University of Glasgow | School of Computing Science

Smartwatch HCI Design

Robert McAlpine

School of Computing Science
Sir Alwyn Williams Building
University of Glasgow
G12 8QQ

Level 4 Project — March 18, 2016

Abstract

The Notification History application is a system developed for Android Wear smartwatches. The main objective of this project is to provide an application for Android Wear smartwatches that can be used to review notifications. Currently, the Android Wear OS does not maintain notifications, once the user dismisses a notification from the smartwatch, it cannot be retrieved. This issue is compounded by the fact that Android Wear provides no way for the user to determine how many new or unread notifications there are on the smartwatch. This may sound like trivial problem, however, this lack of functionality detracts from the usefulness of a smartwatch as a notification centre. In order to tackle both of these issues, a special Notification watch face was developed for Android Wear, which allows the user to determine the number of new notifications at a glance. By combining the Notification History wear application and watch face, this project aims to overhaul the current Android Wear notification system by giving users a novel way of interacting with their smartwatch while providing much needed functionality to Android Wear.

Education Use Consent

I hereby give my permission for this project to be shown to other University of Glasgow students and to be distributed in an electronic format. **Please note that you are under no obligation to sign this declaration, but doing so would help future students.**

Name: _____ Signature: _____

Contents

1 Introduction **1**

1.1 Context 1

1.2 Problem Statement and Motivation 1

1.3 Objectives of the Project 2

1.4 Structure of Report 2

1.5 Conclusion of Chapter 3

2 Literature Survey **4**

2.1 Introduction 4

2.2 Background 4

2.3 Literature Survey 5

2.4 Literature Review: Understanding Everyday Uses of Consumer Smart Watches 5

2.5 Conclusion of Chapter 6

3 Requirements Specification **7**

3.1 Introduction 7

3.2 Requirements Elicitation 7

3.3 Use Case Diagram 8

3.4 Functional Requirements 9

3.5 Non-Functional Requirements 10

3.6 Conclusion of Chapter 11

4 Design **12**

4.1 Introduction . . . 12

4.2 Software Design . . . 12

4.2.1 Mobile application . . . 12

4.2.2 Wear Application . . . 13

4.3 System Architecture . . . 14

4.3.1 Mobile Application System Architecture . . . 14

4.3.2 Wear Application System Architecture . . . 14

4.4 Interface Design . . . 15

4.5 Conclusion of the Chapter . . . 15

5 Implementation **16**

5.1 Introduction . . . 16

5.2 Software Development Method Used . . . 16

5.3 Technology Used . . . 17

5.3.1 External Libraries . . . 17

5.4 Implementation . . . 18

5.4.1 Mobile Application . . . 18

5.4.2 Wear Application . . . 19

5.5 Challenges . . . 19

5.6 Testing . . . 20

5.7 Conclusion to Chapter . . . 20

6 Evaluation **21**

6.1 Introduction . . . 21

6.2 System Testing . . . 21

6.2.1 Unit Testing . . . 21

6.2.2 Interface Testing . . . 21

6.2.3 Integration Testing . . . 22

6.3 User Evaluation . . . 22

6.3.1 Aim of the evaluation . . . 22
6.3.2 Procedure to recruit subjects . . . 22
6.3.3 Demographic Information . . . 23
6.3.4 Evaluation Framework . . . 23
6.4 Conclusion to Chapter . . . 24

7 Results **25**
7.1 Introduction . . . 25
7.2 Research Question 1 . . . 25
7.3 Research Question 2 . . . 26
7.4 Research Question 3 . . . 28
7.4.1 Intuitiveness . . . 28
7.4.2 Usefulness . . . 29
7.4.3 Information . . . 29
7.4.4 Aesthetics . . . 29
7.4.5 Expectation . . . 30
7.5 Conclusion to Chapter . . . 30

8 Conclusion **31**
8.1 Introduction . . . 31
8.2 Conclusion . . . 31
8.3 Future Directions . . . 32
8.4 Learning Outcomes . . . 32

Appendices **35**

A Consent Form **36**

B Evaluation Questionnaire **40**

Chapter 1

Introduction

1.1 Context

In recent years, there has been a drive to produce wearable devices, allowing users to get information while freeing up their fingertips from use of their smartphones. This has led to the production of smartwatches, whose purpose spreads beyond simply timekeeping. A smartwatchs basic purpose is to provide access to user specific information at a glance (e.g. user notifications, distance travelled, heart rate readings). The smartwatch notifies the user of new information (often received from the smartphone it is connected to via Bluetooth), generally taking the form of a notification. Notifications are messages or alerts from an application used to update the user of new information as it becomes available, for instance, a notification from a messaging application could be used to alert the user to the fact that a new message has arrived. Smartwatches also provide a source of information gathering [1]. Many smartwatches have a vast array of sensors, the most prominent ones are heart-rate sensors, pedometers, accelerometers, and even ambient light sensors and cameras [1]. Smartwatch applications make use of these sensors to monitor user activity and provide feedback. For all intends and purposes, the smartwatch is essentially a wrist computer, for which there is a growing number of choices in the market today.

The four main competing smartwatch operating systems (as of Jan 1st 2016) are: Android Wear (Google); Watch OS (Apple); Pebble OS (Pebble); and Tizen OS (An open-source operating system produced by Tizen). Android makes up 82.8% of the worldwide smartphone operating system market share [2] and while Android Wear (Androids smartwatch operating system) is relatively new (released March 2014 [3]), as the number of Android Wear vendors grows and smartwatches become more popular, it is assumed that Android Wear will too, control the majority of the market share [4]. For this project, the Android Wear operating system was the obvious choice to investigate HCI smartwatch design, due to its popularity and impact on the smartwatch market.

1.2 Problem Statement and Motivation

Android Wear currently lacks a way to review notifications. It is a common issue that users face when trying to retrieve their old notifications from their smartwatch, they simply, cannot [5]. Once a notification is dismissed, it cannot be retrieved using the smartwatch alone. This issue is compounded by the fact that Android Wear provides no way for the user to determine how many new or unread notifications there are.

The primary aim of the project is to develop a notification application for smartwatches running the Android Wear operating system, which can be used to receive and review notifications on the fly. Additionally, by

developing a watch face with an embedded notification counter used to display the number of new or unread notifications the user could then determine at a glance, how many new notifications there are.

The delivery of notifications is another issue faced by Android Wear users. Currently, when notifications arrive on Android Wear smartwatches, they are presented to the user in the form of a preview card (Figure 1). These preview cards, cover the majority of the smartwatch screen and demand attention. Smartwatches should remain discreet and not draw unnecessary attention to themselves [6], and so, by developing an Android Wear notification application and watch face, it would negate the need for a preview card, as new notifications would simply be alerted to the user by the notification counter on the watch face, maintaining discretion and appeal of a watch face.

Figure 1.1: Preview Card

By combining the watch face to the notification application, the project provides a novel approach to completely overhauling the notification service in Android Wear for the better, whilst adding much needed functionality. The user has the freedom of being notified about new notifications discreetly by glancing at their watch face, while when given the time, will have the ability to review notifications and inspect their contents when the appropriate opportunity presents itself.

1.3 Objectives of the Project

Primary Objectives:

1. Develop a notification application for Android Wear smartwatches that will allow users to review notifications.

Secondary Objectives:

1. Develop a watch face that will allow the user to determine the number of new/unread notifications.
2. Alert the user to new notifications with discretion.

1.4 Structure of Report

The remainder of this report is structured into the following chapters **Chapter 2 Background** This chapter aims to give the reader context for the project providing a literature review within the area of study.

Chapter 3 - **Requirements**

This chapter aims to describe the elicitation process used, and provide an overview of the requirements identified for the project.

Chapter 4 - **Design**

This chapter aims to present the key design decisions for the project to satisfy the requirements identified in chapter 3.

Chapter 5 - **Implementation**

This chapter aims to provide an overview of the technologies and architecture used to develop the project.

Chapter 6 - **Evaluation**

The chapter aims to describe the evaluation strategy adopted to ensure proper functionality, interface consistency, and usability of the system.

Chapter 7 - **Results**

In this chapter, the results of the research questions formulated in Chapter 6 are analysed and discussed in depth.

Chapter 8 - **Conclusion**

This final chapter forms a summary for the dissertation, followed by possible future direction for the project, and learning outcomes.

1.5 Conclusion of Chapter

It is important to understand that smartwatches, at the end of the day, are still viewed as watches. This makes it imperative that any user interaction should be minimal. The notification watch face does simply that, it allows the user to be alerted to new notifications by glancing at the watch face. The notification application itself stores notifications and allows the user to review them at any time, giving the user greater freedom to pick and choose when it is suitable to review their notifications. These two features combined provide a lucrative project to overhaul how Android Wear users interact with their smartwatches. The design choices were greatly influenced on the current reaction to smartwatches on the consumer market. The Literature Review will be used to go into detail showing how the project addresses the issues of current smartwatch users.

Chapter 2

Literature Survey

2.1 Introduction

This chapter will give an overview of the existing Android Wear design principles and give a review of academic literature in relation to smartwatch interface development.

2.2 Background

To agree on user interface requirements focused specifically with Android Wear applications, a study aiming to identify Android Wear design principles is required. Because of this, the decision was made to adopt the official Android Wear rules for interface design:

- **Focus on not stopping the user** [7]

  A typical wrist watch does not hold the users attention for any longer than a second or two. Since Android Wear applications reside on the smartwatch, the idea here is that smartwatch applications should not hold the users attention for more than a short period of time to allow the user to carry on with some other task.

- **Design for big gestures** [7]

  Precision gestures should not be used as way of interacting with the application. Big gestures (such as swiping) are preferred. This is because smartwatches are designed to be used when the user is on the move. Simple, big gestures, make it easier for the user to interact with the application no matter what the circumstance.

- **Do one thing, really fast** [7]

  Users may only engage in the application for short periods of time, but users may want to use the application frequently. It is then necessary to ensure that the application is fast at handling tasks. Splitting an application up into simple tasks is key.

- **Design for the corner of the eye** [7]

  Wear applications should be designed to be glanceable. They should not pull the user away from another task and force further interaction with the application.

- **Dont be a constant shoulder tapper** [7]

  Applications should not vibrate too often. Since smartwatches are attached to the user physically, the vibrate function of a smartwatch should be used infrequently to avoid constantly interrupting the user.

These interface design principles will be considered in the requirements capture in chapter 3. The Android Wear design heuristics will be followed throughout the interface design in chapter 4.

2.3 Literature Survey

Since wearable smart devices are still relatively state-of-the-art. There are a very limited number of studies investigating smartwatch usability. The majority of which, are focused on textual input interfaces for smartwatches. As part of the literature review, it then became necessary to read Android Wear reviews in a bid to find more specific information addressing interface issues of Android Wear.

Reference	Contribution	Limitations	Comments
Understanding Everyday Uses of Consumer Smart Watches. Schirra, S. and Bentley, F R. (2015)	Study to explore current smartwatch use.	Very small sample group. Explores smartwatches using different operating systems besides Android Wear.	Gave insight into the importance of style qualities of smartwatches. Found that users preferred smartwatches to be discreet/unseen devices.
Is Touch-based Text Input Practical for a Smartwatch? (Chaparro. B S et al, 2015)	Investigated different textual input methods for smartwatches and discussed their practicality.	Solely aimed at evaluating textual input performance.	Highlighted the importance of a practical user interface.
WatchMe: A Novel Input Method Combining a Smartwatch and Bimanual Interaction. (Van Vlaenderen, W. et al, 2015)	Investigated the WatchMe smartwatch application and its practicality for use as a textual input method.	Purely focussed on textual input methods.	Makes use of a camera as an input device – highlighting other methods for user interaction with smartwatches.
SplitBoard: A Simple Split Soft Keyboard for wristwatch-sized Touch Screens. (Hong, J et al. 2015)	Compares different keyboard layout designs for use on smartwatch displays.	Purely focussed on evaluating keyboard layout design.	Recommends the splitboard keyboard for textual input on smartwatches over other keyboard designs.
7 Things I Hate About Android Wear (Fedewa, J. 2014)	Reviews Android Wear limitations.	Some issues stated are out of date, and have since been solved by Android.	Highlights that Android Wear users cannot review notifications from their smartwatch.

Table 2.1: Reviewed Literature

Table 1 summarises the literature reviewed by stating the literatures contribution to the scientific community, limitations of the literature in relation to this project, and any comments on the literature that impact this project.

2.4 Literature Review: Understanding Everyday Uses of Consumer Smart Watches

Currently the perception of a smartwatch is that it should primarily still function, as a watch. Style is a major concern when selecting a smartwatch. This conveys the idea that users would like their smartwatch to remain

as watch-like as possible at all times. This is further highlighted when users stated they actively hid their smartwatches while at work [7] , or removed the watch from their wrist entirely if they felt it would draw too much attention to itself [7], or even swapped their smartwatch for their traditional wristwatch if it did not suit the social occasion [7]. When developing the user interface for the Android Wear application, it is necessary to take this into account. Users must be alerted to the arrival of notifications in a discreet manner, so that anyone in close vicinity to the user will not notice the notification alert.

