CORA Cork Open Research Archive
Cartlann Taighde Oscailte Chorcaí

Title	Cyclists' route planner mobileapp for Dublin city
Authors	Gavin, Maurice;Pakrashi, Vikram;Barton, John;O'Flynn, Brendan;Bouroche, Mélanie;Ghosh, Bidisha
Publication date	2012-10-01
Original Citation	Gavin, M., Pakrashi, V., Barton, J., O'Flynn, B., Bouroche, M. and Ghosh, B. (2012) 'Cyclists' route planner mobileapp for Dublin city', 40th European Transport Conference, Glasgow, 8-10 October. Available at: https://aetransport.org/past-etc-papers/conference-papers-2012?abstractId=3932&state=b (Accessed: 26 May 2022)
Type of publication	Conference item
Rights	© 2012, Association for European Transport and Contributors.
Download date	2026-11-02 02:25:23
Item downloaded from	https://hdl.handle.net/10468/13264

UCC
University College Cork, Ireland
Coláiste na hOllscoile Corcaigh

Cyclists' route planner mobileapp for Dublin city

Maurice Gavin
Trinity College Dublin

Vikram Pakrashi
University College Cork

John Barton
Tyndall National Institute, Cork

Brendan O'Flynn
Tyndall National Institute, Cork

Mélanie Bouroche
Trinity College Dublin

Bidisha Ghosh
Trinity College Dublin

ABSTRACT

This paper presents the development of a new mobile application, Rothaím, as Ireland's first adaptive route planner for cyclists. The mobile application is developed on an Android platform and is dependent on Global Positioning System (GPS) technology. The application is capable of tracking the behaviour, the motivation and the perceptions of cyclists in the Irish road network. Although Dublin is one of the upcoming cities in friendliness towards cycling, there has been very little data collected in this regard and the development of Rothaím is expected to remedy this deficiency to a significant extent. The impact of the application on urban mobility is immediate and it also creates a data repository to analyse the behaviour of cyclists in a mixed mode network. Consequently, Rothaím is expected to play a positive role in choosing cycling as a green and sustainable option, thereby offering a push towards a modal split to the advantage of urban cycling. Cycling is unique in terms of mode choice and the desired route may significantly vary from the least path. This application has the potential to quantify these varied options of route choice. The design of the application is discussed in detail. The development of the application, related practical issues and implementation are presented. The capability of the collected information to personalize data is demonstrated.

© Association for European Transport and Contributors 2012

INTRODUCTION

Dublin has recently started to establish itself as one of the friendliest cities for cycling, despite relatively limited existing infrastructure towards promotion of cycling. In 2008, it was estimated that traffic congestion costs Dublin businesses in excess of €2.5 billion each year in loss of productivity [1]. The related positive effects on health through cycling are expected to be very significant and far reaching, although a specific number is yet to be established quantitatively. The *Smarter Travel Plan 2009-2020 - the Framework Policy for Sustainable Transport* sets out the objective to have 10% of all journeys to work taken on bicycles by 2020 [2] in Dublin. Between 2005 and 2010 there was a 35% in the number of cyclists on the roads of Dublin [3]. Despite this encouraging modal shift towards cycling, urban transport planning involving bicycles is still significantly steered by information derived from local knowledge and common sense. To accommodate and integrate cycling into the transportation network in Ireland, it is important to quantitatively assess the journey behaviour of cyclists in detail. Such quantification not only addresses the route choice of the cyclists, but can also help shed light on aspects of safety that have been qualitatively identified. A large volume of cycling data needs to be amassed in this regard and it is believed that the development of a mobile application, helpful to both the cyclists and the researchers, is important for Dublin (and Ireland). The few existing cycling-specific route planners only return routes where the path is optimised for least distance or least time. In a road network, cyclists are the group exposed to the maximum amount of risk. Consequently, their optimal route can often be influenced by other factors like level of expertise, perceived or actual road risks, personal decisions, and weather conditions as opposed to least path or least time. The San Francisco County Transportation Authority recently funded the development of an application for Android and iPhone [4]. The goal was to identify the routes most frequently travelled by cyclists in San Francisco. In addition to routes, the application also captures route information and the experience level of the cyclist. TrailBlaze, from Aqualab, is another recently released Android application which tracks the route taken by a cyclist. This application is popular in Chicago and it encourags users to record their favourite routes so the city planners are aware of these popular routes [5]. IBM is running a project through Smarter Cities Technology Centre to investigate the impedances to people moving about the city. Factors from road width to the availability of public transport are being considered and model updating and calibration through the incorporation of sensors have been considered [6].

This paper presents a mobile phone application, Rothaím, which has the ability to collect a large volume of cycling route data and journey meta-data. By collecting this data with software this study signals a departure point from the traditional time intensive methods of gathering the same information, such as meeting with users to discuss their trips. This autonomous system, where users do all of the data collection, is achieved by creating a mobile application to facilitate the recording of data and a central server to store this data securely. A secondary goal is for users to use the mobile application not only for the purpose of research but rather as a fully fledged cycling application to support their cycling activities. Success of this goal will complement the first one as users will be self-motivated to gather more data through their continued use of this software. This application has been evaluated on the transportation network of Dublin, but has the capability to be used anywhere. It is expected that the data collection and personalized route selection will lead to a better understanding to how cyclists may be fully and safely integrated into the transport network.

© Association for European Transport and Contributors 2012

DESIGN OF ROTHAÍM

Rothaím consists of two distinct packages, the application and the server. Figure 1 presents a flowchart of the top level design.

