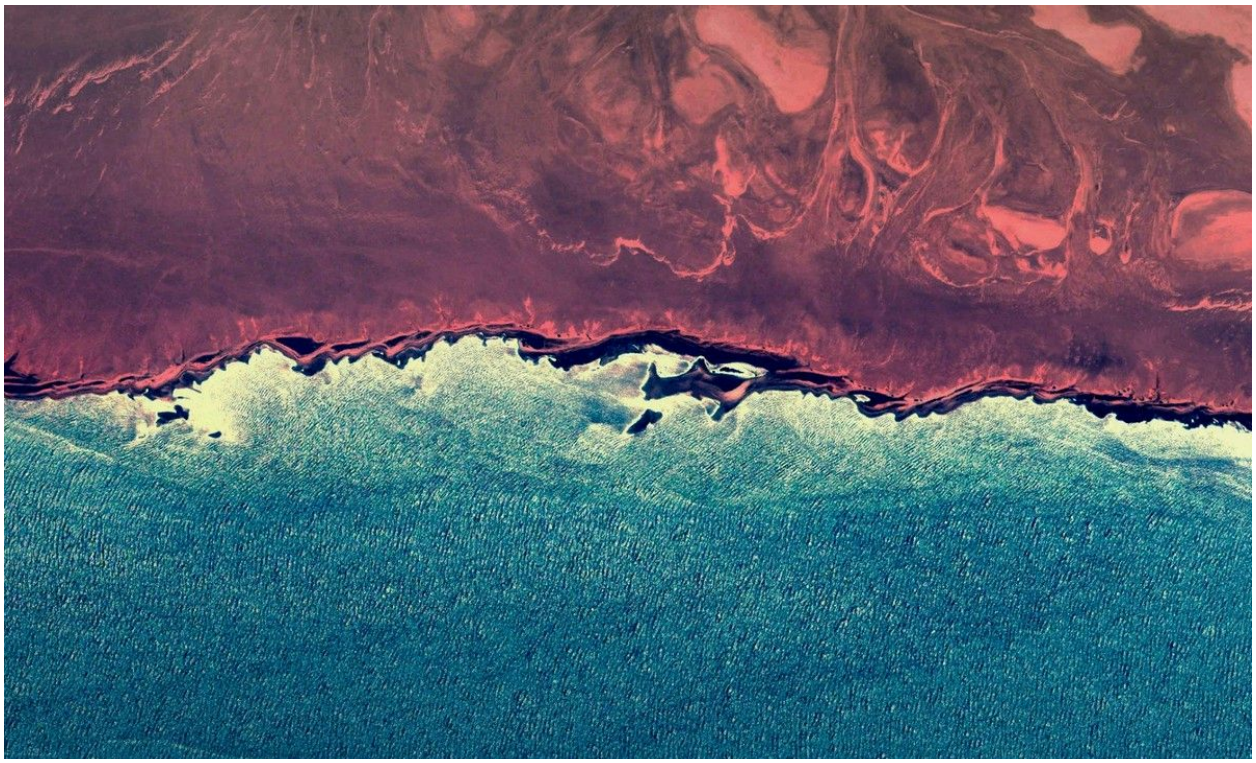


Programming an intelligent watch

Navik - Cycling navigation, with smartwatch compatibility

Project Plan

CSIS0801 Final Year Project, Department of Computer Science, The University of Hong Kong

Lam Ka Fun Gavin | Supervisor: Prof Lau Francis | Programming an intelligent watch | Project Plan

