2015

A Survey of Smartwatch Platforms from a Developer's Perspective

Ehsan Valizadeh
Grand Valley State University

A Survey of Smartwatch Platforms from a Developer's Perspective

By

Ehsan Valizadeh

April, 2015

A Survey of Smartwatch Platforms from a Developer's Perspective

By

Ehsan Valizadeh

A project submitted in partial fulfillment of the requirements for the degree of

Master of Science in

Computer Information Systems

At

Grand Valley State University

April, 2015

_____

Dr. Jonathan Engelsma **April 23, 2015**

Abstract

Technological innovation has made it possible to package a powerful processor and memory subsystem coupled with a high-resolution display, wireless communication, and specific sensors into a device known as the smartwatch. This device introduces a new set of new challenges such as battery life, user interaction, and other how to create applications for it. Smartwatches are connected to the Internet and provide gesture interaction and the ability to continuously monitor a user's physical activity. The smartwatch has access to the smartphone and therefore can be used as a second display to show users various notifications from the phone such as calls and messaging as well as information from the Internet, such as social networking apps (Facebook, Twitter etc.), to do lists and many other applications.

The objective of this project is to explore smartwatch technology from a developer's perspective. A short history of smartwatch technology is given along with a discussion of the typical use cases. This is followed by a deeper technology dive into the two most popular smartwatch platforms on the market today: the Apple Watch and Android Wear. This paper will look at the features of both devices side by side, and take a look at the development platforms available to create applications on them. This study will discuss the frameworks, the tools, and some of the challenges we encountered in learning to build applications with them. A sample application for both platforms will be presented and discussed.

Introduction

What is a Smartwatch

A smartwatch is a wearable processing gadget that looks like a wristwatch or other time-keeping device. In addition to telling time, many smartwatches are Bluetooth-capable. The watch becomes, in effect, a wireless Bluetooth accessory capable of extending the capabilities of the wearer's smartphone to the watch. A smartwatch can connect directly to a person's smartphone and display notifications, updates, and messages right from user's wrist. That means no one needs to get his/her smartphone out of pocket to check their email. [1]

A Brief History of SmartWatches

Smartwatches are not really new technology. Casio and other companies as far back as the 1980s developed calculator or 'data bank' watches. Companies like Microsoft and Sony created simple wrist-mounted notification devices (the SPOT and the Sony LiveView respectively) that also failed to capture the public imagination. Why? Perhaps they were ahead of their time. A new generation of teenagers had grown up without ever having a need for a wristwatch. They already have smartphones, which meets their time keeping needs.

The most recent generation of smartwatches really began in 2013, led by the kick-starter Pebble smartwatch project that secured a record amount of funding for a start-up smartwatch company in a record time. [2]

1. Pulsar	• The Worlds First LED Digital Watch	• 1972 - 1977
2. Casio Databank Telememo 10 (CD-40)	• They allow data storage such as telephone numbers and email addresses, and usually provide a calculator. • Function that could save and recall 10 groups of 16 letters or numerals. o Storage: 10*16	• 1984
3. Seiko RC-1000 Wrist Terminal	• Interface with Apple II, Commodore 64, IBM PC, NEC, & Tandy	• 1985
4. Samsung SPH-WP10	• It weighed 50 grams with its battery. • Could last for 90 minutes of talk time and 60 hours on standby. • It also cost close to $700. • Analog cell phone + watch.	• 1999
5. Fossil Wrist PDA	• Palm OS 4.1 • 160 x 160 illuminated screen • 2MB internal memory • Address book, memo pad, to do list and calculator. • $250	• 2003
6. Microsoft SPOT • Smart Personal Object Technology	• The SPOT technology used MSN Direct network services, delivered across the United States and Canada based on FM radio broadcast signals in about 100 metropolitan areas. The service cost $59 a year.	• 2004

7. Samsung S9110	• Touch screen • Cell phone (GPRS) • MP3 player • Voice recognition • Bluetooth • 40MB storage • Up to 4 h 15 min	• 2009
8. Allerta inPulse	• Bluetooth 2.0 • Call notifications • Android and BlackBerry • Vibrating motor • Charges via microUSB	• 2009
9. iPod Nano 6th Gen	• Touch screen • Watch App • Music player • Watchband	• 2010
10. WIMM One	• Developer preview device • Android core • Wifi + Bluetooth classic • Compete against iPod Nano • Wrist-strap add-ons • Bought by Google in 2012 • (Folded into AndroidWear) • Run time: 30 hours	• 2011
11. Sony SmartWatch 2	• Custom apps • Music player • Bluetooth classic • Notifications • Android support	• 2012
12. MetaWatch	• Development platform • Ex-Fossil engineers • Bluetooth 2.1+LE • 6 buttons • Notifications + live data • iOS & Android •	• 2012

13. Pebble	• e-Paper display • 5-7 days battery • Apps + watchfaces • Phone + music control • iOS & Android • Bluetooth (classic+LE) • Accelerometer • Vibration • $99 - $199	• 2012
14. Samsung Galaxy Gear	• Custom Android or Tizen OS • Phone controls • Voice commands • Camera (watchband) • Pair only with Samsung phones	• 2013
15. Martian	• Voice Command • Speaker + Mic • Notifications • Bluetooth classic + LE • Analog + digital	• 2013
16. Basis B1	• Heart rate monitor • Step & sleep tracker • Skin temperature • Perspiration • iOS & Android • Bought by Intel	• 2013
17. AndroidWear	• Square & round format • Voice control (Google Now) • Bluetooth LE • Watchfaces • Java-based apps • Accelerometer • Based on KitKat • Pair with Android only	• 2014

18. i.am+ Puls	• Cell GSM 3G • Voice control • Wi-Fi • Bluetooth • GPS • Pedometer	• 2014
19. Apple Watch	• Custom apps • Watchfaces • Notifications • Health tracking + HRM • Phone/media control • Touchscreen • Digital Crown • Haptic touch • NFC/Apple Pay • iOS only	• 2015

Table 1. History of Smart watches

Not Just A Wrist Display

Smartwatches that merely attempt to do the same things that a wristwatch can already do are bound to fail. A survey published in August, indicates only three per cent of respondents in the U.S. and U.K. said that they owned a smartwatch. Another two per cent owned one but did not use it anymore. Only sixteen per cent of men and ten per cent of women said that they planned to buy one by the end of the year. [3] The reasons for not buying a smartwatch are as follows: people did not see the point, and they found them too expensive. What is notable is the percentage of people who do not see what makes a smartwatch particularly useful. While MP3 players could be marketed as a replacement for CD players, and smartphones could be sold as better cell phones, smartwatches have nothing to displace. Companies have to persuade people to add a device to their lives.

Pop-up notifications about e-mails, appointments, and missed calls can be useful, but they can also be overwhelming, and many of the existing smartwatches are worse than smartphones at letting users customize the alerts. Fitness features like heart-rate

trackers are nice, but specialized fitness devices like the Fitbit already do all that. Smartwatches, with small screen sizes and limited functionality, do not make anything that one can do on a smartphone more convenient. Smartphones are already so portable that the need for general-purpose smartwatches just does not seem to exist.

Do I Need a Smartwatch?

Just because smartwatches are becoming popular and garnering plenty of media attention does not mean they are something we should all be rushing to buy. It is important to understand that most smartwatches are not standalone products. They often need to be associated with a smartphone. The reason for that is smartphones act like a gateway between these two devices, so smartwatches can use the phone's Internet connection to send/receive data, and other services such as GPS and phone calling and text messaging. People cannot replace a phone with a smartwatch. They act more like a second display for a phone.

There are a number of reasonable use cases emerging that make the smartwatch a reasonable solution. Examples of these use cases would include the following typical scenarios:

- Health and Fitness
    - If people are looking for a type of health and fitness monitoring sensor device, then a smartwatch might be a great way of getting this functionality. They can keep track of their workouts, locations, and sleep patterns and it can be used to help keep track of a diet.
- Business
    - There are people who cannot stop looking at their phone. Maybe they are checking emails, or setting up appointments between meetings. Some of the features of many models of smartwatches include alerts, which notify their owners that a message or email has been received. Therefore smartwatches make people watch their hands instead of their phone.
- Customize/control other devices/appliances in a personal environment (Home)
    - Users can pair their watch with iBeacon or hue bulbs through phones.
- High tech fashion

- People wear a watch not only to tell time, but also to make a statement to others. A watch is the finishing touch on fashion. Smartwatches face a challenge unique in the tech industry: balancing function and fashion.