2.5 Conclusion of Chapter

In order to develop a usable, intuitive wear application, it is necessary to follow the heuristics for developing Android Wear applications. By following these rules of thumb, and taking into account the user feedback gathered from the usability studies in the literature review, requirements for the project can then be established. The next chapter describes this process and states the requirements and their associated priorities.

Chapter 3

Requirements Specification

3.1 Introduction

The Requirements Specification chapter will describe the requirements gathering process used to collate the requirements for the project. The project is described in terms of functional and non-functional requirements. The requirements are then broken down into use cases and their associated priorities for development using the MoSCoW approach.

3.2 Requirements Elicitation

Initially, the main focus was to identify potential limitations of smartwatches using the Android Wear operating system. In order to do this, time was spent with a Motorola Moto 360 smartwatch. The requirements were gathered using the smartwatch, reviewing case studies [5] [6] [7] [8] [9], and online reviews [5] [10].

Once literature survey had been reviewed (chapter 2), discussions were held with the project supervisor to establish the scope of the project. The creation of use cases was used to form a basis for which the requirements of the project could be made. The project requirements and their associated priorities were then finalised with the project supervisor based upon the project objectives.

In order to understand the requirements of the project as a whole, it is necessary to split the system into two parts: the mobile application that runs on the mobile device and the wear application which runs on the smartwatch. The reason for this is because all Android Wear applications must have a companion mobile application to be published to the Google Play store [11]. The following use case diagram was used to establish the requirements.

3.3 Use Case Diagram

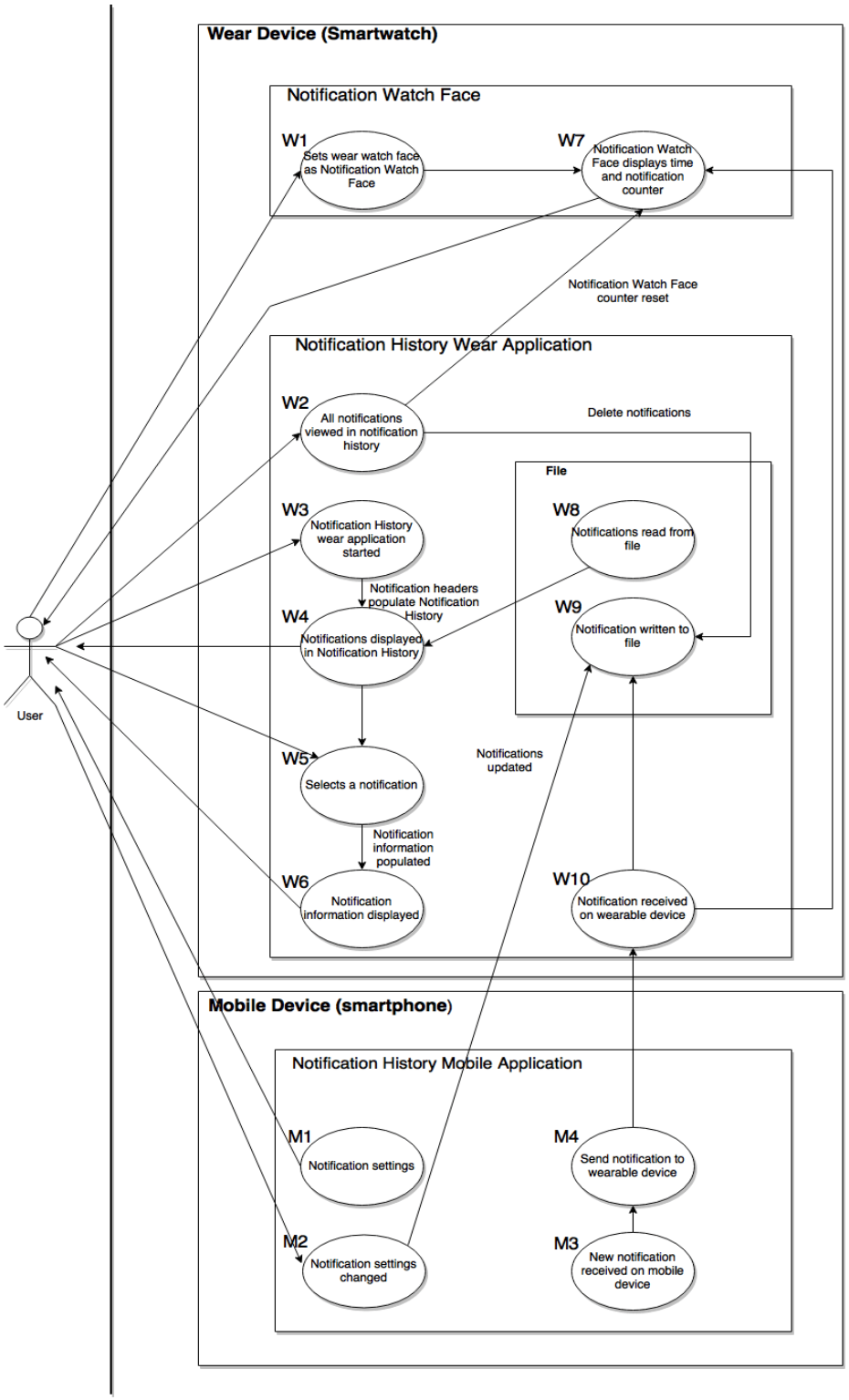

Figure 3.1: Use Case Diagram

Figure 2 is the use case diagram developed for the system. The system is split into two parts: the smartwatch, and the smartphone for which it is connected to. The use cases were used to form the functional and non-functional requirements of the system, together with the literature review, and discussions with the project supervisor.

3.4 Functional Requirements

The functional requirements are used to illustrate the expected system behavior. The functional requirements can then be used to assess the developed system to ensure it meets the stated requirements. Each functional requirement was classified according to the MoSCoW technique, which groups requirements into four distinct groups determined by its associated priority: Must Have, Should Have, Could Have and Would Like to Have.

MUST HAVE			
Requirement Id	**Requirement name**	**Brief description of the requirement**	**Which use-case the requirement will map to**
MH1	Notification history	Must display notifications in reverse chronological order in Android Wear device.	W3: Notifications History wear application started W4: Notifications displayed in notification history
MH2	Notification information	Must be able to view the details of any notification received in Android Wear device.	W6: Notification information displayed
MH3	Display number of unread/new notifications.	Watch face must display this information.	W7: Watch face displays time and notification counter
MH4	Reset notification counter when notifications have been accessed by the user	When all new/unread notifications have been accessed by the user, the notification counter on the watch face must be reset.	W2: All notifications viewed in notification history
MH5	New notification received on mobile device	When the mobile device receives a notification it must send it to the wear device.	M3: New notification received on mobile device M4: Send notification to wearable device
MH6	New notification received on wear device	Wear device must receive notifications as they arrive – update the notification counter.	W10: Notification received on wearable device
MH7	Display notification history settings	The user must be able to view the current notification history settings.	M1: Notification settings
MH8	Notifications must be maintainable	Notifications must be stored by the application so they can be accessed at all times from the smartwatch.	W8: Notification written to file W9: Notification read from file

Table 3.1: Must Have Requirements

Must Have requirements are fundamental for the project to be a success.

SHOULD HAVE			
Requirement Id	**Requirement name**	**Brief description of the requirement**	**Which use-case the requirement will map to**
SH1	Choice to use the Notification Watch Face	The user should have the option to use the Notification Watch Face for the Android Wear device.	W1: Sets wear device watch face as Notification Watch Face
SH2	Changing the settings for the notification history from the mobile device	The user should be able to change the notification history settings from the mobile device.	M2: Notification settings changed
SH3	Notifications displayed in reverse chronological order	The user should view the newest notification first, and oldest notification last.	W4: Notifications displayed in notification history

Table 3.2: Should Have Requirements

Should Have requirements are not necessary, but should be completed if time permits. All Should Have requirements were completed in this project.

COULD HAVE			
Requirement Id	**Requirement name**	**Brief description of the requirement**	**Which use-case the requirement will map to**
CH1	Notifications pulled from mobile device	On launching the wear application, pull new notifications from mobile device if they have not been accessed or read by the user.	W9: Notification written to file

Table 3.3: Could Have Requirements

Could Have requirement is desirable but not necessary.

WOULD LIKE TO HAVE			
Requirement Id	**Requirement name**	**Brief description of the requirement**	**Which use-case the requirement will map to**
WLTH1	Multiple Notification Watch Faces	Would like to allow the user to have a choice between multiple watch faces with the notification watch face functionality.	W1: Sets wear device watch face as Notification Watch Face

Table 3.4: Would Like To Have Requirements

Would Like To Have requirement represents future work and do not affect functionality of the system at the time of release.

3.5 Non-Functional Requirements

Non-functional requirements were established through discussion with the project supervisor.

Requirement Id	Requirement name	Brief description of the requirement
NFR1	Stand alone functionality	Notification History Wear Application must be able to function without the use of the Notification Watch Face

Table 3.5: Non-Functional Requirements

The non-functional requirement specifies system criteria rather than specified system behavior.

3.6 Conclusion of Chapter

The Notification History Wear Application and Notification Watch Face are the two major aspects of the system. The use cases helped to identify the requirements and align them to their associated priorities. This is necessary as developing a functioning prototype is of the biggest concern over the timescale of the project. By following the use case diagram, requirements with priorities, it is possible to design software modules that reflects the functional and non-functional requirements.

Chapter 4

Design

4.1 Introduction

This chapter aims to present the key design modules made to accommodate both the Android wear and mobile requirements described in the previous chapter. The system architecture will be presented and finally, the heuristics followed to design the user interface will be described in detail.

4.2 Software Design

This section will describe the modules of the system and how each of them mapped to the requirements to ensure that the design is reasonable.

4.2.1 Mobile application

Name of Design Module	Description	Requirements The Module Maps To
NotificationService	Service to listen for new notifications and broadcast when new notification occurs	MH5
SendToDataLayerThread	Asynchronous thread used to send notification data to wearable device	MH5
MainActivity	Receiver for new notification and starting SendToDataLayerThread to send the notification to wear device and must allow user to manage notification history settings	MH5 MH7 SH2

Table 4.1: Mobile Application Architecture

Table 4.1 Design modules refer to system architecture model described in Figure 4.1

4.2.2 Wear Application

Name of Design Module	Description	Requirements The Module Maps To
WearableNotificationService	Service to listen for new notifications	MH6
WearableAdapter	User interface element responsible for displaying notification icon and header	MH1
NotificationWatchFace	Notification watch face customised with a notification counter to display unread/new notifications	SH1 MH3 MH4 WLTH1
WearMainActivity	Is responsible for the creation of notification objects, writing and reading them to file, and updating the user interface.	MH1 MH4 MH8 SH3 CH1
WearNotificationActivity	Is responsible for displaying individual notification contents	MH2

Table 4.2: Wear Application Architecture

Table 4.2 Design modules refer to system architecture model described in Figure 4.1

4.3 System Architecture

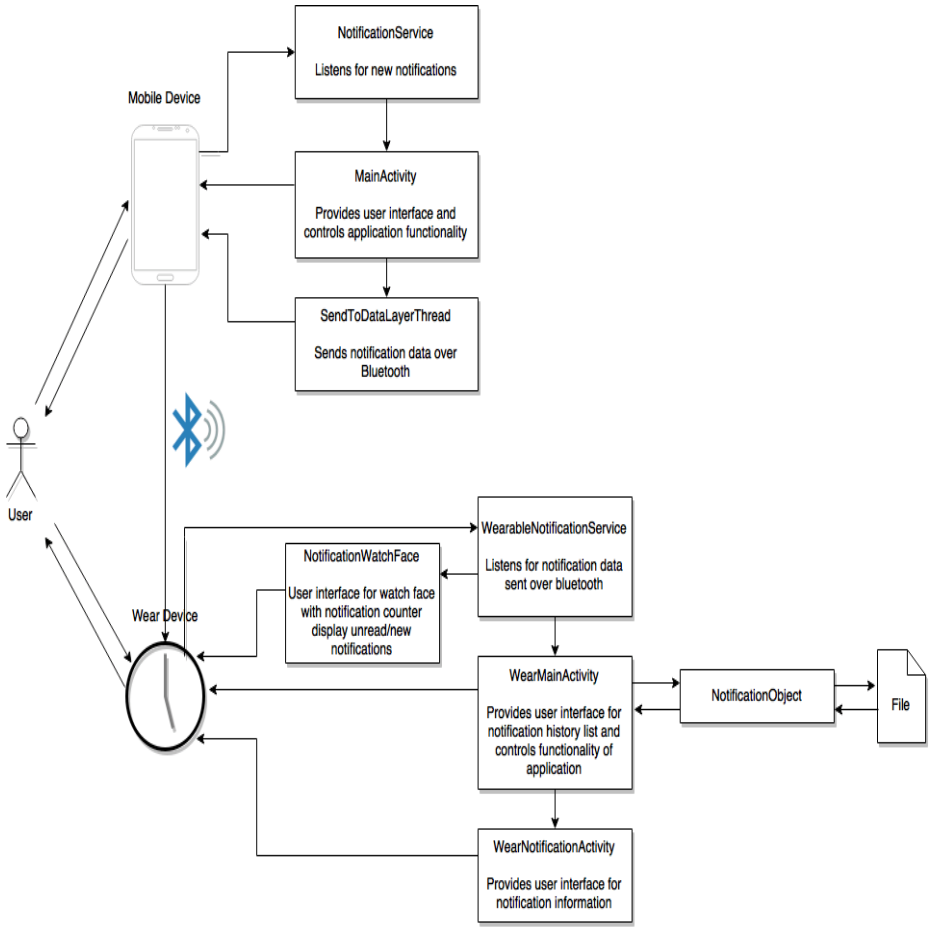

Figure 4.1: System Architecture

The system architecture gives a visual representation of how the modules of the system described in Table 4.1 and 4.2 are used together to complete the software design.

