Chapter 68
Android and iOS Hybrid Applications for Surabaya Public Transport Information

Djoni Haryadi Setiabudi and Lady Joanne Tjahyana

Abstract This study is conducted to address the lack of route information of public transportation in Surabaya by creating an online guide that can be accessed by passengers to get complete information on maps and travel routes for public transportation. This guide is made interactive, simple, accessible and appropriate for transportation that is adapted to the conditions in the city of Surabaya. This research will develop an Android and iOS applications that can be used on smartphones and tablets using Android and iOS operating systems. Maps and routes are obtained from the Department of Transportation of Surabaya. Maps and route are developed using OpenStreetMap, Ajax, Javascript, XML, OpenLayer, PostgreSQL, and PostGIS. The hybrid application is compiled using PhoneGap. Passengers simply point to the destination of their journey, such as the name of the street, landmarks and public places. The system will automatically choose the alternative line of *bemo* they should choose, including the routes to reach the destination. The information includes the connecting line of a public minibus (called *bemo* in Indonesian) if the route needs to be connected by more than one *bemo* line. The information also includes the fare to be paid. From the test results, both the Android and iOS applications can adapt to a wide range of smartphones with a variety of screen sizes, from 3.5 to 5 in. smartphones and 7 in. tablets.

Keywords Routes · Maps · Public transportation · *Bemo* · Openstreetmaps · Android · iOS

D.H. Setiabudi (✉)
Informatics Department, Petra Christian University, Surabaya, Indonesia
e-mail: djonihs@petra.ac.id

L.J. Tjahyana
Communication Science, Petra Christian University, Surabaya, Indonesia

68.1 Introduction

Currently one of the mass public transportations in the city of Surabaya is the type of small and medium 1000 cc minibus commonly called *bemo* with a maximum carrying capacity of 10 persons [1]. Watkins et al. [2] in the studies of open source development of mobile transit traveler OneBusAway information system for King County Metro (KCM) in Seattle suggest that the provision of good information system for public transport passengers will increase passengers' satisfaction and increase the interest of the public transport. It would be useful to encourage people to switch from the use of private transportation to the public transportation. Setiabudi and Tjahyana [3] did a research on Surabaya public transport by using PHP language programming and MySql database, accessed by a web browser. This application has applied website responsive technology, but each time a user will use this application s/he must be connected to the internet. Yulianto et al. [4] did a research on Jakarta public transport developed with web-based framework code igniter. However, the route map displayed was only for private cars and for public transport was only ready for one route on the map.

Furthermore, based on the observation of the initial research, there are some Android applications of Trans Jakarta Busway public transportation in Jakarta. Among them is Komutta that has the highest rating 4.6 out of 1,263 voters and has been downloaded more than 50,000 times in the Google Play Store. However, all the applications that have been created can only be running on one operating system. For instance, the application that can be used on Android cannot be used in iOS, Blackberry and Windows Mobile. Conversely the one that runs on Blackberry cannot be used in other operating systems.

As a result, not all smartphone users can take advantage of software applications that have been created. This is because all of the applications initially were developed using a native application and not with the hybrid application [5]. As a result, if an application is already developed as an Android native application, in order to make it available for iOS native application the developer must re-develop the application from the beginning, due to different programming languages that are used to develop native applications for different operating systems. If initially the applications were developed using a hybrid technology to produce cross-platform application, for example for Android, then only very minor and simple changes needed to be done when it would be developed for iOS. The idea of implementing hybrid application could be seen in the diagram of Fig. 68.1.

The head of IDC Operations of Indonesia, Sudev Bangah argued that many smartphones with varying price would be affordable for smartphone users who were on the middle segment of the market [6].