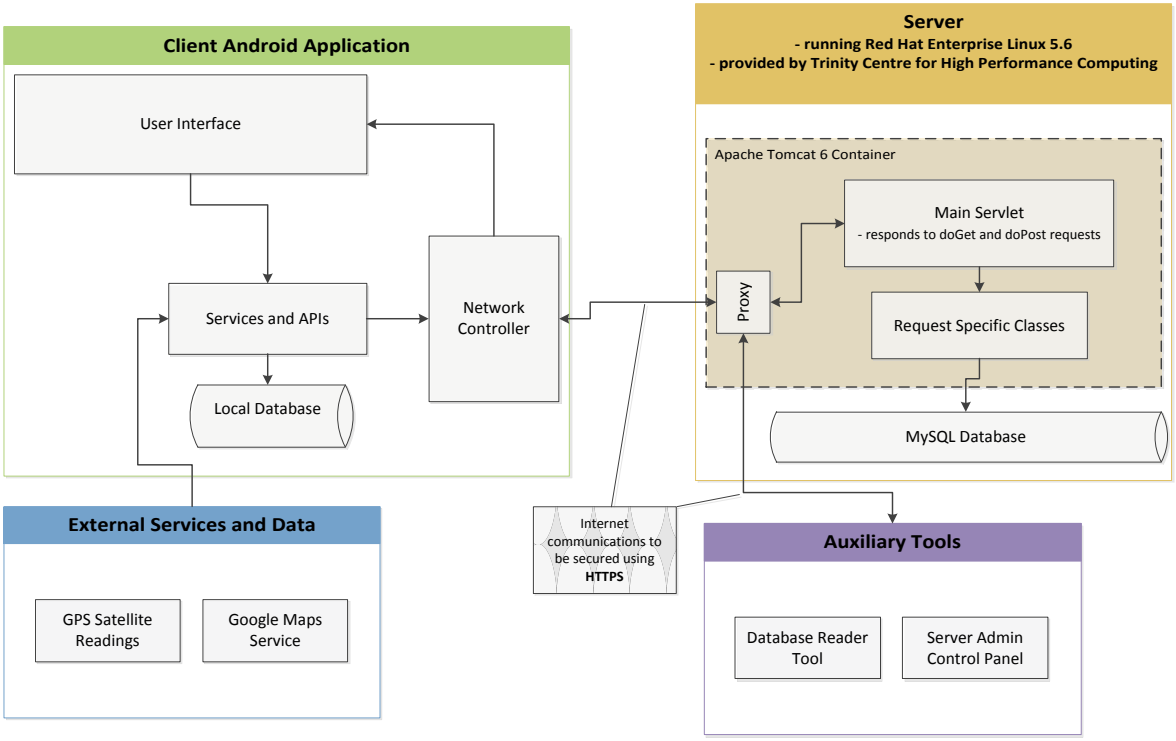


FIGURE 1 System Design

The Android application is operated by the cyclists. Before departing on a cycle the user tags some meta-data about their route preferences and motivation (e.g trip reason = recreation). At the other end of the journey the route travelled and related data is uploaded to a secure server. The data on the server is accessible to the user from the application on the phone and anonymously to the researcher via a variety of auxiliary tools. Each component has its own specific responsibilities and when made to interact with each other together they form the system which fulfils the objectives detailed in the previous section.

Application

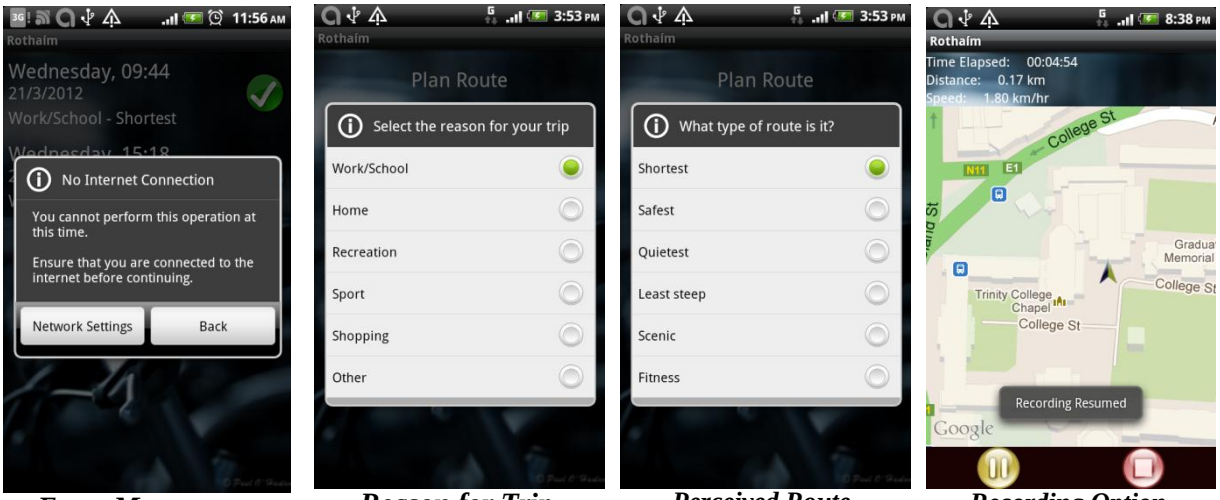
Rothaím is an Android application. A factor for this choice of platform was the bias of local cyclists. After consulting a number of cyclists in Dublin it appeared that phones running iOS or Android outnumbered other devices. The difference between the number of people using Android and iPhone was not sufficiently large enough to favour one over the other. When development costs were taken into account, developing, implementing and validating the application was observed to be cheaper on phones running Android. The detailed design of the application is provided in this subsection.