4.3.1 Mobile Application System Architecture

The mobile application makes use of a service. The NotificationService (Figure 2) listens for any new notifications that are received on the mobile device and broadcasts them to the MainActivity. The MainActivity acts as a receiver for notifications broadcasted by the NotificationService. It is responsible for starting the asynchronous thread (Figure 2, SendToDataLayerThread) to allow the notification data to be sent over Bluetooth to the wear device. The user interface for the mobile application is also supplied by the MainActivity, which allows the user to modify the settings for the Notification History wear application.

4.3.2 Wear Application System Architecture

The wear application also makes use of a service, mimicking the mobile applications design. The WearableNotificationService (Figure 2) is used to listen for notification data sent over Bluetooth to the wear device. This service broadcasts any notifications received. When this happens, the NotificationWatchFaces notification counter is

updated (if the NotificationWatchFace has been set). The WearMainActivity packages the notification data into NotificationObjects and writes them to file as they are received. It then maintains the notification history by reading notifications from the file and then displaying them. The WearNotificationActivity allows the user to inspect each notification and read their contents in full.

4.4 Interface Design

The official Android development guidelines were followed to ensure suitable user interfaces were developed for the Notification History wear and mobile applications [12] [13] [14] [15] and watch face [16].

Heuristic	How Heuristic Was Incorporated Into The Design	Design Module The Interface Maps To
H1: Informative list [15]	WearableListView user interface element makes it possible to display the notification history in a list format suitable for wear devices.	WearMainActivity (Wear Application)
H2: Design for big gestures [16]	WearMainActivity provides only tap and swipe gestures.	WearMainActivity (Wear Application)
H3: Design for big gestures [16]	WearNotificationActivity provides only swipe gesture.	WearNotificationActivity (Wear Application)
H4: Minimal watch faces. [16]	Notification Watch Face designed as an analogue watch face.	NotificationWatchFace (Wear Application)
H5: Make it personal/Use distinct icons [14]	Notification icons, headings and contents used within the wear application's WearableListView (H1).	WearMainActivity (Wear Application) WearNotificationActivity (Wear Application)
H6: Integrate data. [16]	The Notification Watch Face was designed to incorporate a notification counter to replace the traditional date/time counter of an analogue watch face.	NotificationWatchFace (Wear Application)
H7: Integrate data [16]	Once notifications have been viewed from the Notification History Wear Application, the watch face's notification counter must be reset.	WearMainActivity (Wear Application) NotificationWatchFace (Wear Application)
H8: Glanceable [12]	The user can determine the number of new notifications by glancing at the watch face minimising user interaction.	NotificationWatchFace (Wear Application)
H9: Select appropriate settings [13]	Settings were used to capture a user preference – therefore, to be accessed infrequently.	MainActivity (Mobile Application)

Figure 4.2: User Interface Heuristics

4.5 Conclusion of the Chapter

Using listeners for specific data artefacts and broadcasting them to receivers to initiate action, makes the system easy to read and understand. By following the system design, architecture and heuristics, the system can now be implemented in a modular fashion, allowing for expansion, where necessary for future work. This makes the project scalable and allows all requirements to be encompassed within the design.

Chapter 5

Implementation

5.1 Introduction

The aim of the chapter is to provide an overview of the technologies used to develop the wear and mobile applications. The implementation is described in detail followed by challenges faced and finally reasoning behind the decisions made throughout implementation are briefly summarised.

5.2 Software Development Method Used

An iterative waterfall software development method was chosen because the requirements were fixed and objectives agreed upon with the project supervisor during the requirements capture. This helped to follow a formal procedure for implementing the software design.

5.3 Technology Used

Technology	Basic Purpose	Reason for Choosing	Alternative Options
Java (software)	Platform to develop Android applications.	Most popular language to develop Android applications.	JavaScript Python
Android Software Development Kit (SDK) Tools: Android 4.3.1 (software)	Provides tools for implementing Android Applications.	First development kit to feature Android Wear development tools and compatibility.	Any Android (SDK) released after Android 4.3.1
Android Studio IDE (software)	Provides the environment for developing Android applications	The official integrated development environment (IDE) for developing Android applications. Provides support for JUnit and Android tests.	Eclipse IDE with the Android Developer Kit Tools plugin
Motorola Moto 360 (hardware)	Smartwatch running Android Wear 1.3	Easy to program	Any manufacturer of smartwatches using the Android Wear operating system
Motorola Moto E (hardware)	Smartphone with Android OS 5.0.2	Compatible with the Android Wear operating system. Allows the creation of Android companion applications to be used in conjunction with a smartwatch application.	Any manufacturer of smartwatches using the Android operating system 4.3.1 (JellyBean) or later.
Git	Revision control system for software development	Easy to distribute and share code. Well documented. Free to download and use.	Concurrent Versions System (CVS). Apache Subversion (SVN). Mercurial

Table 5.1: Technologies

5.3.1 External Libraries

- **Android Support Library (v7)**

  Provides various user interface elements required for developing Android applications.

- **Google Play Services API**

  Required to sync and send data between the wear and mobile device.

- **Android Wearable Support Library**

  Provides user interface elements used for developing Android Wear applications.

5.4 Implementation

5.4.1 Mobile Application

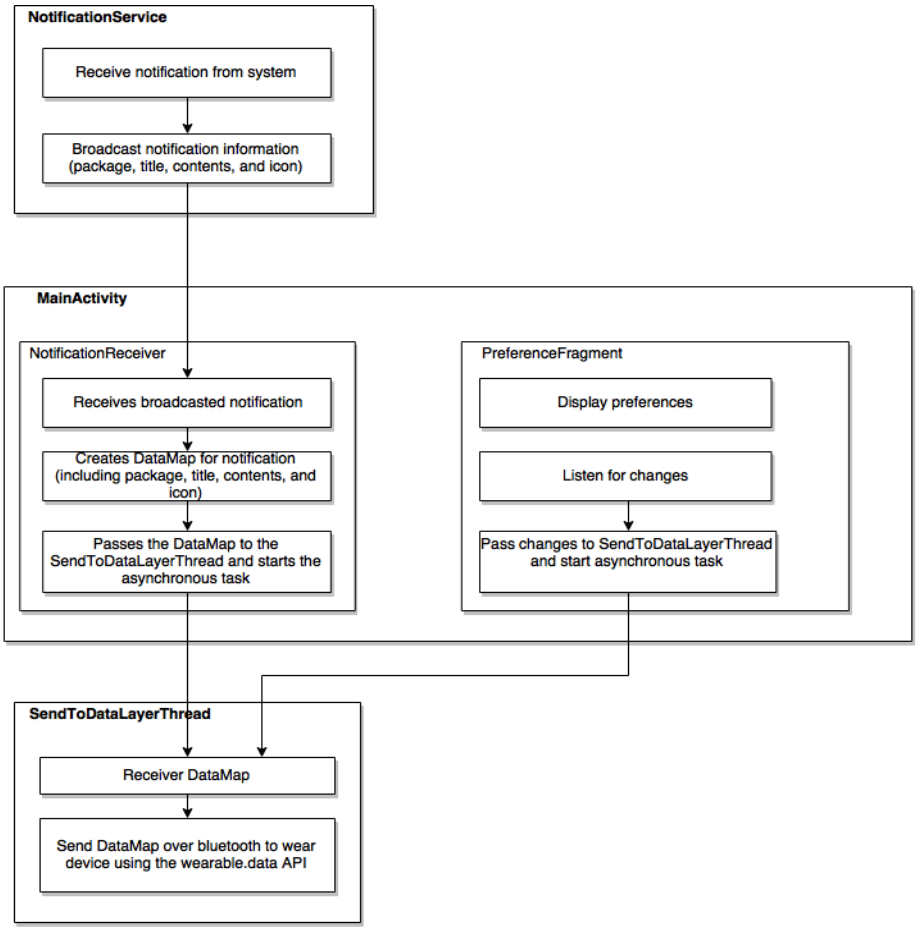

Figure 5.1: Mobile Application Implementation

The main function of the mobile application (Figure 3) is to listen for any notifications that occur on the mobile device (NotificationService) and notify the smartwatchs wear notification application (SendToDataLayerThread). It is also responsible for giving the user access to the settings for the wear application (PreferenceFragment). Any new notifications or changes to the settings are sent to the smartwatch over Bluetooth.

5.4.2 Wear Application

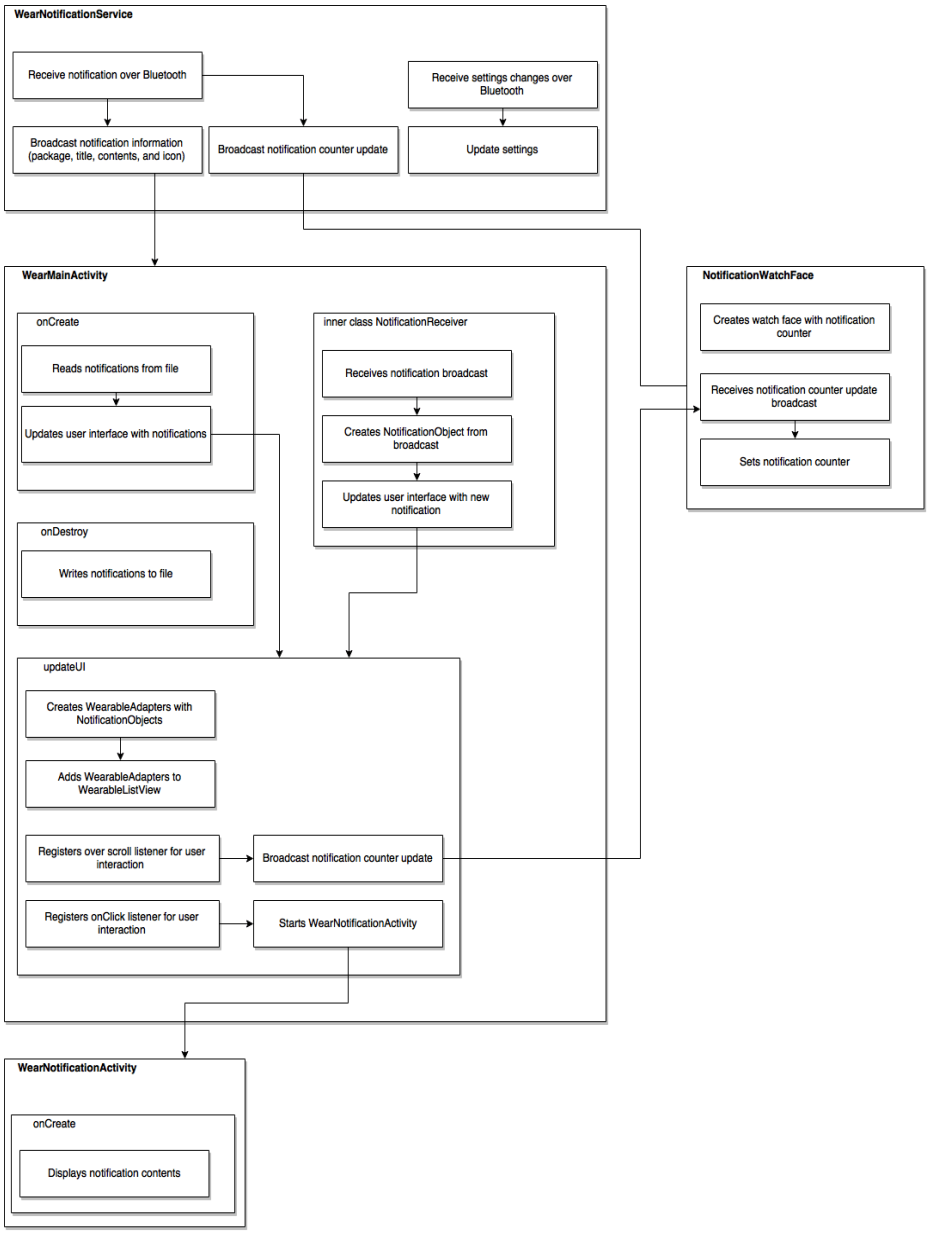

Figure 5.2: Wear Application Implementation

The wear application (Figure 4) has two primary functions: to update the notification history (WearMainActivity) by displaying notification information as it received on the smartwatch, and, to update the notification watch face (NotificationWatchFace) that displays the number of new/unread notifications. Any settings update received also needs to be applied and manifested in the notification history. The WearNotificationActivity allows the user to inspect the contents of each notification.