Table of Contents

Project Background

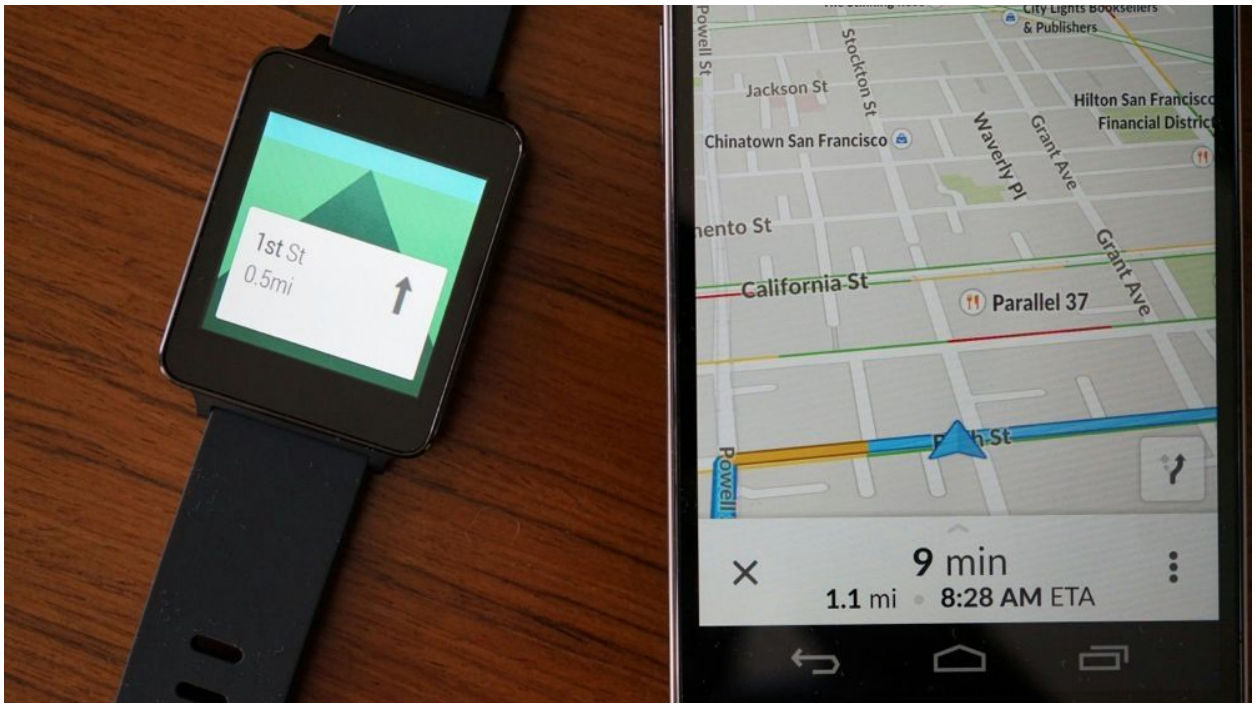
There are various navigation solutions available on the market. However, for cycling? Limited.

If you look towards the ordinary navigation solutions, it does not suit very well. It is likely that you would end up finding yourself riding on a highway or into a cross harbour tunnel. Moreover, such solutions cannot satisfy your unique requirements for cycling. In addition, what if you have no data connection when you are riding in the middle of nowhere?

If you look towards bike specific navigation solutions, make sure you have a deep pocket. A bike computer with navigation functionality would probably costs more than your bike. Besides, why bother purchasing a specific navigation device when we are living in the 'smartphone era'?

As a cycling enthusiast, I proudly introduce *Navik* to you.

Existing Solutions Analysis



Google Maps with Navigation

Google Maps is a cross platform online application with comprehensive maps throughout the globe. Voice-guided navigation is available for driving, public transit and walking. Live traffic condition, street view and point of interests information are also available.

Advantages

- **Free** of charge
- Compatible with **smartwatches**
- **Route construction** with **intermediate destinations**
- Provide **maps** and **turn-by-turn navigation** display

Disadvantages

- Cannot load **custom routes**
- No **offline navigation support** and limited **offline map data**
- Not support **round trip navigation**, which is important for cycling
- No **cycling specific navigation** and related functionalities available

Garmin Bike Computer with Navigation

Garmin produces bike computers of different levels and prices. Model with navigation functionality ranges from USD$249.99 to USD$599.99. User can connect the device for data transmission with computers via cable or the companion smartphone app via Bluetooth.

Advantages

- **Feature rich** bike computer
  - Compatible with wireless bike sensors (not included), such as speed sensor, cadence sensor, heart rate monitor and power meter
- Provide **cycling specific route construction** with **round trip** and **intermediate destinations** support
- Support **external routes import**, such as GPX files
- Provide **maps** and **turn-by-turn** display
- Provide **route analysis**, such as distance and elevation data
- **Offline map data** by OpenStreetMap

Disadvantages

- **Expensive** for casual and amateur riders
- Require **manual update** for system and map data
- **Closed** and **proprietary** system, thus the device is very specific and cannot have other use

HammerHead

HammerHead is an innovative navigation device mounted on your bike. The device will notify users with light signals when you need a turn.

Users have to connect the device with their own navigation app running on your smartphone. They offer the physical device for USD$104.99 (with mount) and the companion smartphone app for free.