© Association for European Transport and Contributors 2012

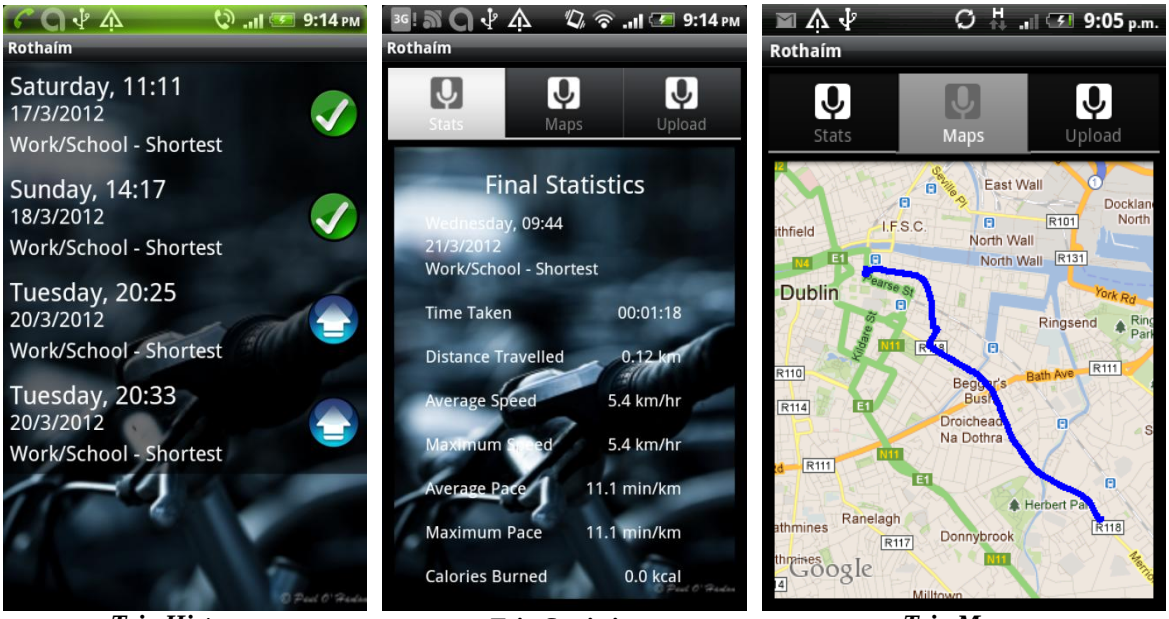
The different aspects of the android application design are discussed in the following sub sections:

A) User Interface

The User Interface (UI) is the attractive front end of the application seen by the user which handles all human interactions.



Error Message ***Reason for Trip*** ***Perceived Route*** ***Recording Option***



Trip History ***Trip Statistics*** ***Trip Map***

FIGURE 2 Screenshots of the developed Rothaím application

Through it, a user can initiate recording of trips, upload information to the server and even publish completed journeys to their favourite social network. A splash screen is shown on launch; this transitions to a screen where the user can configure the recording of a journey by

© Association for European Transport and Contributors 2012

indicating the reason for their trip and their perception of their route. This brings them to the recording screen with a map centred on the user's current location, dynamic statistics about the trip and a panel of control buttons. The application can also be used for navigation purposes since the current location is overlaid on a local map allowing pan and zoom. The user can pause recording their journey at any time. When a user is finished recording their trip they are shown the final statistics for their trip, including time taken, distance travelled, average speed, maximum speed, average pace, maximum pace and an estimation of the calories burned. Figure 2 provides some screenshots of the implemented Rothaím application relating to the discussions detailed in this section.

B) Service and APIs

Rothaím depends on the availability of GPS satellite readings and a connection to the Google Maps Service to record and display route data. Recorded data is stored in a SQLite database on the device. Since users tend to use many applications at once on their devices it is necessary that the app can continue to work in the background. If a user launches Rothaím while recording is already in progress the application must behave the same as if it had never been closed. The user can change the recording state by using the controls provided on the screen to pause, resume and stop recording.

C) Networking

Networking is the communication of the application with external services via the internet. All networking are volatile in nature so a common programming feature called *threading* is used to let the time consuming networking to run independent of the interface. In some cases, such as uploading a trip, a progress bar is displayed while the operation completes. By default all outgoing connections are HTTPS, this means that data sent across the network is encrypted. HTTPS is safer but the encryption overhead also makes it slower than the unencrypted alternative, HTTP. Therefore the user can disable it if they wish through the 'Settings' screen. Depending on the result of the network request, a message indicating success or the type of error is returned.

Server

The server architecture can be explained as three components. There is a proxy for authentication. The 'Main Servlet' executes all validated requests to the server and manages the responses. Behind the server there exists a MySQL database with three tables. This database is used for the safe storage and retrieval of user and trip data. An Apache Tomcat 6 container enables the proxy and main servlets, written in Java, to run on the server. A servlet is simply a Java class which runs on a server. When a request reaches the server the proxy servlet attempts to verify its validity. Authenicated requests are forward the request to the *MainServlet*, which deals with all non-authentication related requests, such as uploading, downloading, etc. The aforementioned have access to the backend database.