5.5 Challenges

The first challenge, before starting the project, was getting accustomed to developing for a smartwatch using Android Wear tools and libraries. Smartwatch development is a relatively new venture, however, Android de-

velopment in particular is incredibly well documented. Android provides many sample projects, in Java, aimed to provide new developers experience with all aspects of Android development. Even as an experience Java developer, due to the steep learning curve, it was a huge benefit to have the resources available to overcome implementation issues as they arose, making for little time was spent searching for solutions.

The initial plan for the project was to implement a stand alone Android Wear application that could receive notifications from the mobile device without the use of a mobile companion application. This proved to be one of the main challenges for the project.

Unfortunately, it turned out Android Wear does not support wear applications listening to all incoming notifications, and so, it then became necessary for the companion mobile application carry this functionality out in order to handle notifications and send them to the wear application. A work around for this issue would be make changes to the underlying firmware but due to the time constraints of the project it was not feasible to carry this out, and so, the project was designed and constructed using both wear and mobile applications to complete the functionality.

5.6 Testing

Since the software development followed an iterative waterfall model, as classes and methods were developed, Junit tests were developed in tandem to ensure the functionality aligned with the design of the mobile and wear applications (Chapter 4, 4.2.1 and 4.2.2).

5.7 Conclusion to Chapter

This chapter gave an overview of the functional modules of the application and how they work together to form a functioning system. The comparison of similar technologies allowed for an informed decision to be made regarding the needs of the project, albeit, under the tight constraints of Android development, limiting the options and choices to be made. The next chapter will discuss system testing and the user study conducted to examine the effectiveness and usability of the system.

Chapter 6

Evaluation

6.1 Introduction

The chapter aims to describe the evaluation strategy adopted to ensure proper functionality of the system, and, describes how the usability of the system is evaluated through experiments carried out as part of the user study.

6.2 System Testing

System testing is used to ensure that the system is fit for purpose. This is carried out by verifying the system against the Requirements Specification, so that the system meets both functional (Chapter 3, 3.4) and non-functional requirements (Chapter 3, 3.5).

6.2.1 Unit Testing

As the development of the system progressed, each functional component was thoroughly tested using unit tests. Each time new functionality was added to the system, the unit tests provided a way of finding small logic errors helping to ensure the system was reliable. As the unit testing was carried out right the way through the development lifecycle, errors could be found early and corrected immediately. Extra care was taken over the exceptional scenarios and error handling, ensuring the robustness of the system and proper code coverage could be achieved.

6.2.2 Interface Testing

Interface testing was incredibly important due to the full functionality of the system spreading over both the wear and mobile applications. Extra attention was taken over the interface between the two applications. This was necessary to ensure that notifications sent from the mobile to the wear device were well formed, and that proper error-handling was carried out to ensure consistency and robustness of the system as a whole.

6.2.3 Integration Testing

Once each component had been tested individually, communication between components was then examined extensively. Integration testing revolved around ensuring the components of the system would communicate with each other properly, i.e. ensuring the parameter passing between components of the system were consistent and robust.

Initially integration testing was carried out on smartphone and smartwatch emulators before moving the final runs of the integration testing on physical devices. The reason for using emulators initially was simply due to speed. Deploying the wear application on to the smartwatch and carrying out tests was a slow process as it was carried out over Bluetooth. This did not prove to be a problem as the emulators provided by the Android AVD manager replicate the functionality of the Motorola Moto 360 smartwatch accurately. There were no noticeable differences between emulators integration testing performance and the devices.

6.3 User Evaluation

6.3.1 Aim of the evaluation

The user study aims to evaluate the usability of the user interface and its consistency, as well as the effectiveness of the systems functionality. Furthermore, the study also aims to evaluate whether the system truly addresses Android Wears shortcomings: no notification center, and no way of checking how many notifications have been received.

In order to perform this evaluation, research questions were constructed:

1. Does the Notification History system have an easy to use, consistent user interface, for each aspect of the system?
2. Is the Notification History system suitable for all users?
3. How effective is the system at providing users with a way to review their notifications?

The results of these questions will be discussed in depth in the following chapter.

6.3.2 Procedure to recruit subjects

The system does not require any special technical expertise to use, this is because the system was designed for anyone who uses an Android Wear smartwatch. Familiarity with the Android mobile platform is obviously advantageous, but was not a required prerequisite to those evaluating the system.

Participants were recruited from varying demographic backgrounds primarily through social media and email. Some participants were recruited simply through word of mouth.

A total of 20 participants volunteered to take part in the study, all of which completed the experiment and user survey.

6.3.3 Demographic Information

The subject pool used consisted of participants from different backgrounds and age. All participants were recruited from Glasgow, all of which were residing in UK at the time of the survey. Of the 20 participants, 13 of which were aged 18-25, and the remaining 7 were aged 25-30.

Number of participants	Background
8	Computer Science
3	Business Studies
2	Medicine
2	Sports Science
5	Working Professional

Table 6.1: Demographic Information of Participants

More than half the participants in the survey were not from a Computing Science background, and so, a comprehensive evaluation of the research questions could be investigated. This was invaluable at providing different perspectives of the system, and the user feedback from the survey will be discussed in the following chapter.

6.3.4 Evaluation Framework

A controlled experiment evaluation framework was adopted to evaluate the system. By presenting each participant with the control (factory default settings for the smartwatch and mobile device) and the experimental set up (deployed Notification History mobile and wear applications with notification watch face set on the smartwatch), the effectiveness and usability of the system can then be evaluated against the control. Each participant had an opportunity to explore the features of both the control setup and experimental setup. This was necessary as none of the participants having any previous experience with Android Wear or smartwatches. The experiment therefore takes the form of a within subjects experiment. All participants filled out a participant consent form which can be found in Appendix A.

Stage 1: Control Setup and Exploration

Scenario: Participants were asked to explore the Motorola Moto 360 smartwatch connected to a Motorola Moto E smartphone. Both of which were set to factory default settings. The user was given 10 minutes to freely explore their functionality.

Stage 2: Control Performing Tasks

Scenario: 10 new test notifications were deployed to the smartphone and subsequently sent to the smartwatch. The user was asked to perform five tasks:

1. Determine how many unread notifications there are using the smartwatch.
2. Inspect the contents of the new notifications.
3. Limit the number of notifications the smartwatch can hold at any one time.
4. Review all the notifications using the smartwatch.
5. Delete all notifications

Stage 3: Experimental Setup and Exploration

Scenario: Participants were asked to explore the Motorola Moto 360 smartwatch with the deployed Notification History wear application and watch face. The mobile companion application was also deployed on to the connected Motorola Moto E mobile. The user was given 10 minutes to freely explore its functionality.

Stage 4: Experiment Performing Tasks

Scenario: 10 new test notifications were sent to the mobile device. The user was asked to perform the same tasks:

1. Determine how many unread notifications there are.
2. Inspect the contents of the new notifications.
3. Limit the number of notifications the smartwatch can hold at any one time.
4. Review all the notifications using the smartwatch.
5. Delete all notifications

Stage 5: Survey

On completion of Stage 4, subjects were asked to complete a web based survey. The survey was based upon the standard user interface evaluation questionnaire developed by Chin et al [17]. The survey was broken down into three sections: the Notification History wear application, the Notification History watch face and the companion mobile application. It was necessary to split the survey in this way due to the contrast in user interfaces used for these three aspects of the system. The questionnaire allowed comparisons to be made between the out the box (control) usability and functionality of both the smartwatch and smartphone compared to the new (experimental) functionality that the Notification History system offered. At the end of the survey each participant had the opportunity to suggest other features they would have liked to have seen in the system that were not provided, what aspects of the system they found to be most negative, and which aspects of the system they found to be the most positive. The questionnaire is attached to the report (Appendix B).

The results of the survey were then analysed and will be discussed in depth in the following chapter.

6.4 Conclusion to Chapter

The chapter was used to give an overview of the evaluation methodology used to explore the research questions posed of the system. Using a control experimental setup, comparisons could be made by the participants of the study between factory functionality and the Notification History system for the smartwatch and mobile device. The analysis of the results of the user study will form the basis for the discussion in the next chapter.

Chapter 7

Results

7.1 Introduction

In this chapter, the results of the research questions formulated in Chapter 6 are analysed and discussed in depth.

7.2 Research Question 1

Question: Does the Notification History system have an easy to use, consistent user interface, for each aspect of the system?

In relation to Research Question 1, the survey evaluated the screens of each part of the system separately. Users were asked to rate on a scale of 1 to 10 (1 being lowest, and 10 being highest) to evaluate how readable the screen text was, the organisation of information on screen, and the sequence of screens used while navigating through the different aspects of the system. The mean scores for each of the aspects of the system are shown in figure 7.1.

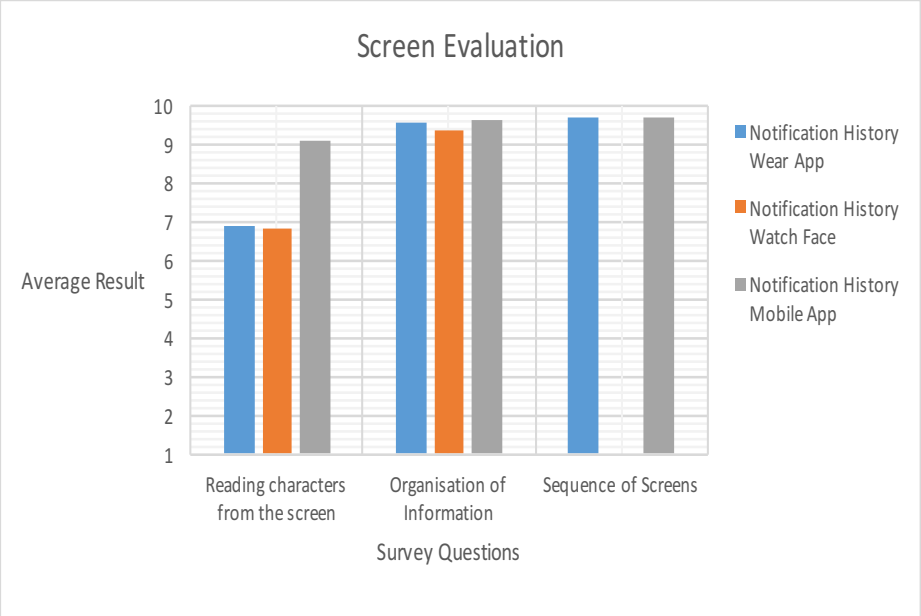

Figure 7.1: Screen Evaluation Chart

Considering the ordinal scale of the data (ratings), by using a repeated measures protocol (i.e. all users were required to answer all questions for the Notification History Wear Application/Watch Face/Mobile application), a non-parametric Friedman test was used to test the statistical significance of the results between each aspect of the system. Note, that the Notification History Watch Face only had a single screen, and so, no Sequence of Screens question was required for this aspect of the system. Because of this, the result for this question could not be used as part of the Friedman test. The results of which show that there exists significant differences between results for reading characters from the screen.

Users found that reading text from the screen of the watch face was moderately difficult for both smartwatch aspects of the system (the Notification History wear application and the Notification History watch face). This is partly due to the size of the smartwatch screen. The Notification History wear app and watch face used a font size of 10sp, whereas, the mobile application used a font size of 14sp. Unfortunately, this is a tradeoff which has to be made due to the small screen size of the smartwatch: it simply is not feasible to increase the font size beyond 10sp without making several adjustments to other user interface to accommodate larger font sizes. In order to improve readability of text within the wear application and watch face, the user interface would need to be redesigned for use of a larger font.

The results for organisation of information and sequence of screens showed no significant statistical differences and so, the high average for both categories (¿9) reaffirms that navigating through the system is simple and easy to do, aided by consistent user interface elements throughout all aspects of the system.

7.3 Research Question 2

Question: Is the Notification History system suitable for all users?

In relation to Research Question 2, the survey evaluated the systems suitability for varying types of users. Participants were asked to rate on a scale of 1 to 10 (1 being lowest, and 10 being highest) to evaluate each aspect of the system. The mean scores for each of the aspects of the system are shown in figure 7.2.

Figure 7.2: Learning Evaluation Chart

Considering the ordinal scale of the data (ratings), by using a repeated measures protocol (i.e. all users were required to answer all questions), a non-parametric Friedman test was used to test the statistical significance of the results for each aspect of the system. Statistical differences were only found between learning to operate the and remember how to carry out tasks.

Users found learning to operate the Notification History watch face relatively difficult (average 7.2) in comparison to the Notification History wear application (average 9.2) and Notification History mobile application (average 9.8). This is partly due to no help menu being available for the watch face, and so, the function of the notification counter was not initially evident to the user. The participants found that the Notification History watch face was easy to explore through trial and error (average 9.75). Once the function of the watch face and notification counter had been established, users found it to be straightforward to understand and easy to remember how to use when recalling how to perform tasks. This confirms that the watch faces function is simple enough to be used by a user of any age or technical ability.