Based on the fact that market share is issued by IDC Indonesia, to increase the satisfaction of *bemo* passengers in Surabaya, this research will develop applications
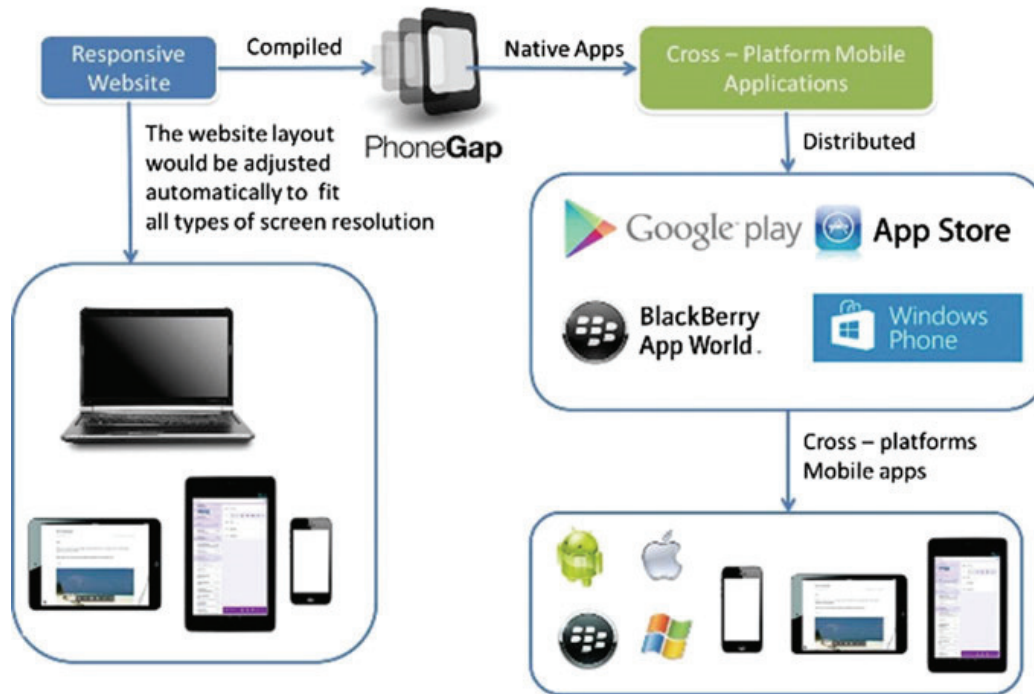
Fig. 68.1 Block diagram of the system

that can be used by smartphones of Androidand iOS operating systems for trip planning using bemo in Surabaya using Open Street Maps. Those applications will help the passengers plan their trips based on routes, timetables, and costs.

As shown in Fig. 68.1, the time can be shortened and costs can be saved because after developing a web-based application in the form of responsive website, it becomesa native applications using PhoneGap that can then be distributed through the application store for each mobile operating system.

The applications will provide information about route guide and timetable of public transportation in Surabaya. The information will be equipped with a search feature and a trip planning using public transportation simply by typing the place of origin and destination specified by the passengers. Both the place of origin and destination can either be a street name or a landmark name and public place such as *bemo* stations, parks, rivers, shops, schools, and others.

The planned features will use a multimodal transport network that takes into account some factors such as multimodal routes, timetables, and costs to provide recommendations for the optimized route. This feature will guide the passengers to the nearest public transportation station, by giving the instruction on which public transportation they should take, to arrive at the nearest public transportation to the destination. This feature will also be equipped with an estimation of the fare for the service.
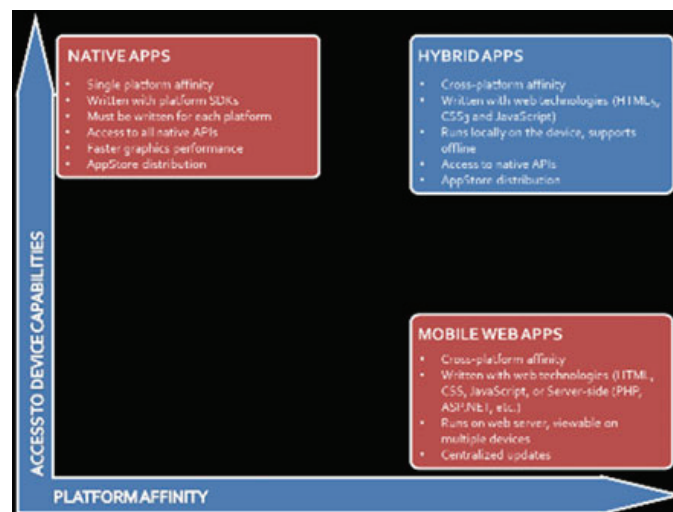
68.2 Hybrid Applications

There are several types of mobile application, which are Native App, Web App and Cross Platform Mobile App or Hybrid App. According to Stark [7], Native application is application installed on the phone, such as Android and iOS-based Smartphones. Native applications have access to the Smartphone hardware features such as camera, speaker, etc. developers have to use different kinds of programming language to develop native applications for different platforms. To develop Android native applications, Java programming language is needed. Objective—C is used to develop iOS native applications. Native applications are available on the official application markets such as Google PlayStore for Android and iTunes App Store for iOS.