Privacy And Security

It is a digital world; it has become even more crucial to be worried about personal privacy and security. People are using smart devices everyday and they are certainly making our lives very simple and easy, but these devices also count as serious threats to important and personal data. As Tim Cook announced the Apple Watch, he named this device as the most personal product that Apple has ever made. The question is how secure is this device? Most smartwatches do not timeout after a certain period of time. Instead, they will lock automatically when they are out of range of a paired phone. In addition, the PIN lock feature does not always protect against USB debug access, especially on Android devices. In some smartwatch models, there are options like using biometric data (fingerprints) to gain access, or additional code entry sequences such as a pattern or gesture lock.

There is a lot of personal data on smartphones and tablets, which users carry everywhere with them. These devices are connected to the Internet most of the time, because they should be. However, that can cause serious security and privacy issues. Facebook, Twitter, Google or others have access to almost all of personal data, which are then used across various platforms, causing serious issues when it comes to privacy. Smartwatches can expose people to a whole new kind of security threat and make sensitive information wide open to hackers.

Smartwatches Vulnerabilities

Some specific vulnerabilities that can be anticipated are as follows:

- Bluetooth connection:

The Bluetooth communication between most smartwatches and smartphones for the purpose of transferring data, depend on a six-digit PIN code, which can be easily brute-forced by hackers to access our data.

- On-Watch Software and Apps:

Some smartwatches, like Pebble 2.0, requires to download apps from its own app store. However, to use these apps on a smartwatch, users must provide username and password. And if at all these apps get hacked, which is quite possible, then it can put a account credentials in danger. [4][6]

Is It Safe?

Where people leave their watch when they are not wearing it, could pose additional risks. It is not like the old days where a person was just sad because he/she lost his/her watch. It is going to be painful, because these day's watches are not just simply a watch. It will be part of their personal life. Users should know how important it is to make sure that they are ready for the responsibility that comes along with getting a smartwatch.

What people think about SmartWatches

A quick Google with the term "smartwatch" turns up some interesting links:

- Smart watches are stupid
- Smart watches are dump
- Smart watches are the future
- Smart watches are they worth it

Clearly companies should work more on marketing the technology. They must show people why they should carry on one of their product. Conventional watches can show me time just fine and users can buy a smartband for $30 that can be used to adequately monitor health.

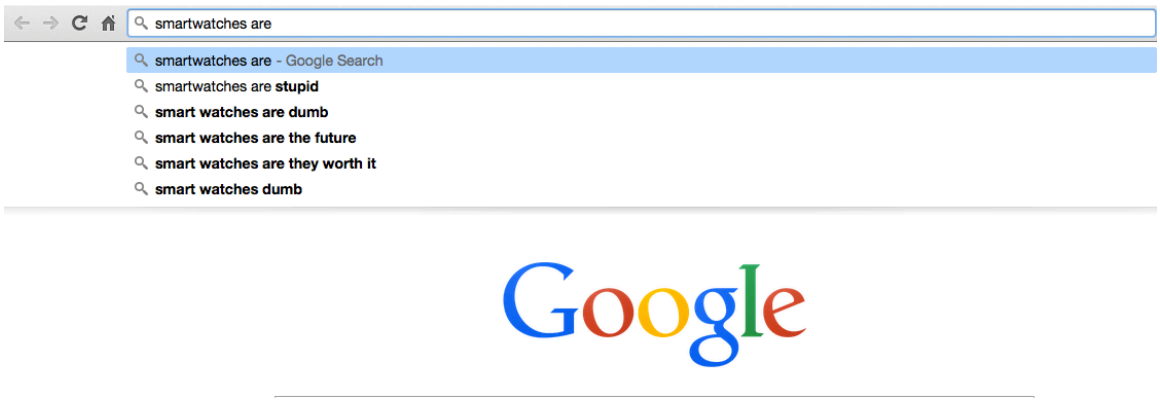

Figure 1. "Smartwatches are" in Google Search

Are all Smartwatches the same?

Smartwatches are not all created equal. There are different brands, models and even variations. Each of them offers separate features and functionality. They are hundred companies out there and each of them has a specific features. But here in this paper I'll focus on two of them. Apple Watch and Google Wear. Here are some brands that make Smart watches:

ConnectedDevice / Razer Nabu / Kronos / MQ / Garmin (GPS + exercise) / Samsung Gear S / Microsoft Band / I'm Watch / Motorola MOTOACTV / MyKronoz / A.I-Watch / Agent / Rufus Cuff / Androidly / VEA Buddy / Dew Motion iStick / Neptune Pine / Omate TrueSmart / Plastic Logic / HP-MB Chronowing

Apple Watch

Overview

The Apple Watch is a new product category for Apple: it's a smartwatch. Apple has been in wearable tech before with the older iPod Nano and its wristbands, but Apple Watch is a completely new platform: it's made to run its own apps, connect with iPhones, and simultaneously make a fashion statement. Apple presented its smartwatch on September 9th, 2014 and later previewed it further on March 9th, 2015. **[5]**

Why is Apple Watch not out yet

Here are some of the reasons Apple Watch is not in store yet: [7]

- It is not precise enough
  - The fitness results were not quite right for people with hairy arm or dry skin. The sensors could also be thrown off depending on how tightly the device was worn. Overall the readings are not fully accurate.
- Equipment delays
  - According to a MarketWatch report from late last year, Apple had trouble to get the necessary parts to build the watch. This has prolonged the release date.
- Potential regulatory issues
  - The best part about Apple Watch is the health Apps. If Apple Watch gives people advice about their diet or health base on biometric data, the FDA and other regulators would have had to approve it.

What Do The World's Top Watchmakers Think of Apple Watch?

Not much. "It looks a little cold, and lacks, for my taste, a bit of personality," said Jean-Claude Biver, a legendary figure among Swiss watchmakers who led the revival of Switzerland's Blancpain brand in the 1980s before stints at Omega and Hublot.

Let’s see what would be the main reasons to buy a Rolex:

- You are a rich person.
- You want to have something nice for your favorite kid after you are done with this world.

- A brand like a Rolex is a peace of art on your hands. Of course when you something like that on your wrist you feel different.
- It gets more valuable as you become older.

And Apple Watch:

- Obviously you are an Apple fan.
- You like Technology.
- In at most 2-3 years there will be a new version of Apple Watch. And yours is not worth the money you paid for it anymore.

It is a new challenge for those watch manufacturers and of course this time it is not just another company. Apple has created a newer and more sophisticated technology, and it is not just another product. The Apple Watch may reach sales of 20 million to 30 million units annually in the first year. Watch manufacturers and retailers have reported rising exports, collectively reaching $23.3 billion in 2013. So traditional companies should think of new products or they should change their prices because there is a new competitor, Apple. There is another fact; Apple has a fantastic software community; which they have awesome ideas. Apple Watch will be the biggest upheaval the Swiss industry has faced since competition from cheaper Japanese quartz watches brought it to its knees in the 1970s.

Why the Apple Watch Exists

In the fourth quarter of last year, Apple sold nearly 75 million iPhones at a profit margin of nearly 40%, making its quarter, the most profitable quarter in the history of any American business. [8] iPhone is the most successful product Apple has ever made. There is no product that can give Apple such a good business benefit. If Apple builds a watch, it's to protect the iPhone business. Apple Watch will make users more dependent on the iPhone and that would make it harder to leave iPhone. But there is a problem with this idea. We have good smartwatches like Pebble that supports with both iOS and Android. So Apple and Google should think about users who don not like to be dependent on OS and try to get them in their business otherwise they would lose them to companies like Pebble.

Apple Watch price

We have three versions of Apple Watch:

- Aluminum Sport edition "starts at $349" and that is for smaller 38mm version. The 42mm size starts at $399
- Glossier stainless steel watch collection starts at $549 and goes up to $1049 Depending on the Apple Watch size and band configuration.
- Gold Watch Edition will be offered at a limited supply at Apple Stores and ranges from $10,000 to $17,000

Screen Size

There are two different Apple Watch models:

- 38mm screen with a 272 * 340 pixels resolution
- 42mm screen with a 312 * 390 pixels resolution

Both cases have retina display.

Figure 2. Screen Sizes for Apple Watch

Battery life

Battery life is the most significant features users pay attention to. Apple Watch group said the watch would last all day long. But later they said users would have it for 18 hours and then they should charge it overnight. [9]

- All-Day Battery Life (Up to 18 hours)

    - All-day battery life is based on 18 hours with the following use: 90 time checks, 90 notifications, 45 minutes of app use, and a 30-minute workout with music playback from Apple Watch via Bluetooth, over the course of 18 hours.
- Talk Time Test (Up to 3 hours)
    - Apple Watch and software paired with an iPhone using pre-production software with a call placed from Apple Watch.
- Audio Playback Test (Up to 6.5 hours)
    - Apple Watch and software paired with an iPhone using preproduction software with music playback from Apple Watch via Bluetooth.
- Workout Test (Up to 6.5 hours)
    - Apple Watch and software paired with an iPhone using preproduction software with a workout session active and the heart rate sensor on.
- Watch Test (Up to 48 hours)
    - Apple Watch and software paired with an iPhone using preproduction software with 5 time checks (4 seconds each) per hour.
- Power Reserve (Up to 72 hours)
    - If battery gets too low, Apple Watch automatically switches into Power Reserve mode so users can continue to see the time for up to 72 hours.
- Charge Time
    - About 1.5 hours to 80%
    - About 2.5 hours to 100%

Before March 9th there were a lot of questions about what will happen after the battery gets too low? Now we have the answer: power reserve. If the watch does not show the time, nothing else matters. Because that is the main reason people have it. Battery is the most important element and so far Pebble has the best battery life.