A) Proxy

The Proxy (*AuthenticationServlet*) maintains a static list of valid session keys. Session keys are given to authorised users to allow them to perform privileged tasks for a short window of time. Each session key has an ID, an expiration time and the name of the user to which it was assigned. All keys are valid for a 15 minute period. Expired session keys are routinely cleaned

© Association for European Transport and Contributors 2012

by a scheduled thread. When a user wishes to perform any operation anywhere on the server they must first provide a valid session key otherwise their request will be immediately dropped and an error message returned in the response.

The proxy also handles signup and the login requests. If a user attempts to login it is expected that they do not have a valid session key. The username and password are checked against the database and if a match is found then it a key is returned. Once the servlet validates a session key it checks the request's *mode*. This indicates the type of request. The proxy's supported modes are:

- Ping: a mode is accompanied by a session key to check whether the session key provided is valid.
- Login: a feature allowing a user to get a session key by providing a valid username and password.
- Signup: requests with this mode are accompanied by the details necessary to create a new user. The servlet creates a writable connection to the backend database and inserts a new row in the user table.

B) Requests

The *MainServlet* deals with all requests forwarded by the proxy and no validation is required with a session key. There are two types of request that are handled by *MainServlet*: 'doGet' and 'doPost'.

GET comprises of:

- List: this request returns a list of trips that the logged in user has permission to access. Most users only have permissions to read their own trips. Privileged users, for example those who are accessing the data for research purposes, are able to access the trips of other users.
- Download: the download request contains the *databaseID* of the trip to be downloaded. The server will verify that the user requesting the download has permission to read it. If it does then it retrieves the trip from the database and writes it to the response.

POST comprises of:

- Create: this request is sent to upload a trip to the server's database. The request header contains the basic trip information like trip reason, perceived route type, the start time and so on. The servlet creates a new row in the database and returns the primary key of the row.
- Update: this request is sent to upload location observations. Each request must also contain the primary key of the parent trip so that rows in the observation table can be associated with the correct trip. These observations are usually sent in chunks so it is not uncommon for an entire upload transaction to have many update requests. This means that if a transmission does not succeed, there is less data to retransmit on the network.
- Finish: a finish request is sent when there are no more observations to be uploaded. The servlet can then close its connection to the database and sends an acknowledgement to indicate a successful upload.

C) Databases

© Association for European Transport and Contributors 2012

At the backend of the server there is a MySQL database. The database can be used to store information in a structured manner. Aside from storage it also provides security, indexing for faster searching and crash recovery support. A user table stores the account information of those users who have registered to use Rothaím. A trip table and an observation table store the trip and route data uploaded by cyclists respectively. The user table is written to exclusively by the proxy when new users are created. Similarly the trip table and observation table are only ever written to by the *MainServlet*.

The tables contain the following information: The user table contains the username and password, email, date of birth, gender, weight and height of the user. The trip table contains the trip identification number, the username of the trip owner, the reason for journey, the route type, the time of day, the start and end times of the trip, the time of upload and the distance travelled. The observation table contains the observation identifier, trip identifier, the latitude and longitude of the device (including an approximation of the accuracy of these readings), the time at which observation was made, the speed and the altitude.

The server is designed to allow a developer to securely and quickly change server properties. The main issues around the aspect of scalability of the application are related to supporting distributed services, managing large hit rates, quality of service concerns, architectural issues and performance of back-end services. It is expected that the Rothaím server will be scaled up eventually when the resources of the current server are no longer sufficient to match the load.

A number of data security measures have been taken to ensure protection of user data. The server proxy forces potential users to authenticate themselves with username and password login credentials. The expiry of session keys ensures that even if someone intercepts the session key that they will only be able to use it for a short time. The server is configured to accept secure HTTPS connections.

In addition to the encryption benefits provided by the Presentation Layer of the OSI Model, HTTPS also facilitates server authentication. When a client receives a response from a server it contains a copy of that server's public certificate. Since the server certificate is not held by any certificate authority it is necessary for the client application to store a local copy of the server certificate to authenticate against. This setup provides reasonable protection from 'man-in-the-middle' style attacks.

The server database has three available user accounts, each with varying levels of privileges. The lowest level of access is Read-Only, the next up is Read-Write and finally one has Root privileges. When the server uses interacts with the database it uses the account with the minimum privileges required to complete the operation. The Root account is not used anywhere in the software and no record of the credentials are stored on the server. SQL injection attack involves the insertion a SQL query to a request, returning sensitive user data from the database or can modify the database. Rothaím uses statements for all queries made to the database. These parameterised queries prevent SQL injection.

Software for accessing the server was developed to aid researchers in accessing the data collected by Rothaím. The "Database Reader Tool" can be used to list trips on the server and to download and export a file describing the trip in a number of useful formats. The software logs the researcher in as a privileged user and synchronises a local list of trips with the available trips it can access on the server. It shows a summary of each trip, including the time it was recorded,

© Association for European Transport and Contributors 2012

the reason for the trip and the route type. It was designed to allow to be fast and transparent to the structured trip data stored in the server database.

DEVELOPMENT

The target version for Rothaím is Android's Platform 2.1 (Eclair) but is backwards compatible as far as Platform 1.6 (Donut). This means that Rothaím can run on 99.6% of active Android devices. The application was developed in Java with the Eclipse IDE and the Android Software Development Kit (Android SDK).

User interfaces in Android are created by defining the desired layout in Extensible Markup Language (XML). Alternatively, if required, they can be programmatically generated. Rothaím combines both of these approaches. Simple layouts are created using XML exclusively. However, more complex screens, such as screens with maps, lists and tabs need to have their contents set programmatically. All static resources, layout margins, static text, font sizes and so on are also all stored in XML files. Each new screen in Rothaím extends a base class that adds some static variables and utility functions used by multiple interfaces. For a user to interact with the interface the screen needs a large number of listeners on buttons, lists, icons, events etc. To add a listener to buttons and views defined in the XML layout the some initialisation functions are overridden and extended.