On the contrary to the Native application, Web applications are not installed on the phone and they are not available on the official application markets such as Google PlayStore and iTunes App Store. However, they are easily accessed with the Smartphone's web browsers and developers are only required to use HTML, CSS, and JavaSript to build Web applications. The downside for Web applications is that it cannot access certain hardware features on the smartphone [7].

Hybrid application or Cross Platform mobile application is considered to be the right solution for developer to build iOS and Android applications without mastering many programming languages and downloading any SDKs (software development kits) for each platform, but it can put the application on Google PlayStore and iTunes App Store [8]. To build a hybrid application or a cross platform mobile application both for iOS and Android, developers only need to use HTML, CSS and JavaScript to develop a web application. Next, with PhoneGap technology, the web application will be packaged into a native application for iOS, Android and other mobile platforms [7].

Fig. 68.2 Three types of applications

A graph that highlights the differences in native, hybrid and mobile web applications can be seen in Fig. 68.2. Native applications are built for a specific platform with the platform SDK, tools and languages, typically provided by the platform vendor (e.g. xCode/Objective-C for iOS, Java for Android, Visual Studio/C# for Windows Phone).

Mobile Web applications are server-side applications, built with any server-side technology (PHP, Node.js, ASP.NET) that render HTML that has been styled so that it renders well on a device form factor. Hybrid applications, like native applications, run on the device, and are written with web technologies (HTML5, CSS and JavaScript). Hybrid applications run inside a native container, and leverage the device's browser engine to render the HTML and process the JavaScript locally. A web-to-native abstraction layer enables access to device capabilities that are not accessible in Mobile Web applications, such as the camera and local storage.

68.3   Research Methodology

68.3.1   Fishbone Diagram

The method used in this research can be seen in Fig. 68.3. The first step includes a survey of the timetables of public transportation, the fare of travel, and the location of public places.
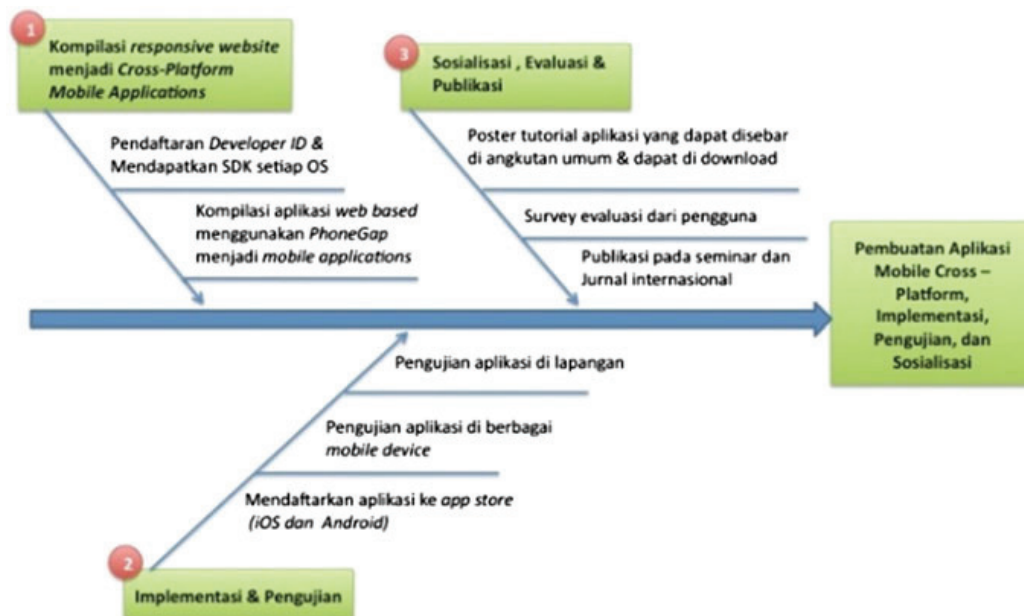