Apple Watch uses The MagSafe style charger magnetically. It attaches itself to the back of the watch, and starts charging the device.

Apple Watch Connectivity

The Apple Watch has Wi-Fi and Bluetooth, so it can connect to a phone or a Wi-Fi network. It can also piggyback onto iPhone's Wi-Fi or GPS. Apple Watch can connect to an iPhone 5, 5C, 5S, 6 or 6 Plus. Sadly, earlier iPhones are excluded.

Communication

- Sketch
    - Users can draw quick images to a friend, and she/he will be notified with a simple vibration on her/his hand.
- Tap
    - Users can get their friends attention with just a tap on the watch. Is Apple trying to bring back Morse code?
- Animated Emoji
    - The little graphic images can be customized. Users can send a smiley face, but they can also make it wink, or stick its tongue out with the animated new feature.
- Receive calls and texts
    - It can accept texts and calls. Apple Watch has a microphone and a speaker. Because other people can hear, users may want to avoid using this feature in public areas. But, it is a good feature when it comes to driving or biking.
- Analyzes texts
    - The watch looks at incoming texts and analyzes the content. Based on that, it can suggest quick replies.
- Walkie talkie
    - Users can send quick voice messages to other users.
- Heartbeat
    - Press two fingers on the screen, the built-in heart rate sensor records and displays user’s heartbeat. Users can also share it with their friends. It is a good feature for long distance relationships. This is a market point for Apple Watch.

User Interaction

There are four different ways to interact with Apple Watch:

- Simple touch
  - A tap on a button is a typical example.
- Force touch
  - Force Touch allows the device to distinguish between a light press and a deep press.
- Gesture
  - Vertical swipe to scroll, horizontal swipe to navigate between pages, left edge swipe to navigate back
- Digital crown:
  - The Digital Crown also acts as a home button. Everyone agreed that actually using it to zoom and pan between apps takes some getting used to.

Fitness

Apple Watch is a perfect fitness tool. It has numerous features that connect to Apple's Health app and personalize it to the user. On March 9th Apple introduced ResearchKit. With ResearchKit, researchers can build out a medical testing app for iOS that is accessible to people far from their physical lab. Users can sign up with a digital signature, and instantly start recording data. This new tool is a great match for Apple Watch and it can help a lot of people and teach them about their health. Pretty soon there will be awesome applications about personal health, and users will need an Apple Watch to use them. The early partners tell Bloomberg that they got thousands of volunteers within a day of launch, including 11,000 for a Stanford University cardiovascular trial -- for context, Stanford says that it would normally take a national year-long effort to get that kind of scale. [10] The combination of Apple Watch and ResearchKit will make great new products.

- Measures heart rate
  - Using a custom heart rate sensor located on the back of the watch, users can measure their heart rate at any time. Using the Watch's Taptic Engine, they can also send to a friend their heartbeat.

- Accelerometer
    - Apple Watch counts steps, calculates the calories burned and generally measures total body activity.
- Set workout goals
    - User can set personal goals and the app will summarize their achievements.

Miscellaneous

- iTunes and Apple TV access
    - Apple watch allows users to access their iTunes library and listen to iTunes Radio, and access Apple TV with a simple swipe. The display is not big but it is fun to have a small display to see short videos. There is also a music control feature, which allows users to pause, play or skip songs, in addition to raising and lowering the volume. Users can also control their iPhone camera from their watch. It makes it easier to get a selfie. But users still need a selfie stick for fun pictures.
- Siri
    - Of course users can talk to their wrist. She is with users on their wrist. Using the watch's microphone, users can access Siri and use it to dictate a text, or ask questions.
- Maps
    - Users can ask Siri for directions, and also use the new Maps feature on the Watch. She will direct users where to turn as they walk, like an invisible guide. Apple is going to use different types of vibrations to deliver its messages. What this means is that one type of vibration could mean turn left, and another type of vibration could mean turn right. Apple claims that this allows navigation without looking at the screen. This feature is awesome. It can help blind people to find their way or it can be a tour guy for them.
- Apple Pay

- This is a unique feature Apple Watch has so far. Apple Pay can be used at various businesses, such as Whole Foods, Duane Reade, Macy's and more. To pay with Apple Watch, “just double-click the side button and hold the face of the Apple Watch near the contactless reader”. Apple says in a press release that the Apple Watch will let iPhone 5, 5C, and 5S users use Apple Pay. That means people would not have to upgrade to the latest phone to use Apple's contactless mobile payment system. It has NFC chip so users can tap it against contactless point-of-sale systems at cash registers to make a payment.

- Apple Key
  - Users can use NFC technology to open their hotel’s room. Apple Watch can unlock a door right from the notification from the watch. By pressing ‘unlock your door’, users can bypass the front desk entirely, go to the room and that is it; the watch is a room key. Users can wave it in front of the door and welcome.
- Sensitive display
  - Raise your wrist, the watch display will automatically turn on. There is an incoming call and it is time to ignore it. Simply cover the display with hand. Apple Watch is very sensitive about wrist. It would be awesome if Apple can make default applications to protect users from rubbery or getting mugged. Users simply need to make a pattern for their wrist. So users can move their wrist based on that pattern and Apple Watch would call 911 and send them their locations.

How about left-handed users

Apple Watch works just fine for lefties, because the display rotates. All to do is flip the watch over so the Digital Crown is on the left side. Then swap out the band so it’s also in the right position, and then strap the watch to the right wrist.

When does it ship?

The Apple Watch ships April 24th, but preorders begin April 10th. That is also the same day Apple Stores will have Watches on hand for people to try on, in case users are not sure which combination of case and band is right for them.

Architecture

The WatchKit framework is like iOS framework in the early days. It is limited, buggy, and different to work with it, but developers had to learn how to use this new shiny cool framework. The primary framework, called WatchKit, was bundled with Xcode 6.2. This framework supports both Swift and Objective-C. An abstract definition of WatchKit is a group of classes and an interface builder that developers can wire them and get an Apple Watch app.
[1]

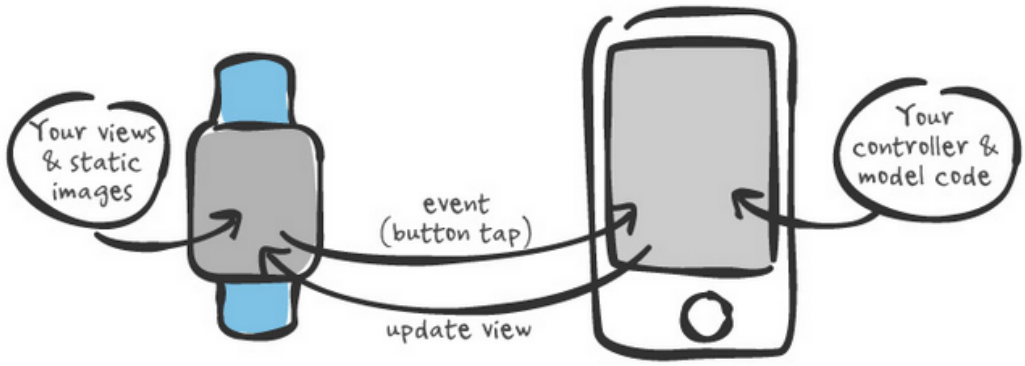

Figure 3. Apple Watch Architecture

WatchKit apps

Apple Watch connects to the iPhone to have access to the Internet and GPS data. This means the iPhone will execute all of the code for the Watch apps. Even though the Apple Watch has its own version of iOS, it does not run any of the code. So be careful. If a code takes a while to run, the watch will just be sitting there not doing anything. In other words, Apple Watch is just another display for now. Consider MVC model, Apple Watch is the view and model and controller are on the iPhone. Developers cannot make

[1] http://www.raywenderlich.com/89473/watchkit-initial-impressions

any independent applications for the watch. Apple promises developers can make native applications for the watch in later versions of WatchKit.