Technical Challenges

A number of programming challenges existed initially, of which some are outlined. When creating lists and tabbed frames, a difficulty was that these content containers were completely different to the previously implemented views. Once a list element is no longer visible it is unbound from the view and the row item is efficiently allocated for new contents. This led to a problem where the row's listeners didn't correspond to the contents of the row.

Additionally, with Android phones being made by many companies, there exists a huge variation in the resolutions, aspect ratios and screen dimensions in target devices. This makes it difficult to design an interface that scales well to each device. Multiple XML layouts were created for the same screen to optimise the usability and presentation of the application on phones with varying screen sizes and pixel densities. Also, images used in the application had to be made available in three sizes. This increased the size of the application, but was necessary to make the application scale appropriate to a diversity of screens.

When users perform operations such as uploading large trips they sometimes thought that the application had frozen or crashed when the uploading ring did not disappear in a timely fashion. The progress bar was implemented to combat this perception. The application calculates how much data is to be transmitted and using the buffer size it calculates the percentage of the total transmission each successful update request fulfils. This results in an accurate indication of how long it takes for the operation to complete.

Rothaím is reliant on GPS location updates and Google's maps service. Nearly all Android devices come equipped with GPS receivers. To determine the location these need to lock on to a minimum of three satellites in a short timeframe. The device can then calculate its position using triangulation [7]. Locations returned by coarse techniques using cell towers and WiFi are only accurate to between 50m and 100m. Assisted GPS (A-GPS) technology can improve the "Time To First Fix" (TTFF) by using an available network connection. Results vary depending on when the device's GPS was last used and hence the integrity of its cached satellite

© Association for European Transport and Contributors 2012

information. An issue with data collection was the length of time that it takes to receive the first location fix. From testing carried out during the project it was found that the TTFFs of Assisted GPS (A-GPS) devices were between 5-30 seconds. This time increased if the receiving device moved while awaiting its first fix. The second issue is related to the fact that cycling of interest predominantly takes place in an urban area and waves can be received after reflecting off a building as opposed to coming direct from a satellite, introducing inaccuracies.

For users with an available network connection, a small packet is downloaded for Rothaím containing the latest positions of the satellites and this significantly decreased the TTFF. To increase accuracy the sampling rate was increased from 12 locations per minute to 120 locations per minute. The effect of an increased sampling rate on battery life was considered and it was established that a user could comfortably complete two trips a day with the maximum sampling rate chosen and still have sufficient charge remaining for normal use. This was also confirmed through volunteer feedback. Increasing the frequency of location updates does not improve the accuracy of the updates. Rather the additional readings can be aggregated to generate a better estimation of the true location.

The issue with crashes occurring on long trips was solved by periodically flushing memory when the buffer becomes too full. Google API's Google Maps add-on was chosen to provide the maps for the application. They both dynamically download their maps upon request. As a result any updates to the maps are automatically available on the phone application without the need for a software update.

The map can pan and zoom and each time a location is received, the map is centred on the user location. A debug certificate is automatically created by the Eclipse IDE and a key generated from this certificate can be used while debugging. Java provides the `keytool` command to manage keys and certificates. After creating a certificate for the release build of the application, its MD5 fingerprint is used to generate a new API key. This key then needs to replace the debug key for maps to be available on the published application. Once observations exceeded a few hundred, the map interface began to slow down and framerate dropped as the user panned and zoomed on the map. For trips with thousands of observation points it could take tens of seconds for the route to be drawn on-screen and then if the user panned or zoomed they would have to be redrawn all over again.

A solution was found by determining the maximum and minimum values for latitude and longitude of a trip. The map is zoomed out so that it can fit the entire trip and the data resolution is scaled to optimise performance. At this level of abstraction missing data points are not noticeable. This solution scales well to trips of all lengths.

EVALUATION

Rothaím was evaluated over a three week testing period. The application was distributed to a group of volunteer cyclists, with their full consent. The Rothaím application does not encourage users to interact with the application while they are cycling and dissuades the users from doing so through its minimalistic features during the course of a trip. Cyclists with no available internet connection can upload their trips at WiFi zones. Some of the challenges in development, as discussed in the previous section, were identified during this time and the application was accordingly updated to address them.

Some initial results during this evaluation period are presented to illustrate the data analysis possibilities made possible by this application for an upcoming cycling friendly city like

© Association for European Transport and Contributors 2012

Dublin. Figure 3 provides a breakdown of the motivation behind cycle trips, as described by the cyclists themselves. The bulk of the trips undertaken were for either going to work or coming home. It was expected that deviation between the number of trips to work or school and the journeys home would not be large. The trend for users is that if they remember to record their commute in the morning they are more likely to log a trip on the journey home also. The recreational trips were all logged at the weekend and tended to be round trips, departing from a location and returning to the same location. The shopping trips were also round trips. For the shopping trips, the application was not paused upon arriving at the shop and so there is a window of time where no GPS locations are observed. Presumably, this is because the application was used indoors at these times. For cycling data with no meta-data, no journey reasons or route perceptions, this deduction could not be reasonably supported. However, knowing the cyclists intention for the journey we can better understand the anomalies that arise in their trips.