Fig. 68.3 Fishbone diagram of research methodology

The first step is to compile the responsive website that has been made to be cross-platform mobile application using Phonegap. At this stage the process will begin with registration of Developer ID in the application store for each operating system, until a developer SDK for each platform is secured. After this, the compilation of web-based applications with the SDK of each platform using phonegap, resulting in cross-platform mobile applications needs to be done. The outcomes are cross-platform mobile applications that can be used for each operating system (iOS and Android).

The second step will be the implementation and testing of applications with the following stages: Registering and uploading applications on each application store for each operating system (iOS and Android), namely AppStore for iOS and Google Play Store for Android. Testing the applications on a variety of mobile devices for each operating system with a different screen resolution size. Testing the use of applications in the real case and doing the journey planner on public transportation in Surabaya with some case studies of travel route need to be done.

The last stage or the third step will be disseminating and evaluating the applications with the following phases: Create tutorial posters about the application that will be shown at the terminal and all public transportation. Upload the tutorial posters at the Department of Transportation website and official Facebook page of the Department of Transportation. Finally evaluate the application by providing a place for criticisms and suggestions the make the application better.

68.3.2 System Design

The design of the client system is shown in the flowchart in Fig. 68.4. Firstly, the users must choose what they want to do, whether they want to look all routes or directly get to the direction. If they choose to see all routes, all lists of bemo lines nearby their current position will be shown.

Furthermore, they have to pick one from the list to see the route on the map. However, if they want to get directly to their destination, they must choose one point of interest available in the server database. Then, they have to choose how the application detects their current location using GPS or manually clicking on the map. Nevertheless, the method to detect the user's location still depends on whether the GPS is available on their device or not. If the application can detect the current location of the user, then it will show the routes to go to their destination using the closest line available near their current location.

The developing process is started by preparing a source code in the form of HTML, Javascript, and CSS on the Client, while on the Server there are database and PHP script. Next, it is converted into a hybrid application by using Phonegap. For development on the Android environment, softwares Node.JS,
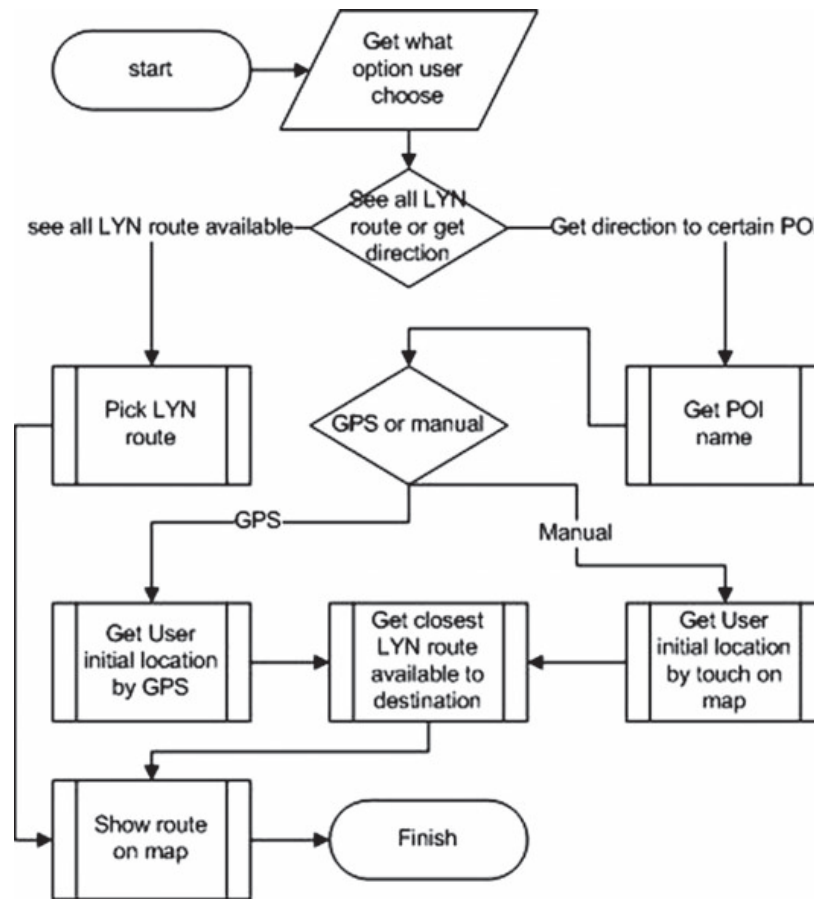
Fig. 68.4 Flowchart of the client system