All aspects of the Notification History wear application and mobile application scored exceptionally high averages (average ¿9) for ease of learning. This is partly due to the incredibly simplistic user interface. Because of this, the system is very easy to learn to use, straightforward, and thereby making it easy to recall how to perform tasks. With limited scope for trial and error exploration too, this reaffirms that the system is not aimed at particular user, and can be used by anyone with varying levels of technical ability.

7.4 Research Question 3

Question: How effective is the system at providing users with a way to review their notifications?

In relation to Research Question 3, the survey evaluated how effective the system is at delivering its functionality. Users were asked to rate on a scale of 1 to 10 (1 being lowest, and 10 being highest) to evaluate each aspect of the system.

The mean scores for each of the aspects of the system are shown in figure 7.3

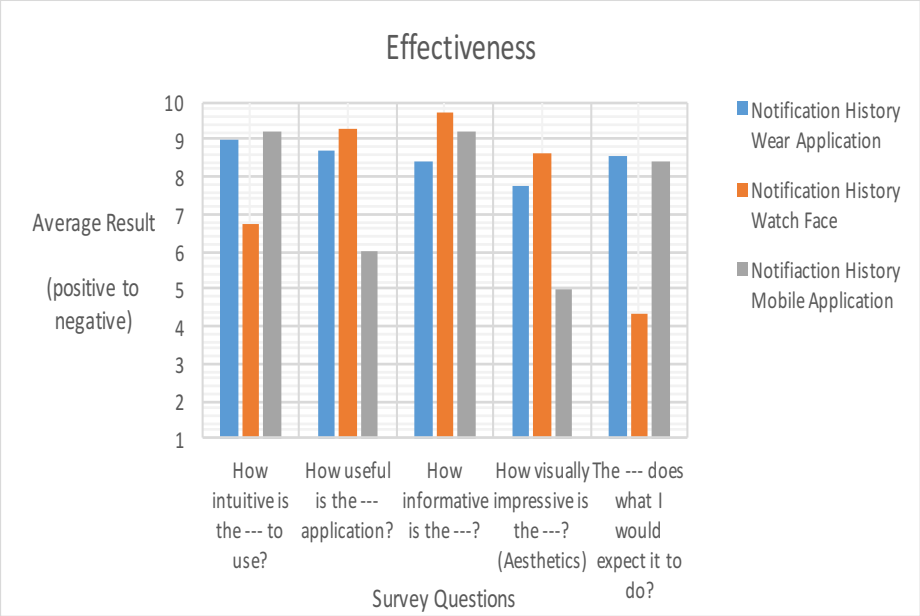

Figure 7.3: Effectiveness Evaluation Chart

Considering the ordinal scale of the data (ratings), by using a repeated measures protocol (i.e. all users were required to answer all questions), a non-parametric Friedman test was used to test the statistical significance of the results for each aspect of the system. The results of which that there are statistical differences between all results for each aspect of the system. Each question of the survey will therefore be discussed in detail to determine the extent of the systems effectiveness.

7.4.1 Intuitiveness

This section addresses the survey question: How intuitive is the Notification History wear application/watch face/mobile application? in relation to answering research question 3.

As previously discussed, the Notification History watch face was not deemed to be relatively intuitive to use (average 6.75), and this partly to fault that no help menu is available for the watch face. On the other hand, the Notification History wear and mobile applications scored highly (average 9.0 and 9.2 respectively) thus conveying that their functions could be established intuitively to the user.

7.4.2 Usefulness

This section addresses the survey question: How useful is the Notification History wear application/watch face/mobile application? in relation to answering research question 3.

Both the Notification History wear application and watch face scored exceptionally high average results for usefulness (8.7 and 9.25 respectively). This result is key in showing that the system does address the shortcomings of the Android Wear default notification service.

However, mobile application did not perform nearly so well (average 4.95). The reasons for this were given in the form of user feedback.

When describing the most negative aspects of the system, one user stated the mobile application was a little pointless, it should have just been part of the watch options. Another stated, didn't see why the mobile app was really needed. Given more time for the project, the mobile application could have been used to provide other features that users would have liked to have seen within the system, giving the mobile application more purpose to the user. Four users stated they would have liked to have had the option to select different types of watch faces for the system. Another user stated a help menu to explain some of the features and how to use it. The mobile application could be further developed to provide functionality for these other features as future work.

7.4.3 Information

This section addresses the question How informative was the Notification History wear application/watch face/mobile application? in relation to answering research question 3.

The wear applications purpose is to allow users to review notifications (Chapter 1, 1.3). The average user rating of 8.4 conveys how well this application delivers this functionality. The high average score shows that the wear application succeeds in meeting the projects primary requirement.

The Notification History watch face has only two functions, to provide the time, and new/unread notification count. Users deemed the watch face to be more than adequate at conveying this information (average 9.75), meaning that the secondary project requirement of allowing the user a way to determine the number of new or unread notifications, is met.

The mobile application which simply gives the user access to the wear application settings, due to the high average result (9.2), users found that the information provided for each setting in the application was sufficient.

7.4.4 Aesthetics

This section addresses the question How visually impressive is the Notification History wear application/watch face/mobile application? in relation to answering research question 3.

For the wear applications aesthetic design had to be minimal, due to the small screen of the smartwatch. The results for this question (standard deviation = 2.36) show that the results are quite spread. This is partly due to whether users liked the minimalistic feel or not. Despite this, users still found it to be particular intuitive and informative, meaning that the minimalistic design did not draw back from the effectiveness of its functionality.

The Notification History watch face had low standard deviation of results (0.94) and high average result (8.6) giving insight to users finding the aesthetics of the watch face to be pleasing, and suitably styled. In order to draw

a true comparison of watch faces, multiple watch faces would need to be developed and evaluated, however, this is not within the scope of this study.

The mobile applications user interface had a particular low average (4.95), which shows that users did not enjoy the aesthetics of the mobile application. This is partly due to the single settings screen of the application. In order to allow the application to be intuitive, it was necessary to follow the Android development guide for designing the user interface for application settings [13], so that any user would understand its purpose and use. Using a more creative user interface design for this aspect of the system, would detract from the Android user interface conventions (reference).

7.4.5 Expectation

This section addresses the survey question: The Notification History wear application/watch face/mobile application does what I would expect it to do? in relation to answering research question 3.

Both the wear application and mobile application had consistent user interfaces which allowed the user to predict the expected outcome of his actions when using the applications (user average rating 8.55 and 8.4 respectively). However, the Notification History watch face scored a low average result of 4.3. Again this conveyed by the lack of a help menu available for the watch face, as previously discussed in research question 2.

7.5 Conclusion to Chapter

The research questions posed in chapter 6 were explored through analysing the results of the survey carried out as part of the user study. The next chapter forms the conclusion of the report which includes future work and learning outcomes.

Chapter 8

Conclusion

8.1 Introduction

This final chapter aims to conclude the project. The results will be summarised and aligned to the objectives stated at the beginning of the report in chapter 1. Directions for future work will be discussed, and finally, the learning outcomes of the project will be highlighted.

8.2 Conclusion

The aim of the project was to develop a notification application for Android Wear smartwatches to allow users to review notifications. In order to successfully fulfil this project aim to its fullest extent, development of a wear application, a custom watch face, and a mobile application were developed.

The Notification History wear application successfully fulfilled this primary aim with its effectiveness at delivering this much needed functionality to Android Wear users (chapter 7, 7.4). The usability of this application was exemplified by its incredibly intuitive interface, making it easy for any user to pick up and learn to use (chapter 7, 7.3).

The combination of the Notification History wear application and Notification watch face gives the user a way to determine the number of new notifications in a novel manner. By replacing the traditional date/time counter within an analog styled watch face, with a notification counter, the user could then determine the number of new notifications from the smartwatch easily, and do so, with discretion. By combining the Notification watch face and wear application together, a completely unique way of interacting with the smartwatch was developed. Both secondary aims (chapter 1, 1.3) were satisfied on the completion of the functioning Notification watch face. Allowing users to establish the link between the Notification History wear application and watch face through trial and error, proved to be successful in providing users with an apt way of determining the functionality of the system (chapter 7, 7.3).

None of this would have been possible without the development of the mobile application, which is responsible for transmitting the notification data to the wear application, but also provides a simple intuitive interface (chapter 7, 7.43) to allow the user to fine tune the system with adjustable settings.

All Must Have and Should Have requirements have been met by developing these components of the system. Testing ensured that the wear application did not rely on the use of the Notification watch face to function, thus

fulfilling the non-functional requirements (NFR 1). The Could Have and Would Like To Have requirements were not feasible to complete due to the strict time constraints of the project.

A system which makes use of a watch face in this way has never been developed before. It is the first of its kind, and is free to download from the Google Play store.

8.3 Future Directions

- **Notification Watch Face Adaptability**

  The development of an adaptable watch face skin so that a smartwatch of any screen shape or size developed in the future, could make use of the Notification watch face. Currently, the Notification watch face can only be used with smartwatches with a round display.

- **Multiple Notification Watch Faces**

  The development of different Notification watch faces to give the user a choice of watch face to choose from (requirement WLTH 1). From the user survey, users requested this as part of the features they would have liked to have seen within the system.

- **Porting the Mobile Application to Other Smartphone Operating Systems**

  Android Wear smartwatches can currently be paired with smartphones running other operating systems besides Android. In order for the Notification History system to function with smartphones not using Android, separate mobile companion applications would need to be developed for those smartphone operating systems.

8.4 Learning Outcomes

Before starting this project, I had no experience developing for wearable devices. Due to the project encompassing both mobile and wearable application development, it gave me a comprehensive experience into development for smart devices. I became immersed in developing for both mobile and wearable platforms alike. Because the project dealt with current limitations of smartwatches with no predefined project aims, it forced me to create a unique and creative solution to a problem that I had to identify myself. I believe this project to be an invaluable experience to me helping to develop both my analytic and creative skills.

Bibliography

[1] Android, ”Android Wear,” 18 March 2014. [Online]. Available: http://www.android.com/wear/.

[2] IDC, ”Smartphone OS Market Share, 2015 Q2,” 1 April 2015. [Online]. Available: http://www.idc.com/prodserv/smartphone-os-market-share.jsp.

[3] BBC, ”Google unveils Android for wearables,” 18 March 2014. [Online]. Available: http://www.bbc.co.uk/news/technology-22255987.

[4] IDC, ”IDC Forecasts Worldwide Shipments of Wearables to Surpass 200 Million in 2019, Driven by Strong Smartwatch Growth,” 17 December 2015. [Online]. Available: https://www.idc.com/getdoc.jsp?containerId=prUS40846515.

[5] J. Fedewa, ”7 Things I Hate About Android Wear,” 6 November 2014. [Online]. Available: http://phandroid.com/2014/11/06/7-things-i-hate-about-android-wear/.

[6] Android, ”Design Principles for Android Wear,” Android, [Online]. Available: http://developer.android.com/design/wear/principles.html.

[7] S. Schirra and F. R. Bentley, ”Understanding Everyday Uses of Consumer Smart Watches,” Yahoo, San Francisco, 2015.

[8] K. Lyons, ”Using Digital Watch Practices to Inform Smartwatch Design,” Yahoo Labs, Sunnyvale, 2015.

[9] J. H. e. al, ”SplitBoard: A Simple Split Soft Keyboard for Wristwatch-sized Touch Screens,” HCI Lab, KAIST, Daejeon, 2015.

[10] W. V. V. e. al, ”WatchMe: A Novel Input Method Combining a Smartwatch and Bimanual Interaction,” Hasselt University, Hasselt, 2015.

[11] B. S. Chaparro, ”Is Touch-based Text Input Practical for a Smartwatch?,” Wichita State University, Wichita, 2015.

[12] A. Villas-Boas, ”android wear notifications are bad on smartwatches 2015 9,” Tech Insider, 24 September 2015. [Online]. Available: http://uk.businessinsider.com/android-wear-notifications-are-bad-on-smartwatches-2015-9. [Accessed 11 November 2015].

[13] Android, ”Creating and Running a Wearable App,” Google, 14 March 2014. [Online]. Available: http://developer.android.com/training/wearables/apps/creating.html. [Accessed 11 November 2015].

[14] Android, ”Creative Vision for Android Wear,” Android, 14 March 2014. [Online]. Available: http://developer.android.com/design/wear/creative-vision.html. [Accessed 18 December 2015].

[15] Google, ”Settings,” Google, [Online]. Available: https://www.google.com/design/spec/patterns/settings.html.

[16] Android, ”Notifications,” Android, 18 March 2014. [Online]. Available: http://developer.android.com/design/patterns/notifications.html.

[17] Google, Creating Lists, San Francisco, California: Google, 2016.

[18] Android, ”Designing Watch Faces,” Android, [Online]. Available: http://developer.android.com/training/wearables/watch-faces/designing.html

[19] C. e. al, ”Development of an Instrument Measuring User Satisfaction of the Human-Computer Interface, Evaluation Questionnaire,” Association for Computing Machinery, Maryland, 1988.

[20] Google, Creating Card Fragments, San Francisco, California: Google, 2016.

Appendices

Appendix A

Consent Form

PARTICIPANT CONSENT FORM

The consent form is to participate in the Level 4 Software Engineering project in the School of Computing Science, University of Glasgow.

The objective of the project is to: Develop an android based notification centre for Android Wear smart watches.