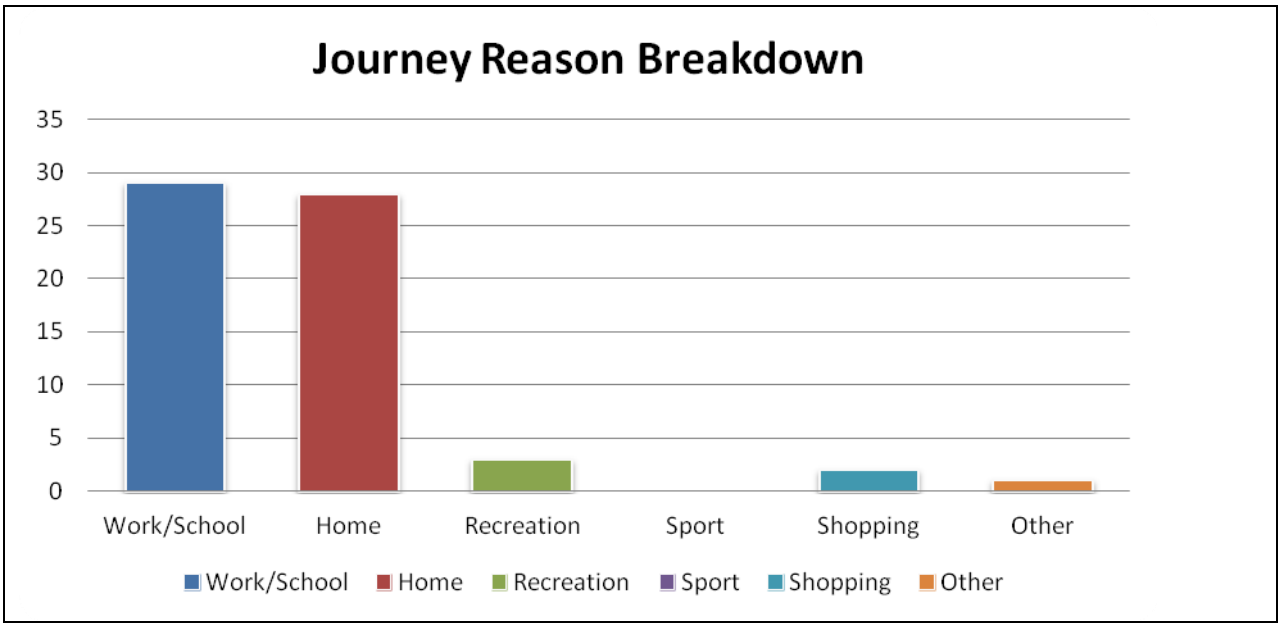


FIGURE 3 Breakdown of journey reason as reported by volunteer cyclists.

There are interesting insights to be drawn from this data set about cyclist perceptions of routes, as presented in Figure 4. 83% of trips uploaded to the server had cited shortest as being their perception of a route. This suggests that short routes are generally to be desired by cyclists. However, if the alternative route types are considered and correlated with the journey reasons, more interesting results are revealed. Although there are few scenic trips, it can be seen that none of them correspond to work journeys. Two of them are for recreational journeys and the remainder are for trips home. Similarly, the safest and quietest routes were taken during trips home and to the shops.

Figure 5 presents the distribution of time at which each journey was begun and the number of related trips relative to the time of day. Most trips take place between the hours of 8am and 10am in the morning and again between 4pm and 6pm in the evening.