Android Studio, Java Runtime Environment, JDK (Java Development Kit), and Android SDK (Software Development Kit) are needed. For development on iOS environment, softwares Node.JS, Xcode dan iOS Simulator are needed.

68.4   Experimental Results

The testing of the applications was done using two devices, namely 3.5 in. smartphone and 7 in. tablet, respectively for Android and iOS. The first time the applications is initialized, it willdisplay the screen like in Fig. 68.5. There are two options to choose, namely 'Search bemo routes to destination' and 'See all existing routes of bemo'.

If the user selects the option 'See all the existing routes of *bemo*', then the screen shows as in Fig. 68.6. If the user types a particular letter or word and presses one of the available buttons from the options, then the application will bring up a page with a map of the selected line.
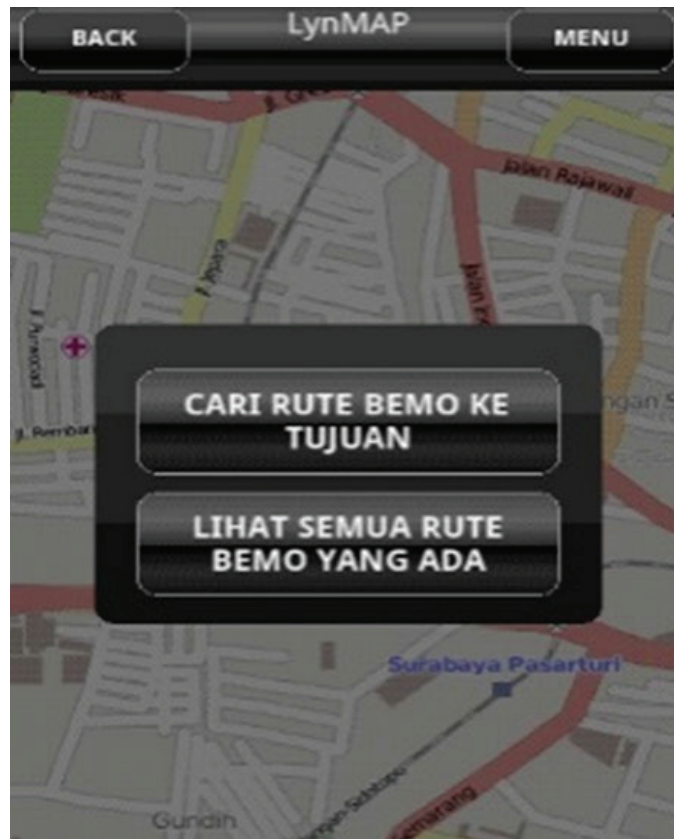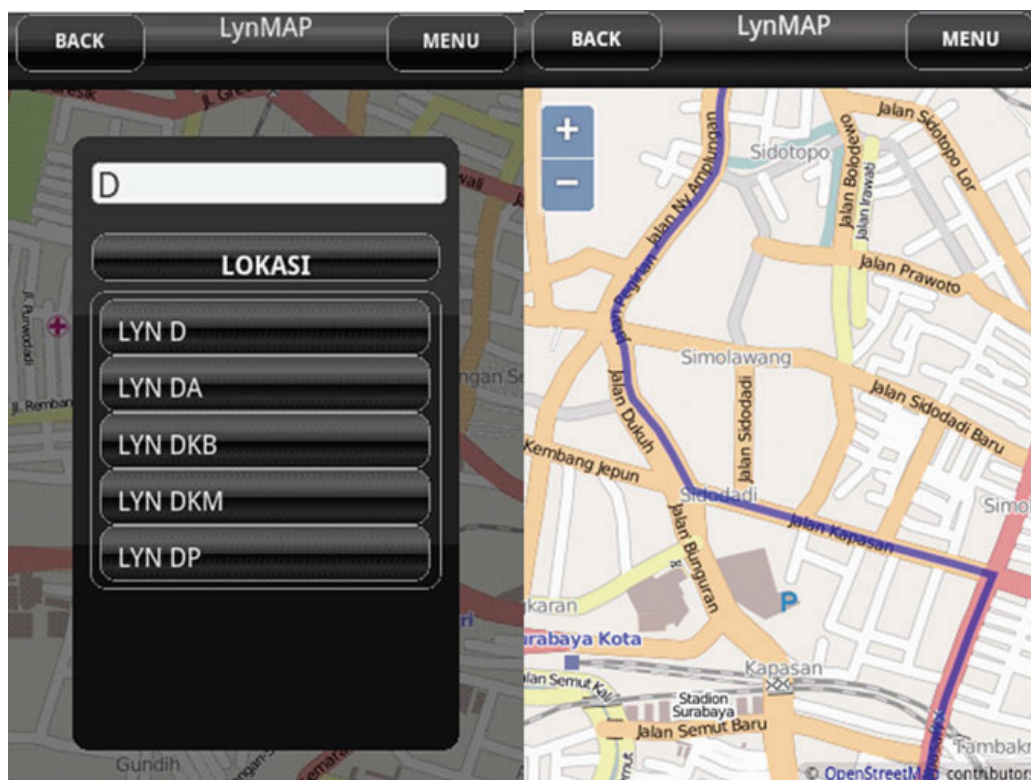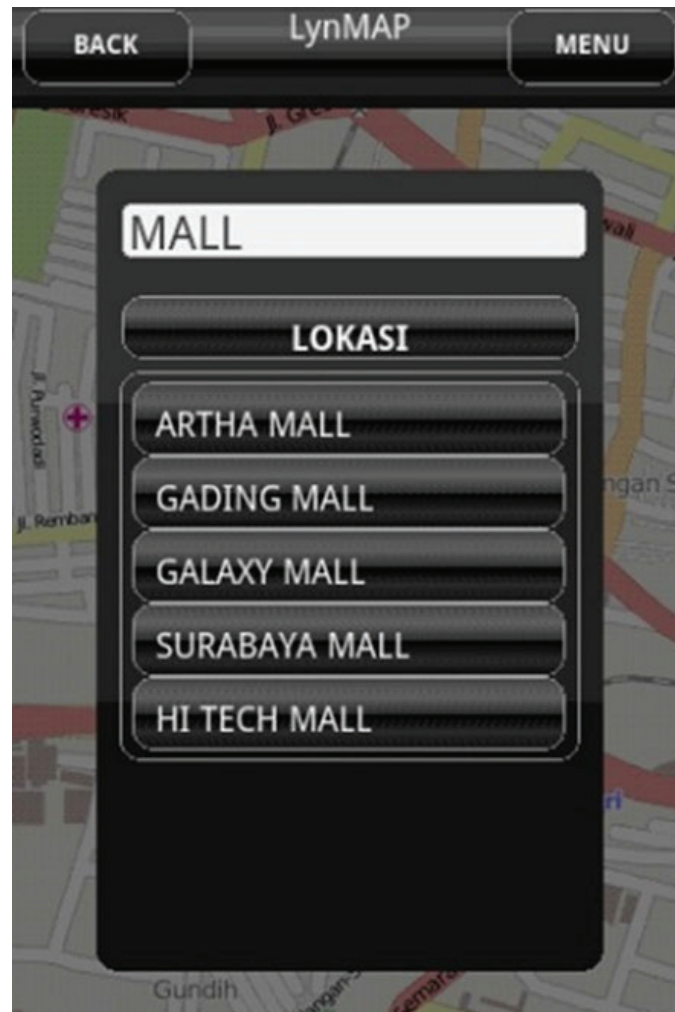
Fig. 68.5 Two options to choose

Fig. 68.6 Alternatives of Bemo routes and maps of the route

Fig. 68.7 5 Point of Interest

If in Fig. 68.5 the user selects the option 'Point of interest', the screen will display options as shown in Fig. 68.7. If the user types a particular word (i.e. MALL) and presses 'ok' button, then it will display a maximum of 5 points of interest containing the word typed by the user. Once the initial location has been determined, it will show the route to be passed and the estimated fare to be paid as in Fig. 68.8.