- Determine the number of new notifications residing on the smart watch.
- Review notifications using the smart watch.
- Inspect the contents of both new and old notifications on the smart watch.
- Set the limit of notifications that the notification history application can hold.
- Delete old the notification history on the smart watch.

Both the Android Wear application and companion mobile application have both been implemented with all the required functionalities and now are in the stage of evaluating the final software product. The evaluation will be conducted over the following stages:

- **Stage 1 Control setup exploration:** You are required to explore the default functionality of an Android Wear smart watch connected to an Android mobile device.
- **Stage 2 Evaluating the control setup:** In this stage, you need to complete the following tasks to the best of your ability, using the smart watch and mobile.
    - **Task 1:** Determine how many unread notifications there are using the smart watch.
    - **Task 2:** Inspect the contents of the new notifications.
    - **Task 3:** Limit the number of notifications the smartwatch can hold at any one time.
    - **Task 4:** Review all the notifications using the smartwatch.
    - **Task 5:** Delete all notifications
- **Stage 3 Experimental setup exploration:** You are required to explore the Notification History wear and mobile applications using the smart watch and mobile device.
- **Stage 4 Evaluating the experimental setup:** In this stage, you are required to perform the same tasks as performed in Stage 2, making use of the new applications functionality.
    - **Task 1:** Determine how many unread notifications there are using the smart watch.
    - **Task 2:** Inspect the contents of the new notifications.
    - **Task 3:** Limit the number of notifications the smartwatch can hold at any one time.
    - **Task 4:** Review all the notifications using the smartwatch.
    - **Task 5:** Delete all notifications
- **Stage 5 Exit Survey:** In this stage, you will be answering a short questionnaire to evaluate the Notification History wear application, mobile application, and Notification History wear watch face. This may take around 10-15 minutes. The questionnaire will be sued for the purpose of research.

Figure A.1: Consent Form page 1

Please note is it is the Notification History applications we are evaluating and not your technical skills.

DATA PRIVACY

- Your answers to the questionnaires will be made anonymous and held in strict confidence. The recording of information using the Google form service is kept secure using SSL encryption. All recorded information will be protected by Google using a private account. Access will be restricted to only those people involved in the project. This data won't be shared with any third-party or used for commercial purposes.
- No personal information will be recorded, except for email address and demographic information (i.e. age, gender, profession). The email address will be used to contact you during the duration of the evaluation only. The anonymous demographic information will be used for the purpose of the evaluation only.

CONSENT TO USE INFORMATION FOR RESEARCH

By signing the form (page 3), you give permission to:

- Contact you during the evaluation, for the purpose of sending the online questionnaire link using the email address that you have provided.
    - You will receive a link to the survey by email contact: 1002167m@student.gla.ac.uk

Figure A.2: Consent Form page 2

CONSENT TO THE USE OF DATA
University of Glasgow, College of Arts Research Ethics Committee

I understand that _ Robert McAlpine _____
(name of researcher)

is collecting data in the form of
_____completed questionnaires_____

for use in an academic research project at the University of Glasgow.

I give my consent to the use of data for this purpose on the understanding that:

- All names and other material likely to identify individuals will be anonymised.
- The material will be treated as confidential and kept in secure storage at all times.
- The material will be retained in secure storage for use in future academic research
- The material may be used in future publications, both print and online.

Signed by the contributor:_____ Date:
Email Address: _____

Signed on behalf of the contributor (i.e. parent/guardian in case of a person under 18)
_____ Date:
Email Address: _____

Researcher's name: Robert McAlpine
Researcher's email contact: 1002167m@student.gla.ac.uk

Supervisor's name: Soumyadeb Chowdhury/Lewis Mackenzie
Supervisor's email contact: soum.chowdhury@singaporetech.edu.sg/ lewis.mackenzie@glasgow.ac.uk

Department address:

School of Computing Science
University department
18 Lilybank Gardens
Glasgow
G12 8RZ