© Association for European Transport and Contributors 2012

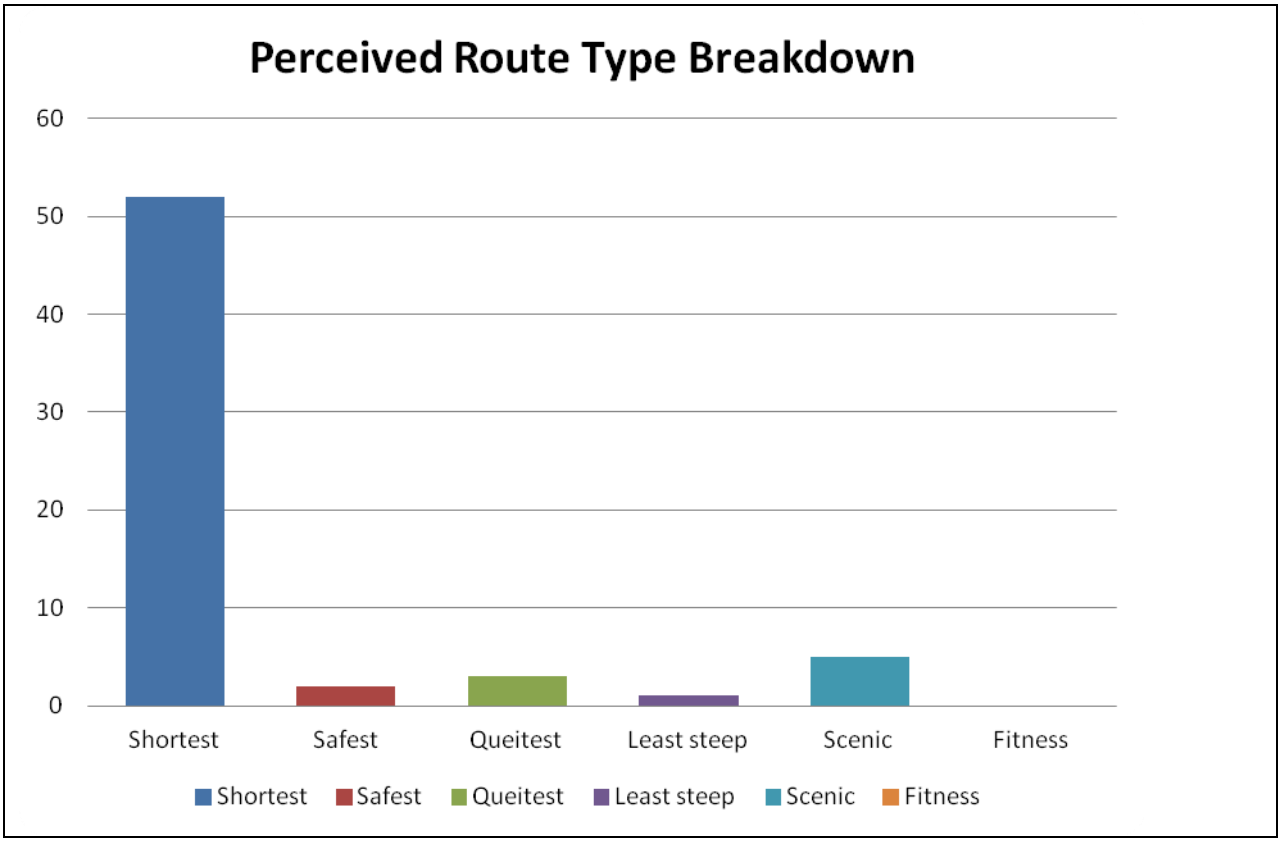


FIGURE 4: Perception of routes by volunteer cyclists.

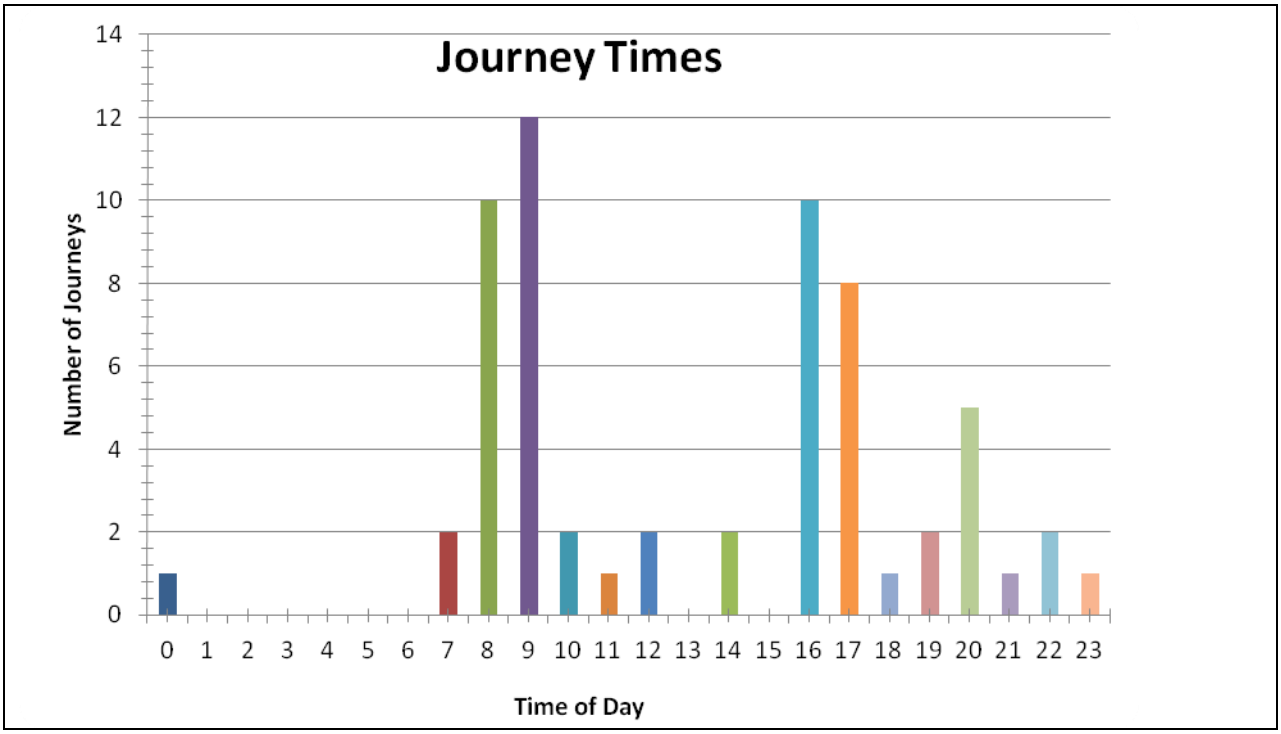


FIGURE 5 Distributions of Trip Times Over The Entire Day.

© Association for European Transport and Contributors 2012

This is unsurprising as these are the rush hour periods in Dublin city, when most commuters are moving around. The demographic of volunteers used for the data collection period was biased towards researchers and students due to the process of selection. This may account for the moderate dispersion in evening journeys as researchers and students are often quite late leaving for home.

Filtering Dirty Data

One of the constraints of this approach of gathering GPS route information is that the handheld mobile devices do not always record high quality data. To make the data more useful it can be filtered in an attempt to remove inaccurate readings. A Kalman Filter was used to process the data. Kalman uses a series of location observations and other readings, all of which contain some inaccuracy or noise, to predict the future state of the system. The prediction is then combined with the actual reading to create an estimate of the current position that is better than either the prediction or the actual reading.