Storyboards and resources

The Watch only stores local resources, like images and files. Images downloaded from the web will have to be downloaded to the phone and then sent across to the Watch. Because data is transferred between the Watch and phone over the air, expect some natural delays. Nothing should ever take more than a few hundred milliseconds to compute and transfer, though. Segues work in WatchKit very similar to the way they work in iOS: They need their own identifier string, and developers create them by Control-dragging between controllers in Interface Builder.

Sharing data

WatchKit uses exactly the same mechanism introduced in iOS 8 to share data between extensions and the parent application. So instead of giving the Watch its own Core Data stack, syncing it with iCloud or enabling it to manage its own file system, simply need to use a **shared app group**. The main app manages syncing and updating all of the data in the shared app group.

UI Controls

1. **WKInterfaceButton**: To Create a standard button.
2. **WKInterfaceDate**: It is a label built to display dates and times.
3. **WKInterfaceGroup**: To handles all of the interface layout and grouping.
4. **WKInterfaceImage**: This subclass is almost exactly the same as UIImageView.
5. **WKInterfaceLabel**: This is just like UILabel.
6. **WKInterfaceMap**: Maps on WatchKit are not interactive.
7. **WKInterfaceSeparator**: Can work in tables and views, and that even for vertical separation.
8. **WKInterfaceSlider**:
9. **WKInterfaceSwitch**: This object is also similar to its iOS counterpart, UISwitch.
10. **WKInterfaceTable**: Tables are row-based and have no notion of sections.
11. **WKInterfaceTimer**: Configure what units to display: seconds, minutes, hours, days, weeks, months and even years.

Layout

Auto Layout is gone. WatchKit introduces an entirely new layout system. The WatchKit layout system defaults to sizing objects based on their content size and putting everything into either horizontal or vertical layout groups. There are only three new concepts to get started building interfaces: **groups**, **content sizing** and **relative spacing**. One of the main reasons for using this new method is that: Apple Watch has a small display and it is expensive. This new layout system is limiting, but on the other hand it makes it so much easier to deal with the layout.

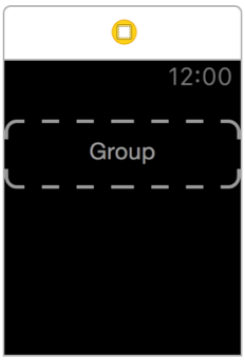

Figure 4. An empty WKInterfaceGroup

Animations

When Apple announced the Apple Watch, there were a number of beautiful animations. Developers thought Apple would introduce some tools for making these kinds of animations, but the WatchKit extension has nothing to offer in the animation area. It handles animations in a very non-dynamic way. To display animations in WatchKit, developers will need both a WKInterfaceImage and a series of images. Combination of those images will create a smooth animation just like a GIF. Some developers have already started working to find a better way to deal with animation in WatchKit. Flipbook is one of them.

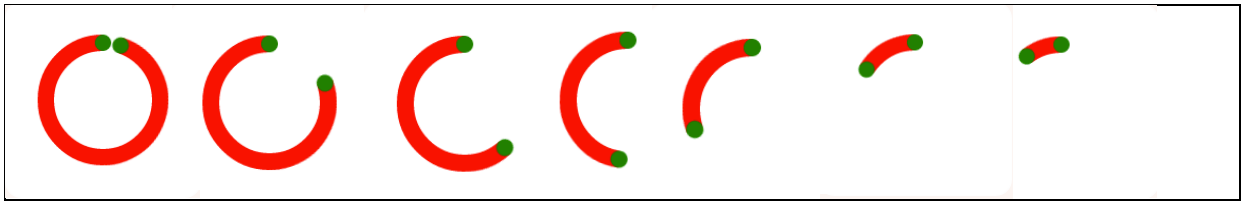

Figure 5. Example of individual progress fill animation frames

Navigation

WatchKit uses familiar concepts and takes the following forms:

- **Hierarchical**: Similar to UINavigationController.
- **Page-based**: Similar to UIPageViewController.
- **Modal**: Any type of presentation or dismissal transition.

Developers are strictly limited to these three methods. There is no custom navigation. In fact, one cannot even mix and match hierarchical navigation and page-based navigation. However, use modals with either type.

Contextual Menu

Apple introduced a new and simple API for creating context menus for WatchKit. Instead of creating menus from scratch and wiring up delegates or blocks, just create menu items in Interface Builder and wire up their actions just like one would with buttons or other interface objects.

Glances

Glances are an extension of a watch app that presents important information to the user needs right away; information users can consume immediately without first having to launch the Watch app. Glances are the WatchKit equivalent of today's extensions in iOS 8. They are read-only, single page, and non-scrollable views. There is no user interaction, except for the tap event. If users tap on the glance, it will simply launch the app, optionally providing a context to let the app display a specific interface controller. To see glances, the user swipes up from the bottom of the Watch screens, which displays any available glances. If the user wants to customize glances, that can be accomplished with the iPhone. Glances are template-based, meaning developers do not have total

control over their appearance. Their interface is split into upper and lower layout groups. Remember each Watch app can only contain a single glance.

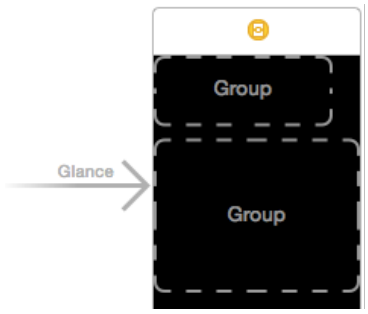

Figure 6. Glance interface controller

Handoff

The users picks up right where he left it off. Handoff is a way to facilitate the seamless transfer of tasks between two devices. It lets a user instantly continue an activity he start on one device on another, without the need to launch the same app on that second device and load a file, or wait for iCloud to sync and so forth. Here is an example: A user start reading an email from her iPhone and pick up where she left off when she sit down at her Mac. Handoff broadcasts activities using Bluetooth Low Energy and transfers them using Wi-Fi, either directly or via iCloud. Bluetooth LE is required because it both uses very little power, and it only works within a very close range.

Notifications

Local and remote notifications enable an app that is not running in the foreground to inform users about new, relevant information that is available. If the watch receives the notification, it will notify the user via a quick vibration. There are two types of notification in WatchKit: short look and long look. The Short Look provides a soft, minimal amount of information. If she lower her wrist, the Short Look interface disappears. The Long Look interface appears when the her wrist remains raised or when she tap the Short Look interface. She will have more detailed information and more functionality and at the end she should dismiss it.

There are two kinds of interfaces to create a notification.

- Static
    - Static interfaces are for the time when dynamic version of the application does not work. Developers can only configure a static notification interface in the storyboard. That means state notification cannot run any code to update its contents or configure its interface.
- Dynamic
    - With dynamic notifications, developers can customize the interface a bit more, but if it takes too long, the system will fall back to the static notification. It can add any action items to the notification, so the user can jump into an application immediately and have it carry out the corresponding action.

Hello, Apple Watch

What a developer need to develop an Apple Watch app:

- A Mac running OS X Mavericks or later.
- Xcode 6.2 Beta 5 or later.
- To run the samples on physical hardware, there should be an iPhone running iOS 8.2 or later, an Apple Watch and a paid membership to the iOS development program.

Getting started

Create a **New Project** in Xcode and select the **iOS\Application\Single View** Application template. Set the **Product Name** to **HelloAppleWatch**, make sure the **Language** is set to **Swift, Devices** is set to **iPhone, Core Data** is unchecked, and click **Next**. Choose a folder in which to save your project and click **Create**.

In the iOS Simulator menu, select **Hardware\External** Displays and then select one of the Apple Watch screen sizes. A small black window appears:

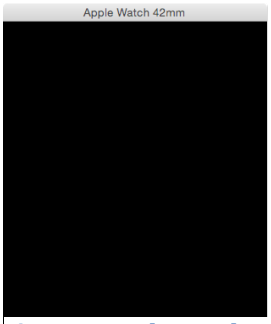

Figure 7. Apple Watch screen on simulator

Developers will interact with the watch display in this secondary window.

Apple Watch app needs a storyboard and an interface controller. To create these, add a Watch app target to a project.