Figure A.3: Consent Form page 3

Appendix B

Evaluation Questionnaire

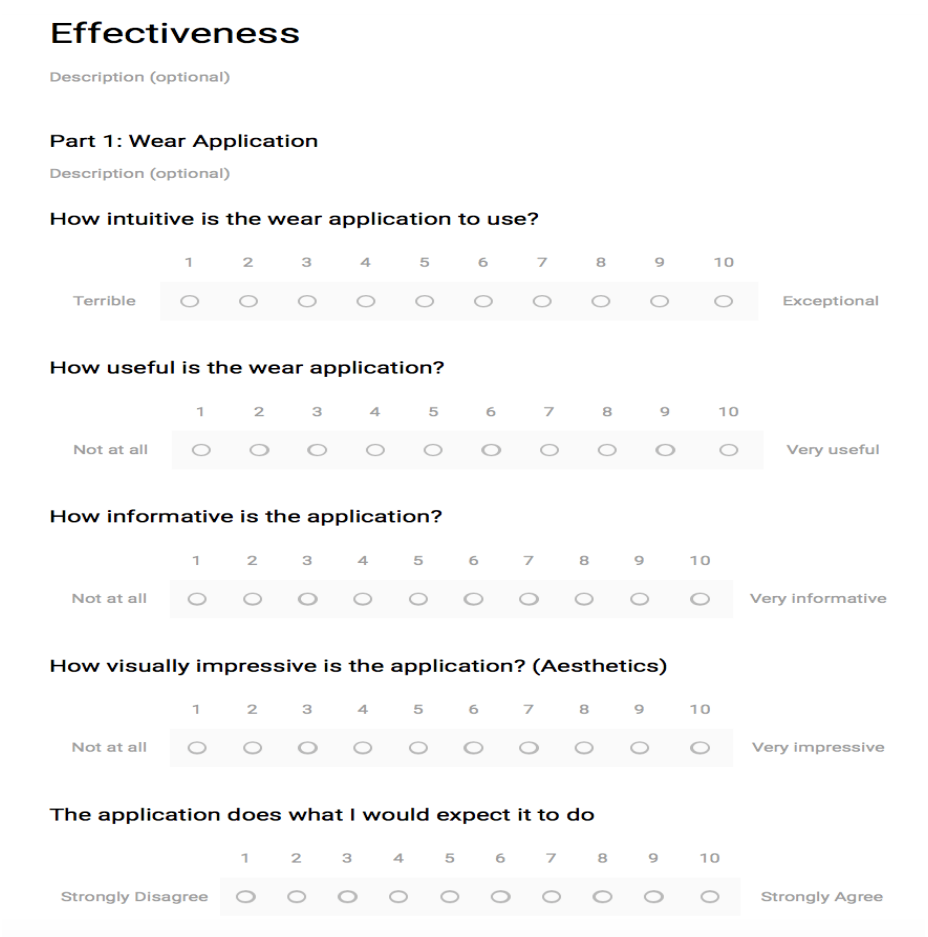

Figure B.1: Evaluation Questionnaire page 1

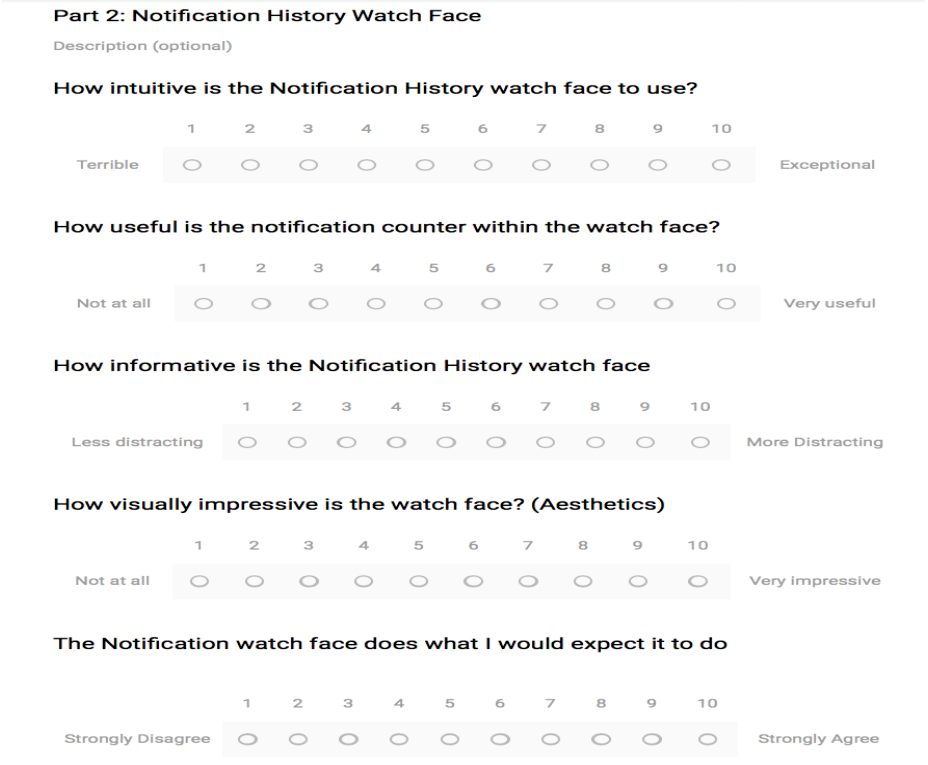

Figure B.2: Evaluation Questionnaire page 2

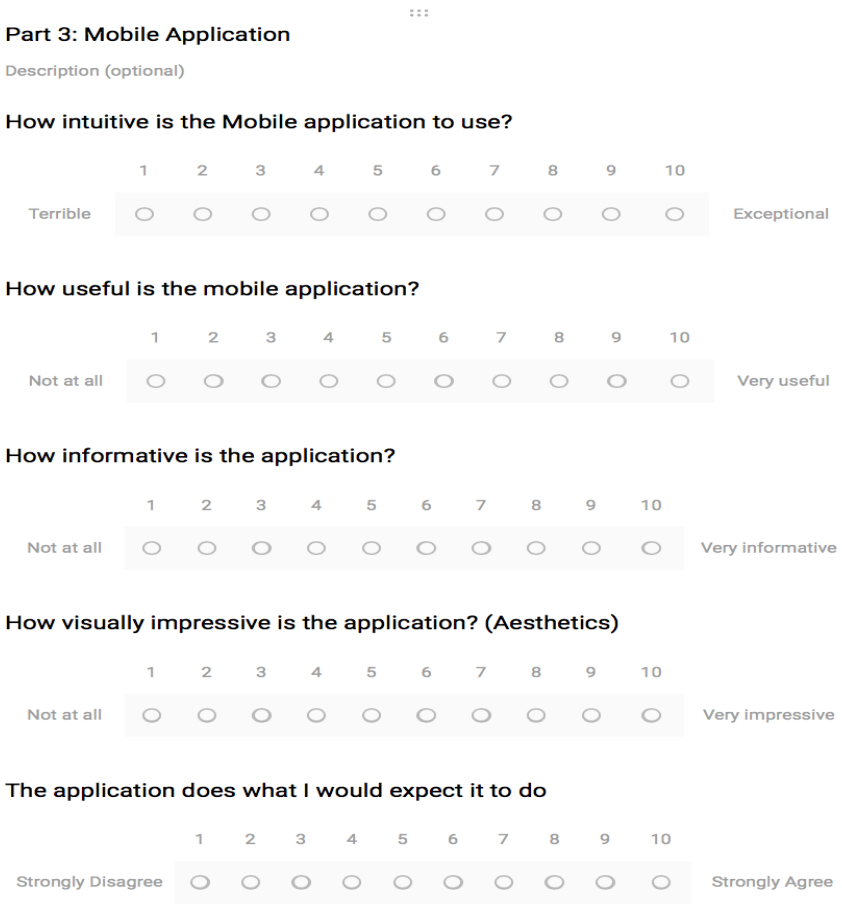

Figure B.3: Evaluation Questionnaire page 3

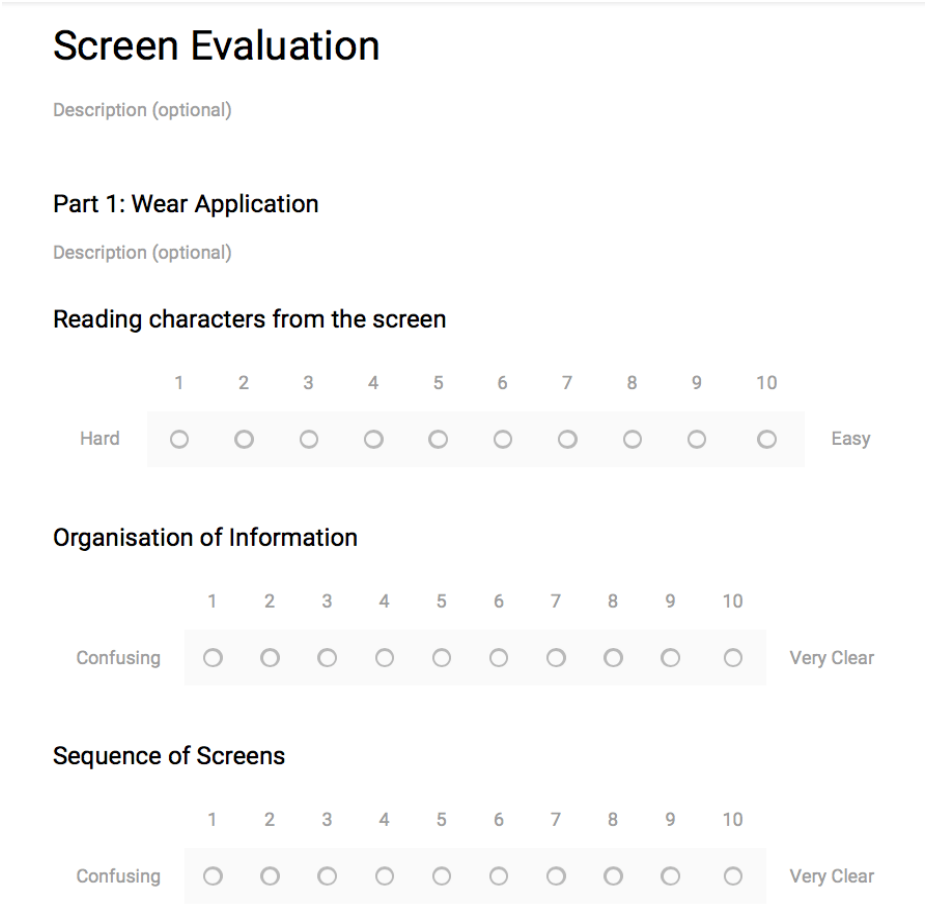

Figure B.4: Evaluation Questionnaire page 4

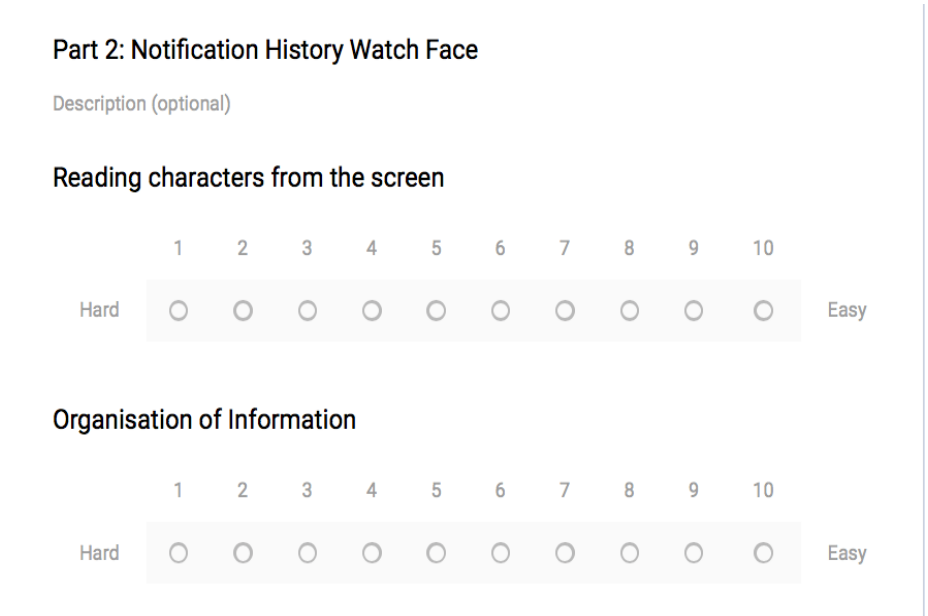

Figure B.5: Evaluation Questionnaire page 5

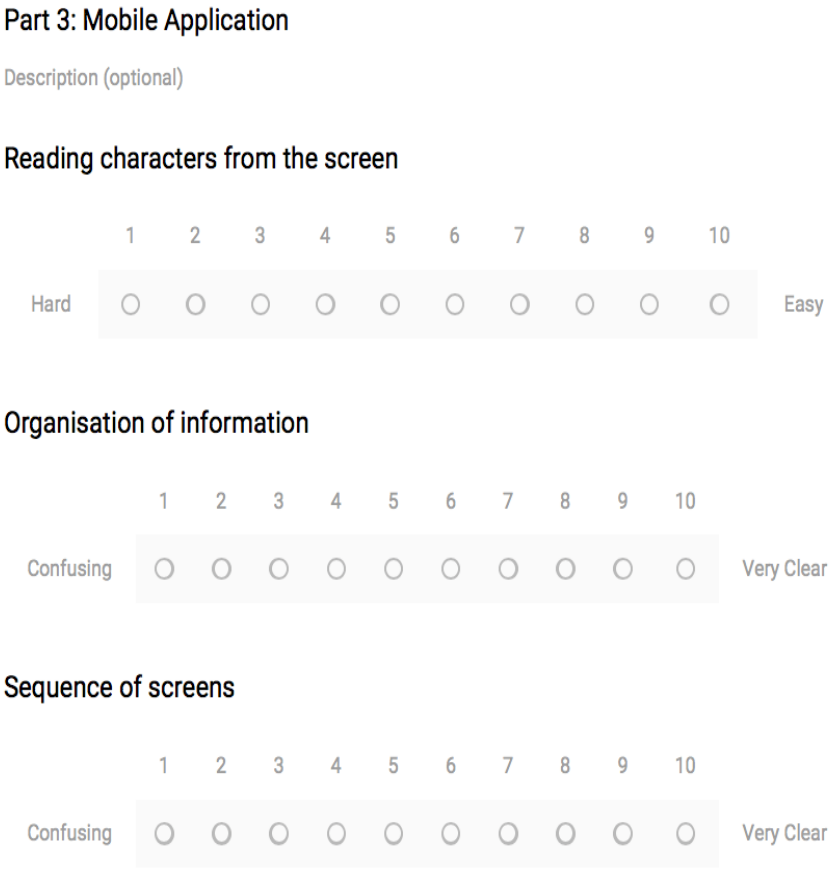

Figure B.6: Evaluation Questionnaire page 6