Advantages

- Provide **cycling specific navigation** with **intermediate destinations**
- Support **external routes import**, such as GPX files
- Cross platform **route sharing** via iMessage, Email, Twitter, Facebook and Whatsapp

Disadvantages

- **Expensive** for the physical device with limited usability
- **Easy to miss** the turning light signals
- There are **limited information** provided on the display of the device
    - For example, map and route display would be useful
- Riders typically mounts cyclometer on their bike, having an **extra device** mounted would not be an elegant solution
    - Messy mounts, extra weights and air resistant are not preferred
- Using **online map and navigation** service
- Not support **round trip navigation**, which is important for cycling

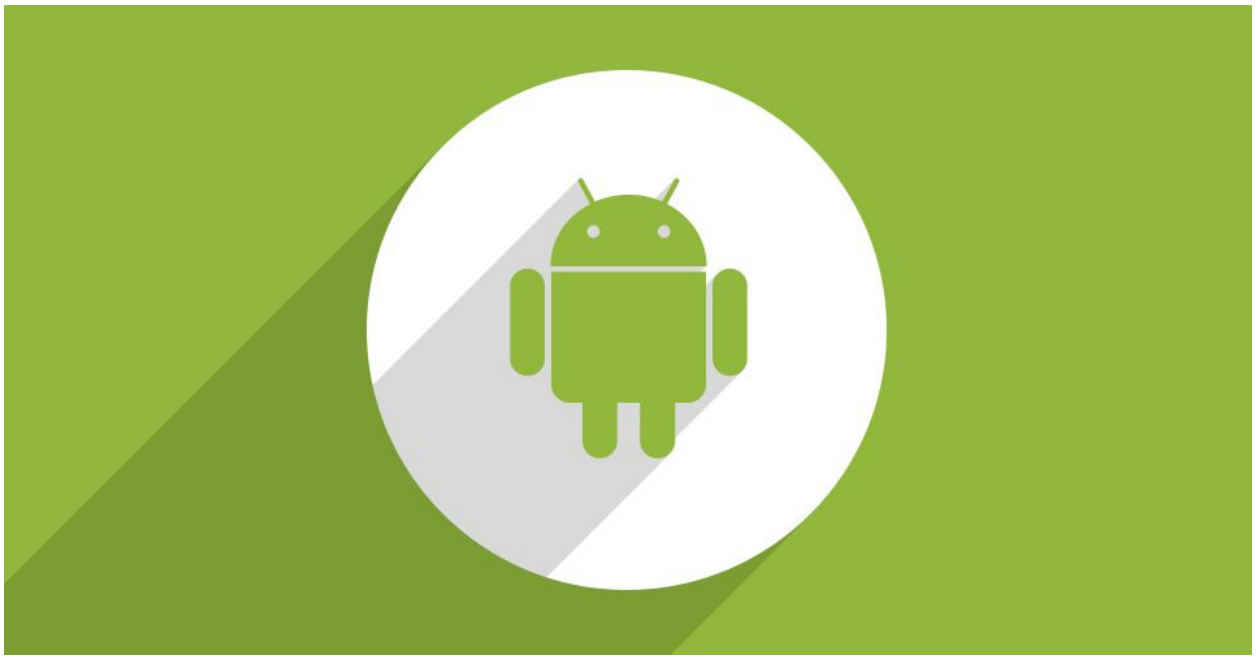
Objectives

A fully working **Android application** running on smartphone

- Provide **offline** map display
- **Construct route** with **round trip** and **intermediate destination**
- Provide **external routes import** via **GPX files**
- Provide **route analysis**, such as **distance** and **elevation data**
- Provide **turn-by-turn navigation**
- Provide **real-time information**, such as speed, distance traveled and distance remaining

An **Android Wear** companion application

- **Turn-by-turn navigation** display with **vibration** notification
- Provide **real-time information**, such as speed, distance traveled and distance remaining



Methodology

Software Development Method

- **Incremental model** will be applied
- **Working implements** will be delivered after each **milestones**

Platform and Technologies

- The application will be implemented in **Java** for recent versions of **Android**
- **OpenStreetMap** will be used for the offline map data

Milestones

Milestone	Incremental Accomplishments
Milestone 1	Smartphone application • Complete **user interface** • Able to **load** and **display** the offline **OpenStreetMap** data
Milestone 2	Smartphone application • Able to **import external routes** via GPX files Android wear companion application • Complete **user interface**
Milestone 3	Smartphone application • Able to **construct routes** from destinations • Able to perform **route analysis**
Milestone 4	Smartphone application • Complete **turn-by-turn navigation** Android wear companion application • **Turn-by-turn navigation** display
Milestone 5	Smartphone application • Provide **real-time information** Android wear companion application • Provide **real-time information**

Schedule

Date	Accomplishments
4 October 2015	Complete **inception phase** • Detailed project plan • Project web page
8 November 2015	Complete **milestone 1**
6 December 2015	Complete **milestone 2**
11-15 January 2016	First **presentation**
17 January 2016	Complete **milestone 3**
24 January 2016	Complete **elaboration phase** • Preliminary implementation
13 March 2016	Complete **milestone 4**
10 April 2016	Complete **milestone 5**
17 April 2016	Complete **construction phase** • Finalized tested implementation • Final report
18-22 April 2016	Final **presentation**
3 May 2016	Project **exhibition**
6 June 2016	Project **competition**