Adding a Watch app target

Select **File\New\Target...** from the Xcode menu. In the target template window, select **iOS\Apple Watch\WatchKit** App:

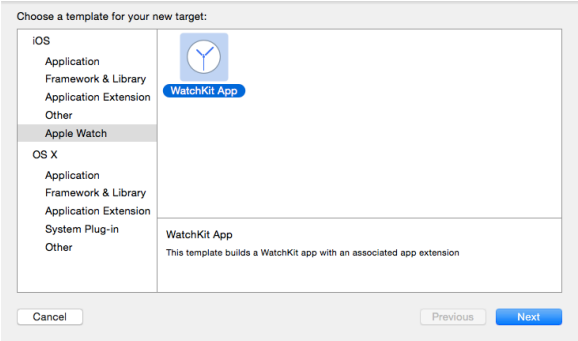

Figure 8. Target template window

Click **Next**. In the target options window, uncheck **Include Notification Scene** and leave everything else as it is. Click **Finish**. A pop-up window appears:

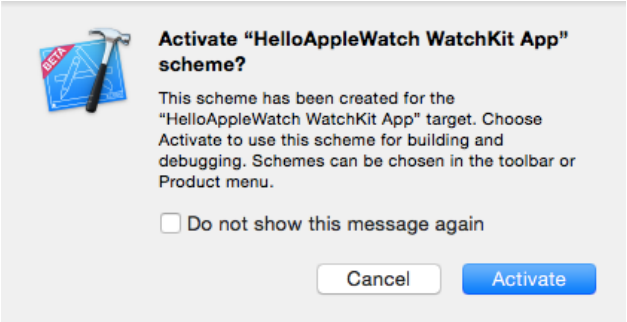

Figure 9. Active WatchKit app message

Click Activate. This is the key step to making the Watch app appear in iOS Simulator.

Two new groups now appear in the **project navigator**: a **WatchKit Extension** group and a **WatchKit App** group. Open these to see that **InterfaceController.swift** is in WatchKit Extension and **Interface.storyboard** is in WatchKit App.

Showing the Watch app in the simulator

Open **Interface.storyboard** and then select the interface, either in the storyboard canvas or in the document outline. Then, in the **Attributes Inspector**, change the **Color** to something other than **Default**. This will change the background color of the top-level interface.

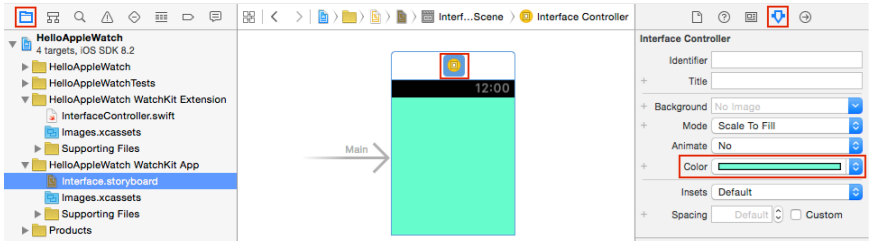

Figure 10. Change the background color

Next, check that the active scheme is set to **HelloAppleWatch Watch App. Run** the app. Is there a small green display? Good job.

Hello, World

Open **Interface.storyboard** and select the interface. Change the background color back to **Default** to start. Drag a **label** from the Object Library onto the single interface controller:

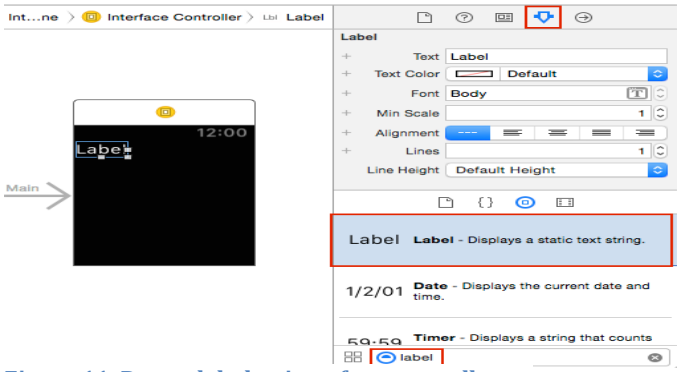

Figure 11. Drag a label to interface controller

By default, the label sits at the top-left of the interface. Select the label and go to the **Attributes Inspector,** change the **Text** to **Hello, World** and change the **Alignment** to **Centered**. The label resizes to fit the new Text value. **Build and run** the Watch app.

Figure 12. Hello, World!

The Android Wear Watches

- **LG G Watch**
- **Samsung Gear Live**
- **Motorola Moto 360**

What is Android Wear

Android Wear is a version of Google's Android operating system designed for smartwatches and other wearables. It comes with a new user interface—the result of Google working to understand how we use our phones today and how we can be more in touch with our environment.

Moto 360

The Moto 360 is the Android Wear running smartwatch that received a lot of attention before the Apple Watch was announced. When Motorola first presented the 360 earlier in 2014, it was the only company that has traditional round case and watch face. With the upcoming arrival of the LG G Watch R, though, it will not be the only round Android Wear watch for much longer. Motorola believes people want to wear a device on their wrist that looks like a real watch, and most people think of watches as being round. Because of that Motorola decided on the round shape of its smartwatch.

Figure 13. Moto 360

Release date and price

Moto 360 launched in the U.S. on September 5, 2014. The Moto 360 price in the U.S. is $250. That is $100 less expensive versus the entry-level Apple Watch.

Specification [11]

- Processor: TI OMAP 3630 single-core ARM Cortex-A8
- Display: 1.56-inch LCD (320 x 290, 205 PPI)
- RAM: 512MB
- Storage: 4GB
- Battery: 320mAh
- Android Wear Version: 4.4W, upgradeable to 4.4W.1
- Size: 46mm diameter, 11.5mm tall
- Weight: 49g
- Connectivity: Bluetooth 4.0 LE, Qi wireless charging
- Sensors: 9-Axis (Gyro, Accelerometer, Compass), Pedometer, Optical heart-rate monitor
- Protection: Dust and Water Resistant (IP67), Gorilla Glass 3

Design and Display

The Moto 360 is round while the other watches are square. This feature alone made Moto 360 unique among other Android devices. The body of the watch is stainless steel with very clean lines and a beveled edge that leads up to the 1.56-inch Gorilla Glass screen. On the left side is a crown button, but it is just a button (it does not spin). The button is metal as well. The attention to detail here is amazing. Pressing the button can wake or sleep the watch, and a long-press will open the settings menu. The back of the watch is plastic, but it feels solid. In the center is the optical heart rate sensor. The Moto 360 will be offered with a variety of metal and leather band options. The Moto 360 is big, but not too big if users have average or small wrists. Women with smaller wrists will definitely find it is a bit on the large side, though. The Moto 360 tends to look more like a traditional wristwatch, but on top of that, it is also really nicely designed.

Screen

The Moto 360 uses a 1.5-inch backlit LCD IPS display. It has a 320 x 290 resolution with 205ppi pixel density, so it should be roughly the same screen quality as the G Watch and the Gear Live, and it is not as high of a quality when its time to display images. It is brighter than rival watches. It has a light sensor at the bottom of the screen so it can change the visibility of the screen in the bright outdoors. If users are not moving they can launch apps perfectly well. On the move, however, it is a slightly different story and it can take the extra swipe or press to get it to work.

Content Gets Cut Off on Round Display

When Google announced Android Wear they actively promoted the fact that it could work on both square and round displays. It does indeed work on both shapes, but the experience is not the same. Sometimes content gets cut off by the round shape of the Moto 360 display.

Battery life and Charging

Moto 360 has an ambient light sensor, which detects how bright the environment is and adjusts the display brightness accordingly. The idea is to maintain a balance between being able to see the display well, and not overpowering it and wasting battery life. On occasional days it can hold 10 or 11 hours of battery life, but we have also had a good number of days where the same amount of time left us with more than half the battery remaining. The Moto 360 uses the Qi wireless charging standard. The Moto 360's battery life was never inefficient, but it wasn't great either, thanks in part to its wildly old and not particularly efficient processor.

Fitness

All Android Wear watches will be able to share information with Google Fit, and every device can get in on the party. Basic watches like the LG G Watch will be able to share step data, while more advanced smartwatches with heart rate sensors and GPS will be able to share more. This means users will be able to track runs and workouts even when they are on a phone or tablet is at home. Google Fit promises to aggregate data like

heart rate, steps taken and blood glucose, and to connect with our favorite fitness apps for real-time speed, distance and time data while walking, running and cycling.

Controlling Music

Android Wear can control media playback on connected handsets. Users can do things like start playlists on-the-fly, skip tracks, pause music and even apply different audio effects with just a few taps.

Smart Documents

Users can travel or participate in any activity that needs documentation much easier with Android Wear smartwatches. He can quickly pull up his document to be scanned, or viewed. He can purchase movie tickets online, show up at the theater and get in just by having the cashier scan a barcode on his watch. Or, perhaps he could get through airport security by tapping his NFC capable watch to designated readers, which will scan his travel information that instantly gets transmitted via a wireless signal. Google demoed smart watches being used in combination with Google Wallet, and QR codes at I/O.