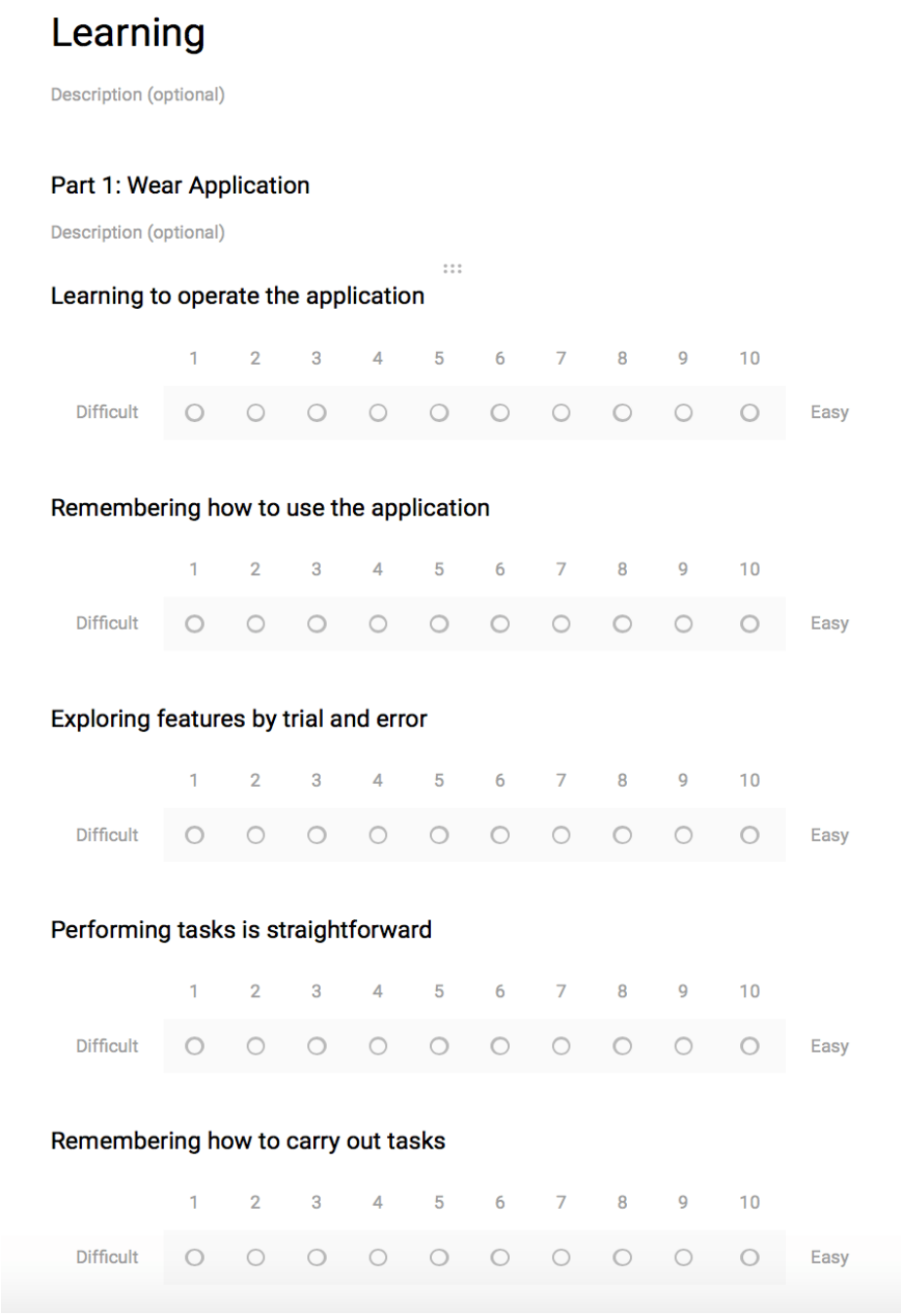

Figure B.7: Evaluation Questionnaire page 7

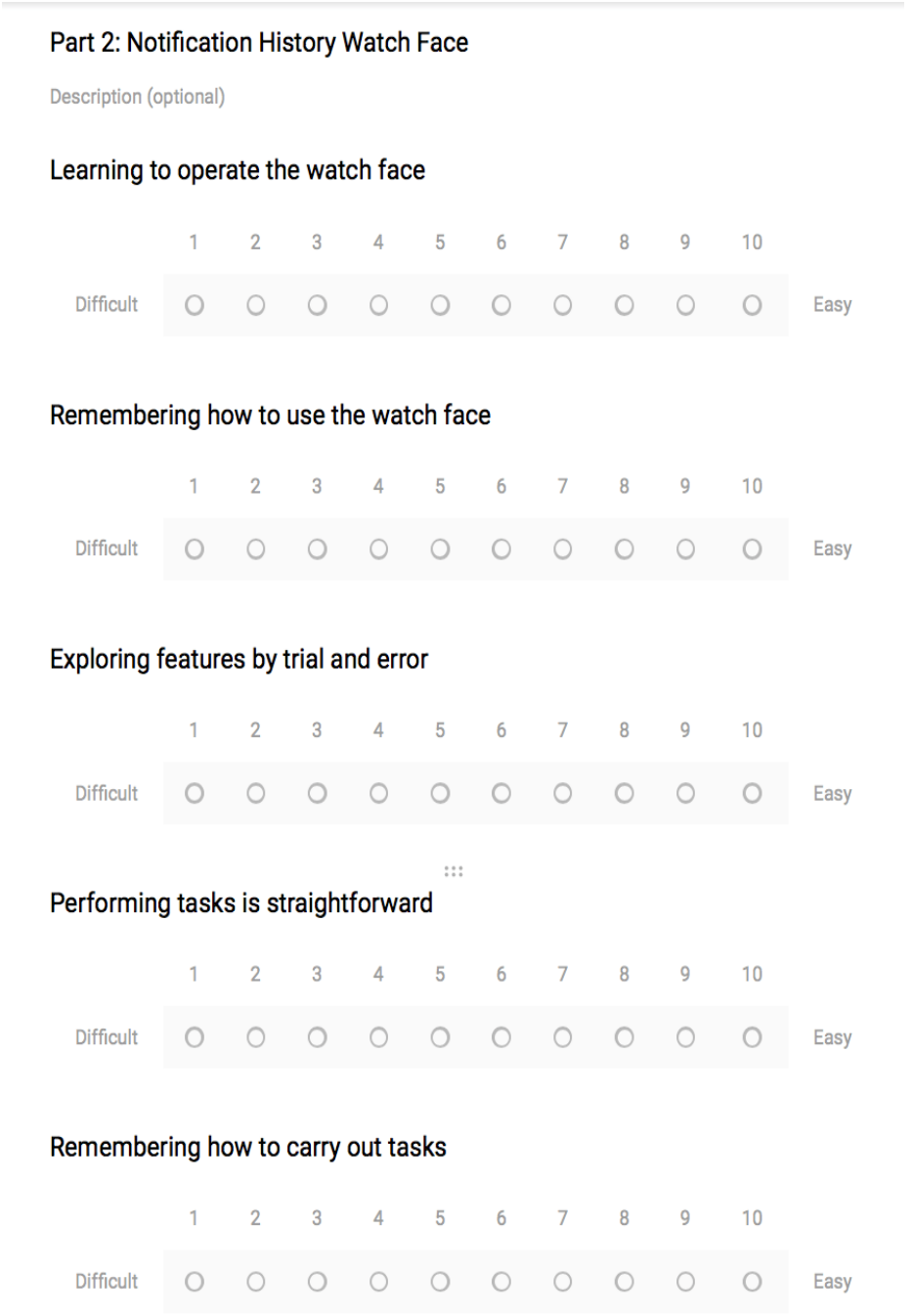

Figure B.8: Evaluation Questionnaire page 8

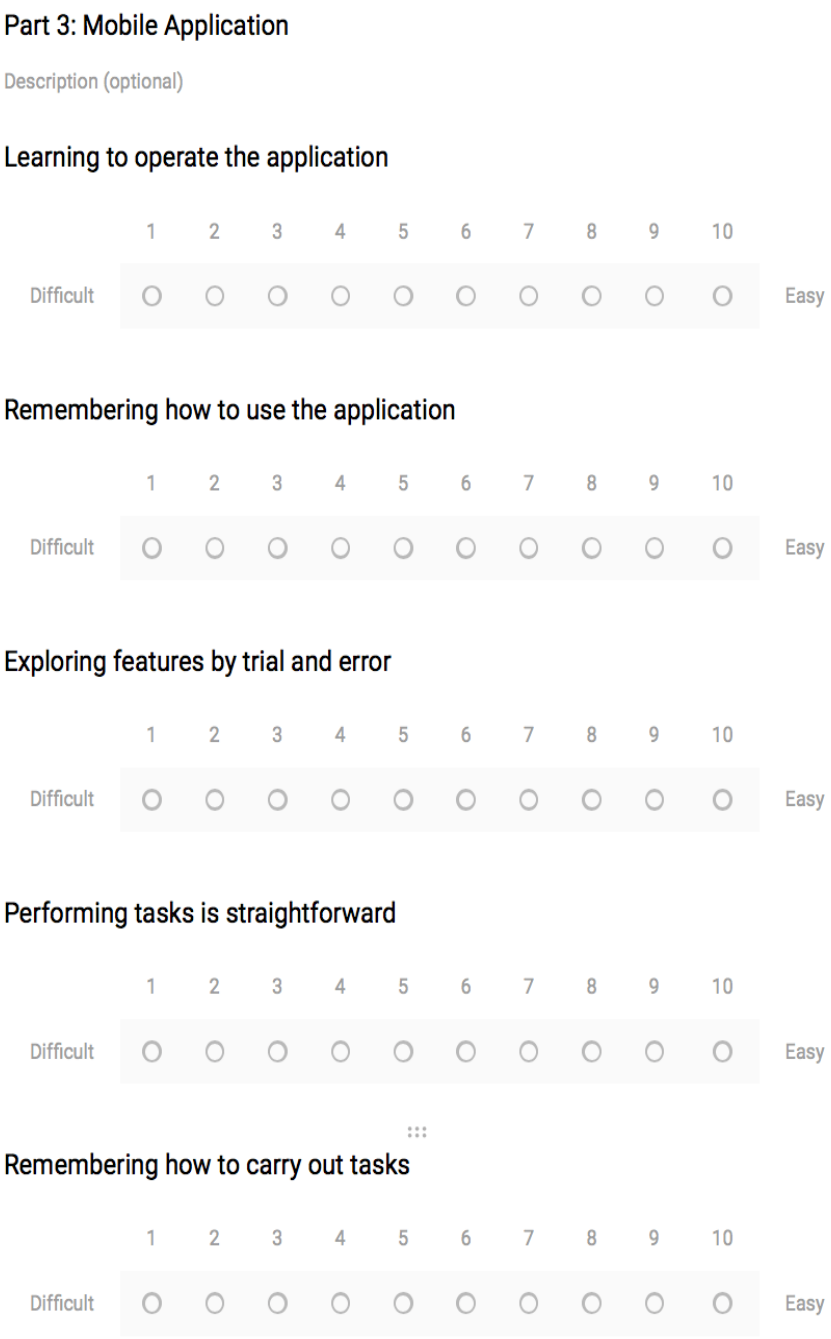

Figure B.9: Evaluation Questionnaire page 9

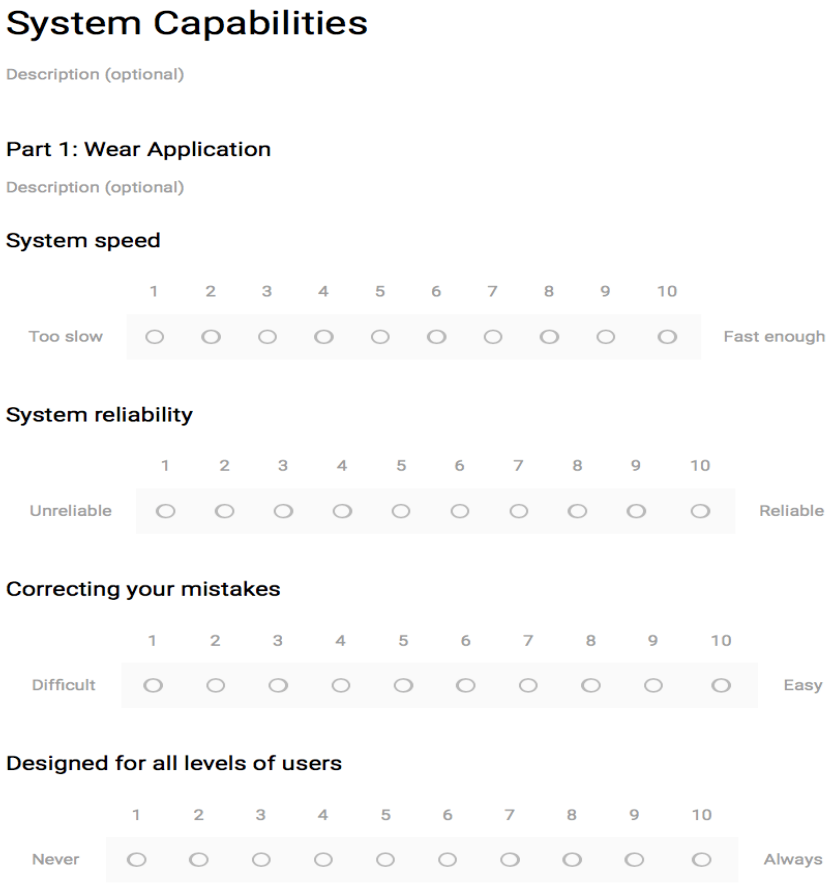

Figure B.10: Evaluation Questionnaire page 10

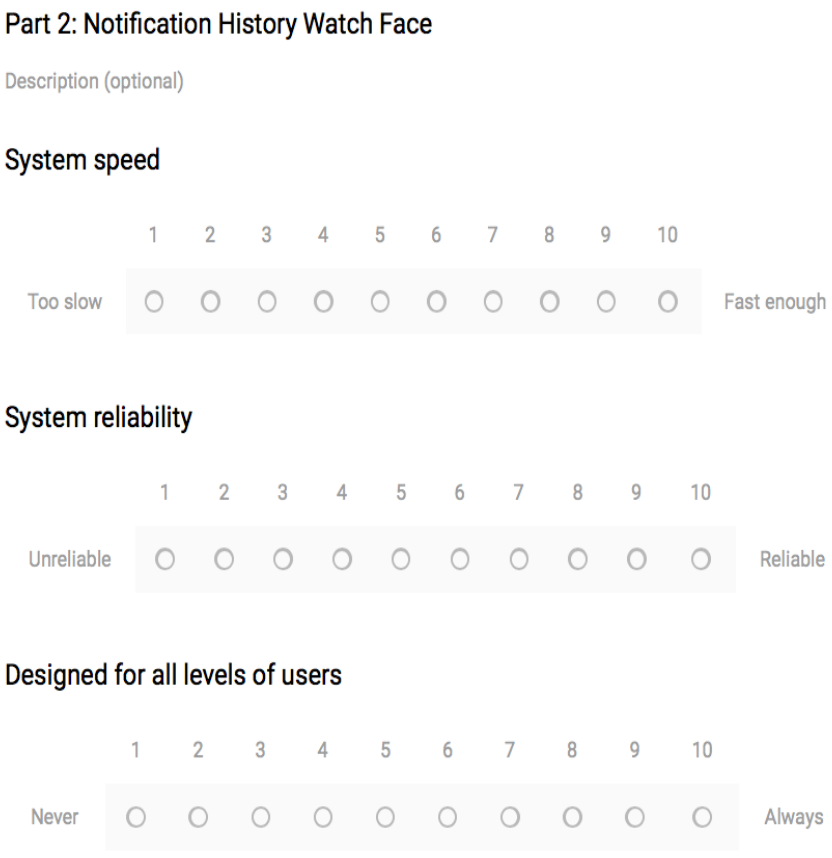

Figure B.11: Evaluation Questionnaire page 11

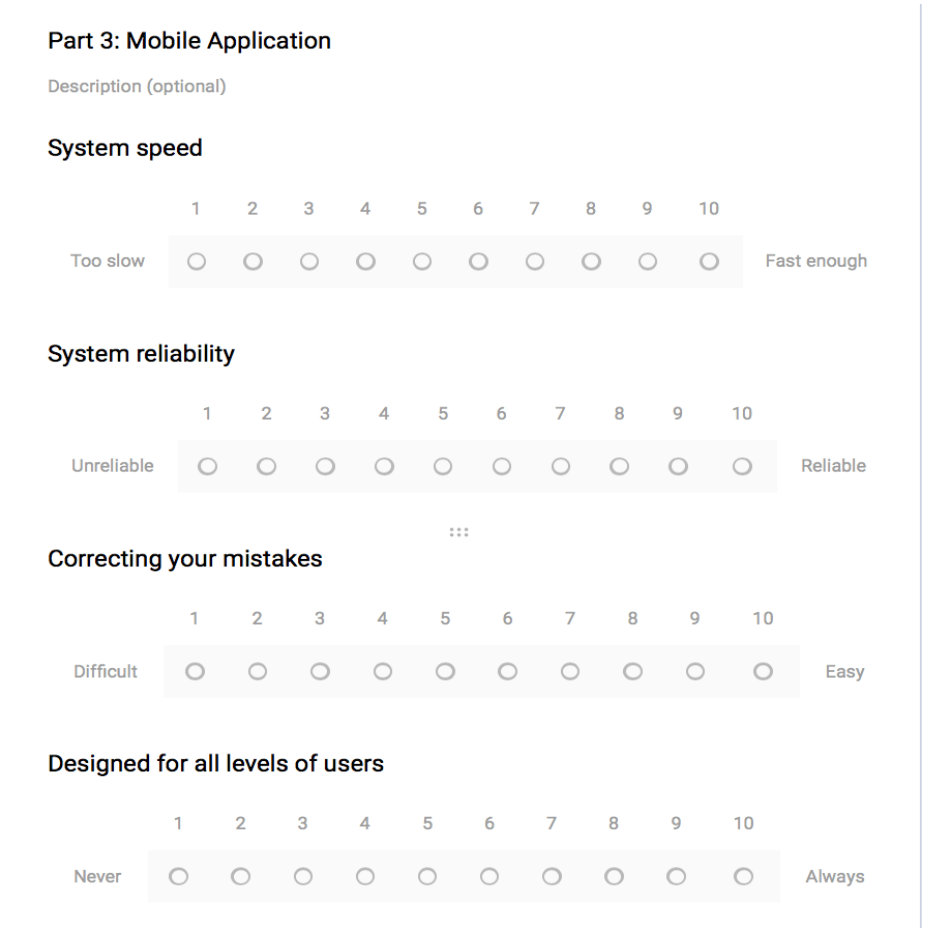

Figure B.12: Evaluation Questionnaire page 12

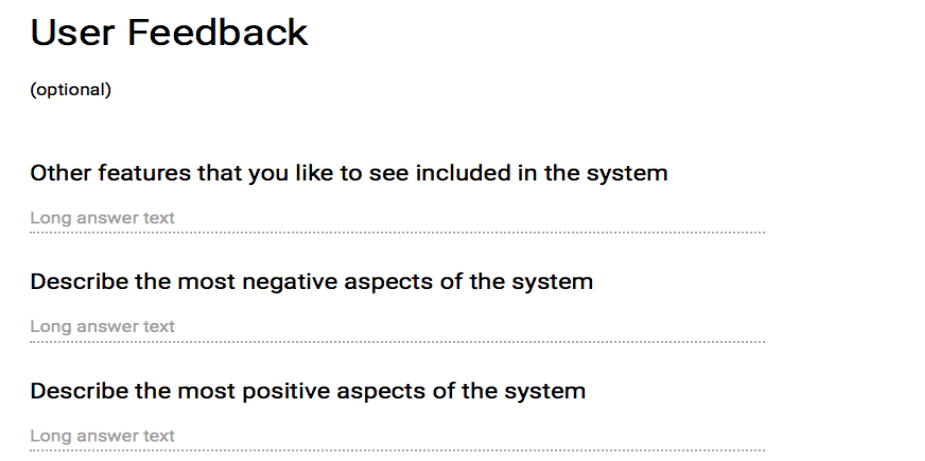

Figure B.13: Evaluation Questionnaire page 13