Figure 68.8 shows that the user originally is on Nyamplungan Street and she or he wants to travel to Ambengan Street. The first time the user is suggested to go by Line A *bemo*. On the way s/he should get off from the Line A *bemo* on Tambaksari Street and walk to Kapasari Street to get on Line B *bemo*. Next she or he has to get off Line B on Tambaksari Street and walk to the destination on Ambengan Street.

When the user wants to see the resulting route s/he can press on the 'view map' button, then options will appear on the display as in Fig. 68.9 on smartphone (left) and tablet (right).
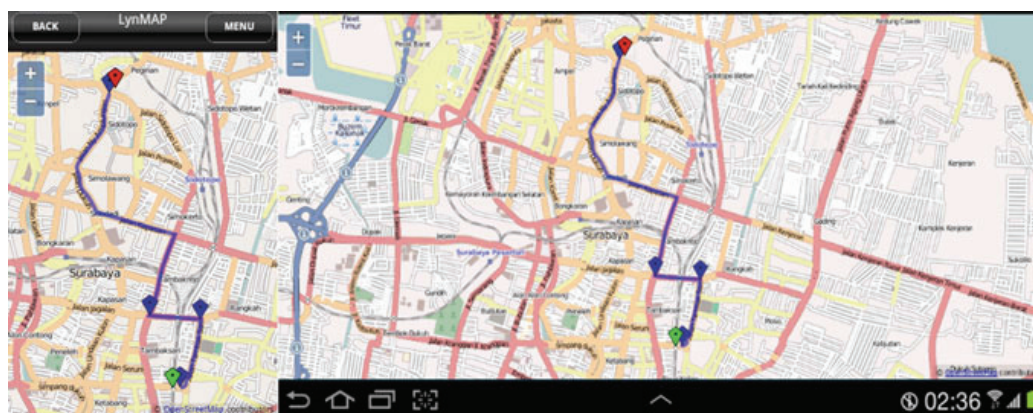
Fig. 68.8 Information of routes

Fig. 68.9 The routes generated on the map (On Smartphone and Tablet)

References

1. Angkutan Massal 'Sekarat'. Available: http://www.surabayapost.co.id/?Mnu=berita&act=view&id=0b8729fc9c2f434ea5ffb8252a78680c&jenis=b706835de79a2b4e80506f582af3676a (2013)
2. Watkins, K., Ferris, B., Borning, A., Rutherford, S., Layton, D.: Where is My Bus? Impact of mobile real-time information on the perceived and actual wait time of transit riders. Transp. Res. Part A **45**, 839–848 (2011)
3. Setiabudi, D.H., Tjahyana, L.J.: Interactive map routes for public transportation in Surabaya running on smartphones and tablets. ARPN J. Eng. Appl. Sci. **9**(10), 1811–1816 (2014)
4. Yulianto, N., Waluyo, Tomi, B., Suryadi.: Desain Web untuk Sistem Informasi Angkutan Umum di Jakarta. TELAAH Jurnal Ilmu Pengetahuan danTeknologi 30 no. 2 (2012)
5. Kor, M., Oksma. E.: Native, HTML5, or hybrid: understanding your mobile application development options. http//wiki.developerforce. com/page/Native,_HTML5,_or_Hybrid:_Understanding_Your_Mobile_Application_Development_Options (2013)
6. Grazella, M.: Android to remain champ, Windows to pick up steam in 2013. http://www.thejakartapost.com/news/2013/01/05/android-remain-champ-windows-pick-steam-2013.html (2013)
7. Jonathan, S.: Building iPhone Apps with HTML, CSS, and JavaScript. O'Reilly Media, Inc., USA (2010)
8. Charland, A., Leroux, B.: Mobile application development: web vs native. Commun. ACM **54**(5), 49–53 (2011)