Sleep Tracking and Convenient Alarms

Android smart watches will sit on the wrist monitoring sleep patterns through integrated sensors. They can accurately predict when users will wake or when they have had a healthy amount of sleep. Waking up to an alarm is no fun for people around us, but we need to do it sometimes. The best part is they can launch a silent alarm by vibrating the device on their wrist.

Realtime Weather and Traffic Updates

Android smartwatches will offer this information via Google Now. Smartwatches will deliver this information right to her wrist. With Google Now voice support, she will also be able to call upon the updates by talking to the wearable. Voice alerts will allow the device to announce information out loud.

Locate Android Devices

One of the most compelling features added for Google and Android users is the Device Manager. Android Device Manager dashboard can ping, locate, lock or even swipe a phone. Users have also seen many smartwatches that include proximity alarms, which is likely something that will be supported in Android Wear too. Any time the user walks away from their paired device while the feature is active, the smartwatch will issue an alarm. This feature saves users from abandoning their phone somewhere, like at a bar or restaurant.

Chromecast and Google TV Support

Smartwatches will be compatible with Chromecast and Google TV devices, it can interact and control with those devices remotely.

Smart Home Controls

Google purchased Nest, best known for their smart thermostat. Opening the Nest platform to outside developers will allow Google to move into the emerging market for connected, smart home devices. Use the Hue Control app to turn on and off lights. A tap on a button or a Star Trek-style voice command will turn on all of users Hue lights or just a specific group or room.

Voice Dictation Support

Users need to say the "OK Google" command first or tap on watch's screen so that it is listening. Just by talking, she can send messages; compose e-mails and type, without any interface that uses the Android keyboard. Wear provides two types of voice actions:

- System-provided
    - These voice actions are task-based and are built into the Wear platform. Examples include: "Take a note" or "Set an alarm".
- App-provided
    - These voice actions are app-based, and developers declare them just like a launcher icon.

One thing is missing, the ability to hear answers audibly. Android Wear devices do not do this because it does not have a speaker. Instead, users will get the response as cards and notifications.

Interacting with Wear Watch

Here are some ways to interact with the Watch. A user can:

- Tap an on-screen item to select, launch it, or open it.
- Swipe upwards to scroll through notifications.
- Swipe right to dismiss notifications or apps.
- Swipe left to reveal more details about a notification, and left again for more actions. Long-pressing the clock faces to choose a new one.
- Cover the entire screen with the hand to dim it.
- Swipe all the way down from the top to the bottom of the screen to mute the watch and perform the same action again to unmute it.

Navigating the Notifications

Cards are notifications from the phone, Google Now cards, or prompts from Wear apps. No matter where the card comes from interacting with them is the same. Swipe from the left to remove it, or swipe from the right to get more options. It is a simple system that needs to be a little more specific for user's convenience. Prior to Android Wear 5.0.1, if she received a message, an application notification, or had any other card type appear on her watch, swiping it to the right would dismiss it with no option to get it back. 5.0.1 introduces a way to quickly undo the latest notification dismissal and save all that trouble. To show notifications swipe from the bottom of the screen up to reveal the last dismissed card. Sizes of cards are inconsistent. On some watch faces the card is small and discreet, but on others it takes up half of the display.

Wear Watch Standalone Functions

There is some stuff can do with the smartwatch even when it is not paired up with a smartphone. Android Wear watch comes with a number of built-in watch faces from which users can choose by tapping-and-holding the current watch face screen and then swiping left and right to choose a new one. Android Wear watch also has a standalone

pedometer function that counts users daily steps, and also applications like “Set a timer” and “Start stopwatch” as well.

Glympse

It is a social location application. The basic idea is she can select a contact with which she wishes to share her current location. The recipient receives a message containing a hyperlink which, when clicked, leads to a map showing her whereabouts.

The World Of Android Wear

Android Wear is a version of Google’s Android operating system designed for smartwatches and other wearables. It comes with a new user interface, the result of Google working to understand how we use our phones today and how we can be more in touch with our environment.

Android Studio IDE

Android Studio is Google development environment for Android. Based on IntelliJ IDEA, Android Studio is the foundation for the next-generation builds systems for Android apps.

Android Wear Architecture

Android Wear apps have two parts. The first part would be the app running on the watch, and the second part is the app running on the paired smartphone or other handheld device. There needs to be a communication mechanism between the two. It is recommended using the Wearable Data Layer API for this purpose, and developers are discouraged from attempting to create their own direct communication channels using low-level sockets.

Cards

One of the most important UI patterns in Android Wear are cards, simply because they are used so frequently. Developers think of a card as being a container used to present information to users with a consistent look and feel across the rest of the platform. It is to display a title, a description, and an icon. Cards hold a collection of widgets using a custom layout. Whether displaying the weather forecast, GPS navigation,

the estimated commute to work, or anything else relevant to reading, cards are employed in a wide range of activities. In fact, Android Wear notifications are also displayed on cards. Only one card is displayed at a time and background images are used to convey additional visual information.

Figure 14. Default card layout

Anatomy Of a Notification

Let's start by describing the basic parts of a notification and how they appear on different devices. At a minimum, all notifications require an icon, a title, and a description. Android supports optional actions that are displayed at the bottom of a handheld notification and as a separate action icon on a wearable.

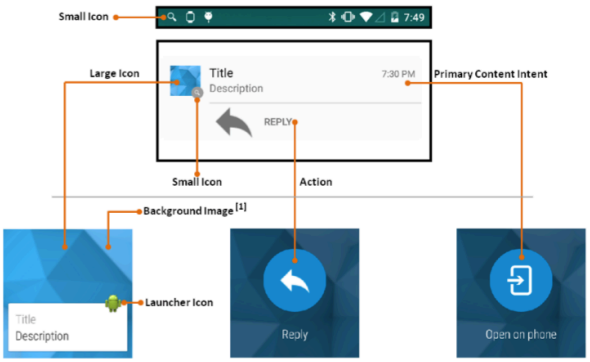

Figure 15. Anatomy Of a Notification

Data Layer API

Data Layer API creates native wearable apps that can communicate securely with a paired smartphone. The Wearable Data Layer API is a part of Google Play services and has an important role. For example, to conserve battery and reduce wait time, developers can offload computationally intensive tasks to the handheld and then send the results back to the wearable. In Android Wear, there are three basic APIs that are callable on both sides and facilitate the sending and syncing of data in addition to discovering nodes on an Android network.

- Message API
- Data API
- Node API

Message API

The Message API provides a low-latency, one-way communication mechanism for sending byte arrays. There is no acknowledgement for sent data but there is a way that the receiver can send a confirmation back through the same mechanism. This API is good for remote procedure calls such as controlling a handheld's media player from the wearable or starting intent on the wearable from the handheld.

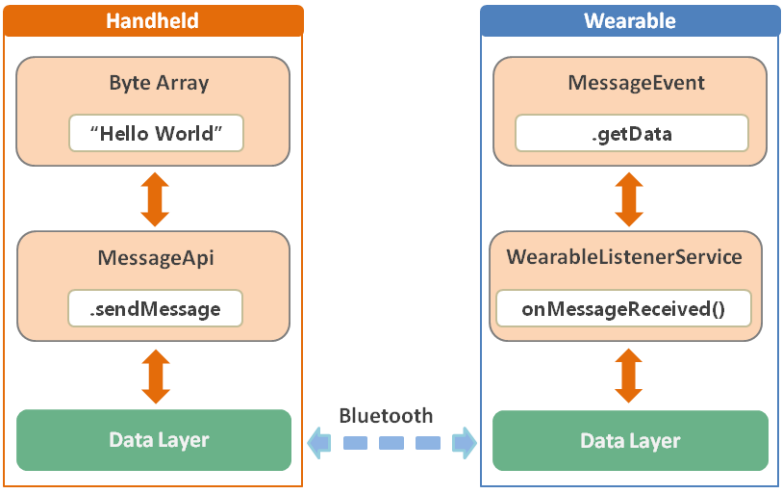

Figure 16. Message API Diagram

Data API

Data API allows the syncing of data between a handheld and a wearable. Data pushed using this API is shared between devices and is available to both of them. Data Layer API is a shared data container with the ability to put data in, take data out, and be notified when this data changes on either side. Developers do not need to worry about devices disconnecting intermittently, because the system supports the caching of data and will synchronize it whenever the connection is re-established.

