complete requirements
June 4, 2015: Complete design
June 9, 2015: Complete test plan
June 23, 2015: Complete testing
June 25, 2015: Project Delivery

Upon approval of this project, the assigned project team will create a detailed schedule to include all tasks and deliverables.

4.4 Tracking and Control

We are planning to utilize Git for our code repository and version control. Git allows for each contributor to maintain an individual branch separate from the primary source code, so that features can be implemented individually and merged into our primary source code once it has been thoroughly reviewed and tested.

We will be assigning tasks to team members via Github Issues. Team members should detail all of their individual accomplishments (features implemented, bugs fixed, hours worked, etc.) in Git commits. This allows for not only the tracking lead to have all progress reported in one place, but allows for anyone on the team to quickly see what has recently be done. Currently, Steven Mitchell is the planning and tracking lead and will be keeping track of all progress and hours worked.

5 Technologies to be used

Since our module focuses on Android and Android Wear, we will be using Android Studio as our main IDE given its sole purpose is Android development. Our team is currently working across all main operating systems OS X, Windows, and Linux with OS X being the majority. Most of the team's development will be done on laptops. Other technologies expected to be

used are several third party libraries that could make things such as our network calls or database easier to manage. Finally, we will be testing on personal Android devices, and one Android Wear device.