The filter employs two distinct stages, a prediction stage and an update stage. The state of the system is modelled using two variables, position (p) and velocity (v).

$$x = \begin{bmatrix} p \\ v \end{bmatrix} \tag{1}$$

In the prediction the first step is to make a state prediction, $\bar{x}_t$. Since the data is coming from a device which is governed by the laws of physics we can make a prediction based on the equations of motion.

$$\bar{x}_t = Ax_{t-1} + B\mu_t + \varepsilon_x \tag{2}$$

$$\bar{x}_t = \begin{bmatrix} p_t \\ v_t \end{bmatrix} + \varepsilon_x = \begin{bmatrix} p_{t-1} + v_{t-1} + \frac{1}{2}a_t T^2 \\ v_{t-1} + a_t T \end{bmatrix} + \varepsilon_x \tag{3}$$

where, μ_t is a control input at timestep t
ϵ_x is the error at timestep t
$p_t$is position at timestep t
v_t is velocity at timestep t
a_t is acceleration at timestep t
T is the time since last reading

Similarly a prediction of the measurement, $\bar{Z}_t$, is made. This is used later.

$$\bar{Z}_t = C\bar{x}_t + \varepsilon_z \tag{4}$$

where ϵ_z is the measurement error at timestep t

The state prediction error, E_x, is expressed as a covariance matrix.

$$E_x = \begin{bmatrix} {\sigma_p}^2 & \sigma_p\sigma_v \\ \sigma_v\sigma_p & {\sigma_v}^2 \end{bmatrix} \tag{5}$$

where σ_p is the standard deviation of position
σ_v is the standard deviation of velocity

© Association for European Transport and Contributors 2012

The measurement error, E_z, is also expressed as a covariance matrix. This value changes dynamically and derives its value from the accuracy estimation which comes with each GPS reading.

$$E_z = {\sigma_z}^2 \tag{6}$$

where σ_z is the standard deviation of the measurement

A value for the predicted covariance matrix is computed using (7).

$$\bar{E}_t = AE_{t-1}A^T + E_x \tag{7}$$

where $\bar{E}_t$ is the predicted covariance
E_{t-1} is the previous covariance

The next stage is to update the model using the actual reading from the satellite.

$$K_t = \bar{E}_t C^T (C\bar{E}_t C^T + E_z)^{-1} \tag{8}$$

where K_t is the Kalman Gain
E_{t-1} is the previous covariance

$$x_t = \bar{x}_t + K_t(Z_t - \bar{Z}_t) \tag{9}$$

where x_t is the final state prediction
Z_t is the true measurement

Finally estimate the final covariance, E_t.

$$E_t = (I - K_t C)\bar{E}_t \tag{10}$$

where I is the identity matrix

The filter constructed the next step is to use it. The recorded latitudes and longitudes are translated into a local coordinate system where the difference between observations can be expressed in meters. The data is then passed through the filter along with corresponding velocities, computed accelerations and accuracy estimates from the GPS satellites themselves. The result is translated again to express it in terms of latitude and longitude.

A noticeable improvement is apparent as can be seen in the comparison shown in Figure 6 below. To improve the accuracy more information would be required. An auxiliary source of data might be to use an inertial navigation system, such as a mobile device's accelerometer, to record acceleration, bearing and so on. The filter could be altered to account for this new information and thus provide a better approximation.

© Association for European Transport and Contributors 2012



FIGURE 6 Comparison of Kalman Filtered Route

CONCLUSIONS

This paper presents a new mobile application, Rothaím, geared towards the promotion of personalized travel for cyclists in Dublin. The design of this Android platform based application is discussed in detail. The issues related to the development and implementation of the application is then considered. Specific challenges related to the actual development and the solutions to the problems encountered are presented. The application has been tested on volunteer cyclists. This paper also presents the capability of the application to collect and analyze the personalized travel patterns on cyclists. In the present study, a pilot data collection exercise has been presented. With market availability of Rothaím, it is expected a much bigger trip database will be available to the researchers for future analyses purposes. These analyses will facilitate new bicycle infrastructure and route design projects and will also facilitate personalized trip option provision for individual cyclists. This will be a unique tool which will not only serve the researchers but also serve the users to improve their travel efficiency.

Future research in relation to this tool will support planning or integrated (bicycle, cars and public transport) multi-modal transport networks.

REFERENCES

1. Dublin Regional Authority, “2020 Vision – Sustainable Travel and Transport,” Dublin, 2008.
2. Department of Transport, “A Sustainable Transport Future. A New Transport Policy for Ireland 2009-2020.,” Dublin, 2009.
3. Gormley N., “Report of the Chairperson and Members of the Transport and Traffic Strategic Policy Committee”, [Online]. http://www.dublincity.ie/RoadsandTraffic/Documents/7%20Canal%20Cordon%20Counts%202010.pdf [Accessed 31 07 2012].

© Association for European Transport and Contributors 2012

4. Zorn L., Sall. E., Bomberg. M., “Completing the Cycle: Incorporating Cycletracks into SF-CHAMP,” 4th Transportation Research Board Conference on Innovations in Travel Modeling (ITM), 2012.
5. Aqualabs, 26 08 2011. [Online]. Available: http://cycle.aqualab.cs.northwestern.edu/. [Accessed 04 04 2012].
6. IBM Corporation, “Smarter Cities,” [Online]. Available: http://www.ibm.com/smarterplanet/ie/en/sustainable_cities/visions/index.html. [Accessed 04 04 2012].
7. abeckwith@colonial.net, “Global Positioning System - Advanced Topics in Mathematics,” 01 08 2008. [Online]. Available: http://mail.colonial.net/~abeckwith/FOV1-0004FCD4/006B0D3A-70E903AC-036322B9. [Accessed 30 03 2012].

© Association for European Transport and Contributors 2012