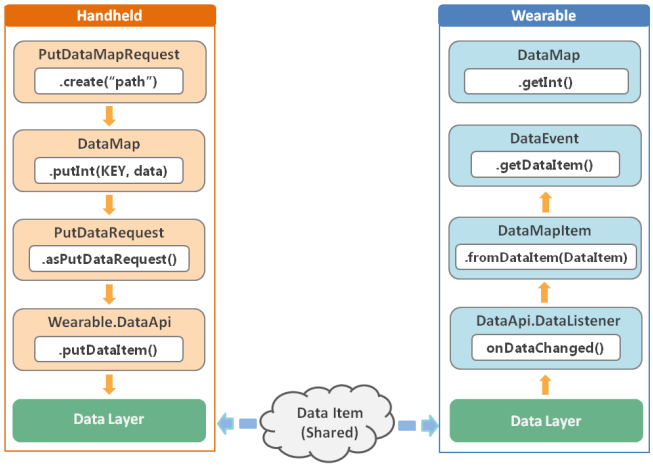

Figure 17. Data API Diagram

Node API

The Node API is used to handle events related to local and connected device nodes. It keeps track of device connections via the NodeListener interface method and is therefore useful for discovering nodes that compose Android network. This API also provides a method for retrieving a list of all the connected nodes.

Hello World

1. Open Android Studio and create a new project by clicking on "Start a new Android Studio Project".

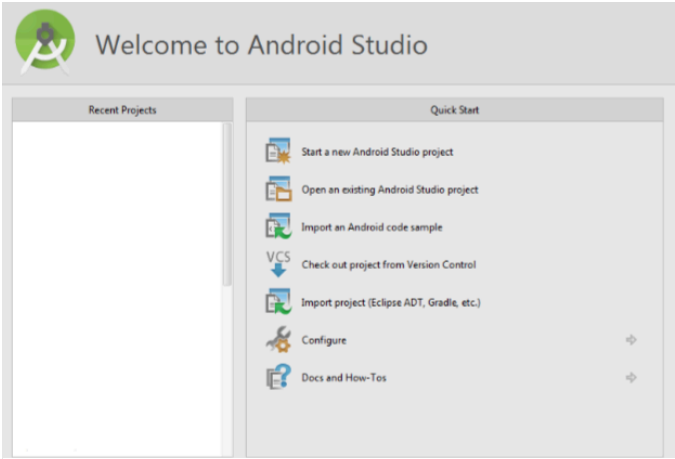

Figure 18. Welcome page

2. In the Create New project window, name the application "HelloWorld" and key in a company domain.

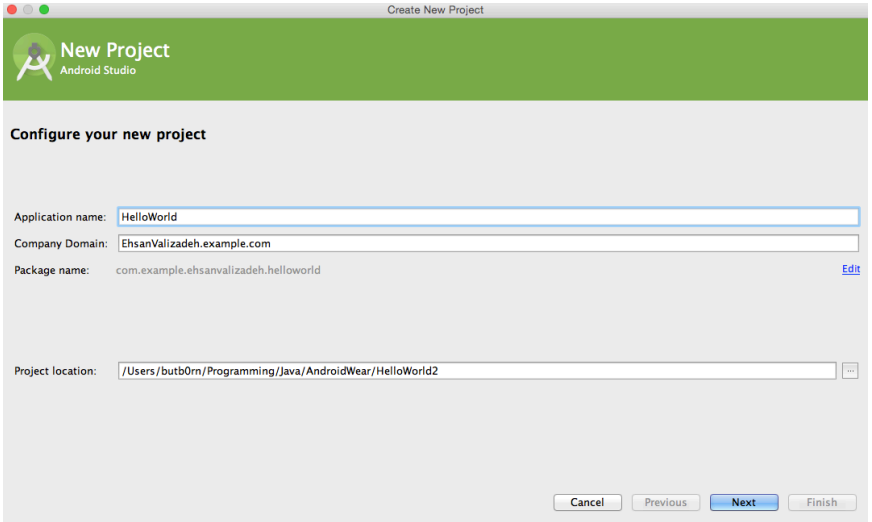

Figure 19. Application name

3. Click on the Wear platform and select API21 as the Minimum SDK. ClickNext.

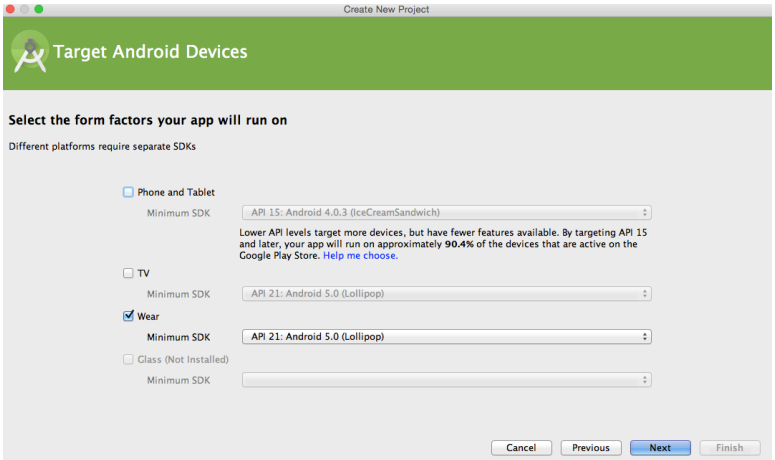

Figure 20. Select platform

4. Select Blank Wear Activity then click Next.

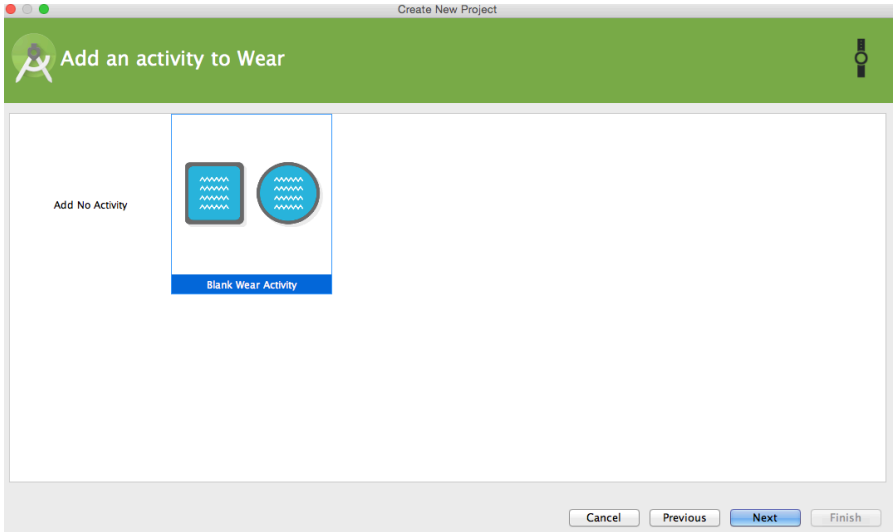

Figure 21. Add Activity

5. Fill in the application details for the wearable activity. Click Finish.

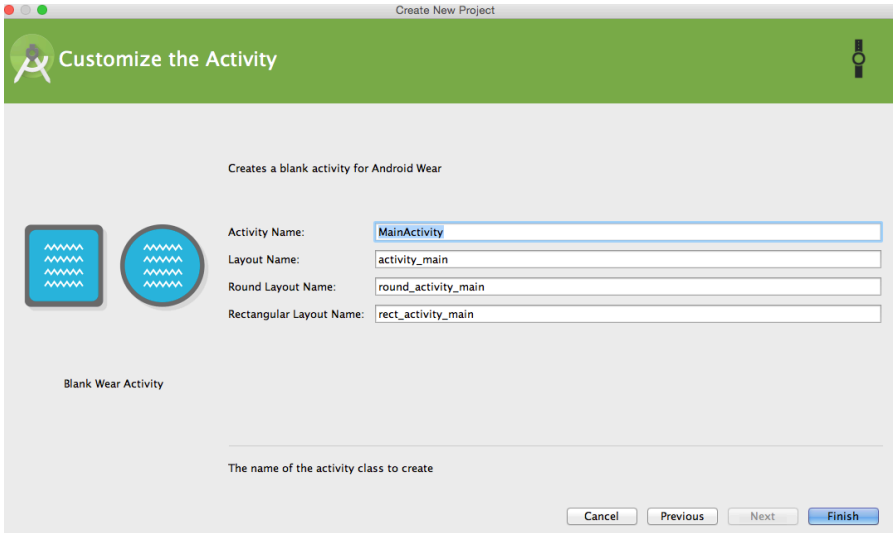

Figure 22. Select the name of the activity class

6. The Android Studio IDE should now look like Figure 23.

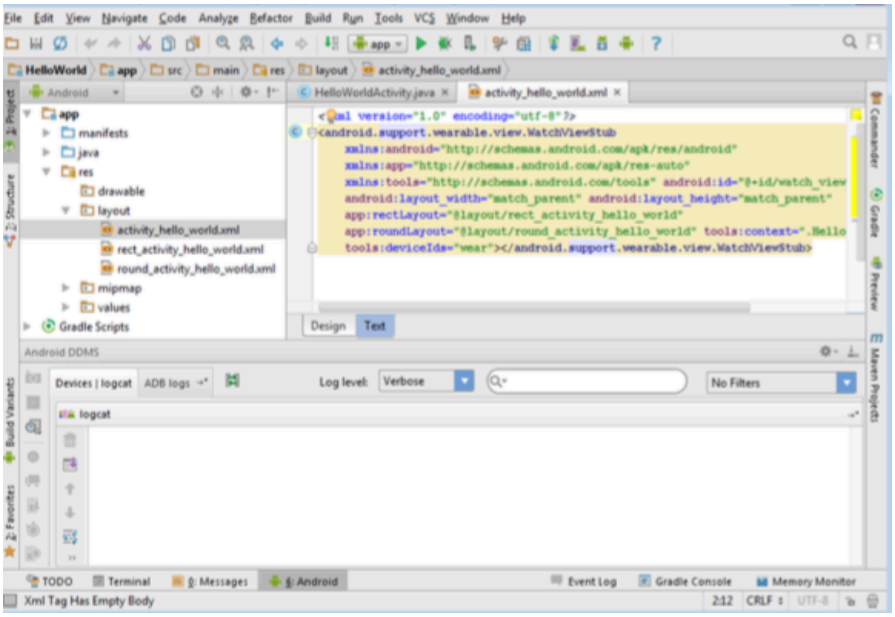

Figure 23. Android Studio IDE

7. Launch Android Wear Square.

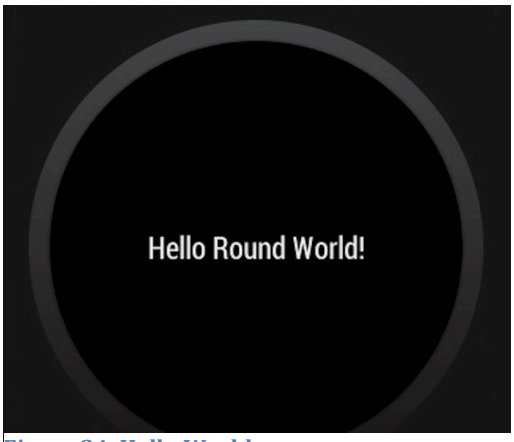

Figure 24. Hello World

Apple Watch Vs. Android Wear

Apple is trying to do a lot with its smartwatch platform. There is a maps app, photos, and interesting watch faces. It plans to have a high-resolution screen on the Apple Watch, and the digital crown works as a zoom controller. On the other hand, Google has a simple definition on wearables. Android Wear is designed to do certain things that Google thinks users would want to do on a watch. That does not include looking at maps or reviewing pictures. Users have a phone for that. Instead, Android Wear is Google Now for her wrist, plus a powerful developer toolkit that allows third parties to pick up where Google left off.

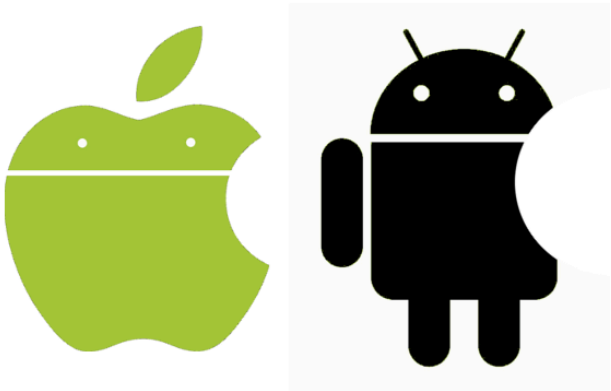

Figure 25. Android Vs. Apple

WatchKit Vs. Android Wear API

WatchKit SDK is a strange thing. It is SDK for a device is not in market, and never been used. But a lot of people are trying to update their apps to be the first on the new market. Since there is no device yet, they are working with some "imaginary" device that will solve all the user's problems.

- WatchKit's architecture makes it impossible for a developer to make mistakes and end up implementing something that will drain the battery. The framework is very strictly sandboxed in this first release. Android Wear is much more flexible.
- WatchKit apps can transparently access the web, via the WatchKit extension running on the phone. To achieve the same thing on Android

Wear, developers need to manually implement a proxy transferring data to the watch via Bluetooth.

Xcode Vs. Android Studio

Android Studio has background compilation and will quickly highlight errors, while Xcode needs an explicit build stage. Both let developers debug on emulators or real hardware. Both IDE offer navigation, refactoring, debugging, etc. Both Android Studio and Xcode are free. It is only when developers want to deploy apps into their iOS devices or distribute them in the App store that a developer have to pay Apple $99 a year.

Programming Languages

Java is easier to learn than Objective-C, which is an older, lower level language. That means a developer have to write more code to do the same thing, and the advanced data structures available to Java developers are well ahead of those in Objective-C. But now, with Swift, it is almost as easy as Java to learn.

Simulators

Working with Apple is way easier. With such a small range of models, they provide just one simulator for the appropriate model. With Android, there are different operating system versions, screen size.

Documentation

Both Android and iOS development are very well documented. Apple has put a lot of effort into explaining many concepts, like View Controllers, and supports both Swift and Objective-C. But Google's docs are very comprehensive as well.

Do not expect Android Wear

- To send a heartbeat to another Apple Watch.
- To scroll and zoom using the digital crown instead of a touchscreen.
- To draw a sketch on a watch and immediately send it to another Apple Watch.
- To send a vibration to another wearer's wrist.
- To add new bits of information, such as the weather, to a watch face instead of just changing it.

- To pay with the watch by simply waving over an NFC payments terminal.
- To send audio snippets as a reply to a text message.
- To view all of application on one home screen.
- To perform different actions by pressing hard or softly on the touch screen.

Conclusion

Apple watches look sharp, at least given the limitations of current technology with lots of watch face styling and band materials. Apple also did the smart thing and announced large and small watch sizes, acknowledging that not all wrists are the same. Deciding which smartwatch users want depends just as much as what smartphone they own. Both Android Wear and Apple Watch offer up tons of functionality and plenty of choices.

Lessons Learned

There were many interesting learning experiences involved in this project. A few of them are mentioned here:

- When it comes to dealing with a new technology you are almost alone on this road. You should deal with technical learning, solving problems and removing unfamiliar obstacles. Apple Watch is really a new gadget and there is not enough resources and material out there. While learning about cutting edge technologies is exciting, the limited amount of resources (books, tutorials, etc.) can be very frustrating. With new technologies, the majority of the time is spent on getting to know the concepts, ideas, problems associated with that technology. When the technology is becoming more and more familiar, it has a potential of being useful in different areas and domains.
- Though these platforms are getting a lot of attention and are interesting, current programming environments are very limited in terms of what features third party developers and utilize in their apps.
- In this project I had this opportunity to create some applications for both platforms. If this was a horserace, my developer instincts tell me to bet on the Apple Watch platform over the Android Wear alternative.
- The quality of the developer tools is just as important as the platform features/capabilities. The tools a developer has give him enough power to create fantastic applications.
- Advertisement matters more than you think. Advertising helps to raise your target demographics' awareness of issues with which they may be unfamiliar as well as educate them on the related benefits of your product or service. When it comes to product demos, Apple has long been king and it is one of the most important elements from the business perspective of their success. Smartwatch has a tricky area: users do not know what they want from this product. But Apple really did a good job so far about introducing their product to users.

- It is amazing how much Apple users respect this company. The Apple Watch sold out in less than 6 hours. [12] This is not just a success for this company; it shows how much users believe in this company and their products.

Bibliography

[1] http://smartwatches.org/learn/what-is-a-smartwatch/
[2] http://www.bidnessetc.com/38371-how-the-pebble-time-blew-away-kickstarter-records-to-reach-20-million-in-fu/
[3] http://www.ccsinsight.com/press/company-news/1957-clear-marketing-needed-to-drive-sales-of-smartwatches-and-fitness-trackers
[4] http://smartwatches.org/learn/privacy-smartwatch-security-and-you/
[5] http://www.businessinsider.com/a-timeline-of-how-the-apple-watch-was-created-2015-3
[6] http://blog.trendmicro.com/smartwatches-create-new-cybersecurity-issues/
[7] http://fortune.com/2015/02/17/apple-watch/?xid=timefb
[8] http://fortune.com/2015/01/27/apple-record-earnings-iphone/
[9] https://www.apple.com/watch/battery.html
[10] http://www.engadget.com/2015/03/12/apple-researchkit-sign-ups/
[11] http://www.androidpolice.com/2014/09/16/moto-360-review-better-than-you-feared-worse-than-you-hoped/
[12] http://9to5mac.com/2015/04/10/almost-all-apple-watch-models-sold-out-applecare-pricing-revealed/