

The Digital Age of Campus Maps on Mobile Devices

Fan Wu¹, Dwayne Clarke¹, Jian Jiang¹, Asif Baba¹, Sherman Buford²

¹Computer Science Department, Tuskegee University, Tuskegee, USA

²Police Department, Tuskegee University, Tuskegee, USA

Email: wuf@mytu.tuskegee.edu, DClarke0412@mytu.tuskegee.edu, jjiang8341@mytu.tuskegee.edu, ababa@mytu.tuskegee.edu, bufords@mytu.tuskegee.edu

Received 3 April 2016; accepted 16 May 2016; published 19 May 2016

Copyright © 2016 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Developing mobile applications have always been a rising topic in the technology world. With the recent development in technology, mobile applications play an important role in various applications throughout the world. Mobile applications are constantly evolving. There are several ongoing research and developments in both industry and academia. In this paper, we present the design and implementation of a mobile application that creates an electronic map or e-map application for the campus of Tuskegee University. The goals for this mobile application are to make the campus map easier and user-friendly for parents, visitors, and students using mobile devices. With this mobile application, the users will be able to search and find campus buildings, as well as give feedback on the application to eliminate the need for paper documentation.

Keywords

Mobile Application, Mobile Device, E-Map

1. Introduction

There has been extensive research on mobile applications in recent years, and there has been many successfully applications. As the number of people who own and use cell phones continues to grow, so does the use of smart phones and ninety-one percent of the US adult population currently owns a cell phone, and of that total, 61% are smartphones [1]. In the technology field, mobile applications are increasing more and more all over the world.

Historically, many institutions of higher learning are criticized for their failure to embrace modern technology. The rapid growth of mobile adoption—2/3 of Americans now connect to the Internet via a mobile device. It has forced some colleges and universities to jump on the mobile app bandwagon. From mobile bookstores to mobile

dining halls, colleges are developing some innovative mobile applications—and students are increasingly moving to the mobile Web in order to stay connected with social and current events at their schools [2].

An example of this innovative approach is the University of Alabama Mobile Application. The University of Alabama wants students to stay connected, even while on the move. This suite of apps provides instant access to the campus directory, map, news, Crimson Tide sports, library catalogs, class schedule, etc., this is the official mobile app for The University of Alabama, and it can be downloaded from the following locations [3]. Learning the concepts for mobile applications is more important now than ever. This is why there is continuing research and development mobile applications for the academic and technology world.

Building any mobile application can be very difficult. The higher the complexity of a mobile application, the more challenging it will be to complete the application. Simple tasks such as creating functions, classes, and designs are very important to successfully achieve the goals. Moreover, the mobile application's features play an important role in the application's performance. Finding connections on how a mobile application work is the central focus for mobile application development. Therefore, programmers must think rationally and logically in order to find solutions for mobile applications problems. Electronic maps also have the potential to help overcome a number of challenges currently plaguing the electric utility industry, including the aging workforce, operational inefficiency, emergency response preparedness, and dynamic workload management [4].

In this paper, a description of our application for designing and implementing an e-map based mobile application is illustrated. The e-map mobile application described in this paper is Tigermap. It is designed to make the campus map process much easier and user-friendly for students and visitors to use. Tigermap is a mobile application that creates a digital or electronic campus map for the students and guests at Tuskegee University. Some of the features of the mobile applications are to locate and describe the buildings on the campus. The main software used to develop this mobile application was the Android SDK with Eclipse the Android SDK is a software development kit used to develop the applications for the Android platform [5]. Most applications are written using the Java programming language.

The rest of the sections of the paper are organized as follows: Section 2 describes Tigermap Design. Section 3 defines System Architecture. Section 4 presents the Tigermap Algorithm. Section 5 is the Application Results and Section 6 concludes this paper with suggested recommendations for future research.

2. Tigermap Design

Tigermap is designed into number of layouts which are interconnected with each other to form a complete application. In order to implement a successful design and implement an application, some information is needed to complete the application, which is presented in this section.

2.1. Main Layout

The Main Layout is one of the very important components, because it is the beginning structure of the application. It can be considered as the critical layout for the application. This is where all of the layouts connect with one another. The layout includes three buttons that connect to three different classes of Java, as shown on **Figure 1**. The Tigermap design uses several different layouts throughout the application.

The design of this layout represents on how the application is upheld. Another important thing to consider, the

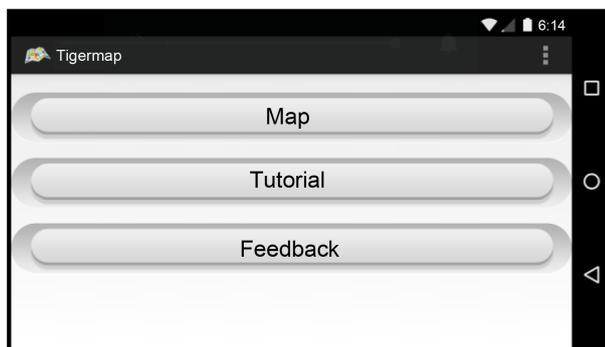


Figure 1. Interface for Tigermap.

more layouts added to the application, the more complex the application will become. Therefore, the creator decides the design to create a successful application.

2.2. Map Layout

The Map Layout is in charge of searching and finding buildings on campus. This is the most important layout in the whole application.

The Map Layout uses TileView widget which is a subclass of ViewGroup that provides a mechanism to asynchronously display tile-based images, with additional functionality for 2D dragging, flinging, pinching (double-tap to zoom), adding overlaying Views (markers), and providing built-in Hot Spot support, dynamic path drawing, multiple levels of detail, and support for any relative positioning or coordinate system [6].

The design of this layout has created a digital map to search and locate buildings on campus, as shown in **Figure 2**. Furthermore, another important thing to consider the application must be able to connect to the Internet or use cellular data to use the application.

2.3. Dialog Layout

The Dialog Layout shows the user the building selected. It is the heart of the application's function. Without this layout, it would be impossible for the application to get directions to the particular building on campus.

The Dialog Layout uses Image, Scroll, and Two TextView for this application. The ImageView loads or displays an image, such as an icon, logo, and a picture. Some of the features of ImageView are scaling and tinting an image. For the Dialog Layout, we used ImageView to display an image of a building the user has selected. The TextView displays text to the user and may allow the user to edit the text. Some of the features of TextView are to change the size, color, and font of the text. For Dialog Layout, a TextView displays text of the name and information to the building. The ScrollView allows the user to scroll up and down the dialog box to select a building.

The design of this layout allows the user to get directions to a building. Once, the user activates the Dialog Layout; the directions for the particular building on campus are displayed.

2.4. Tutorial Layout

The Tutorial Layout helps the user to learn how the application works. It is the eyes of the application's function. Without this layout, it would be very difficult to understand how the application works. The main purpose of Tutorial Layout is to create a list of video tutorials for the user to understand, how the application works and to use the application properly.

The Tutorial Layout uses three different XML files, which are List view, Tutorial, and Video that has a variety of views. For ListView, we used two TextView and one ImageView, which we used in the Tutorial Layout to create a list of the video tutorials. For the tutorial, we used TextView and ListView for this application. The ListView shows a list of items in a vertical form. Some of the features of ListView can display text or image item. For Video XML file, we used WebView displays web pages. Some of the features of WebView are used to navigate, zoom, and search web pages.

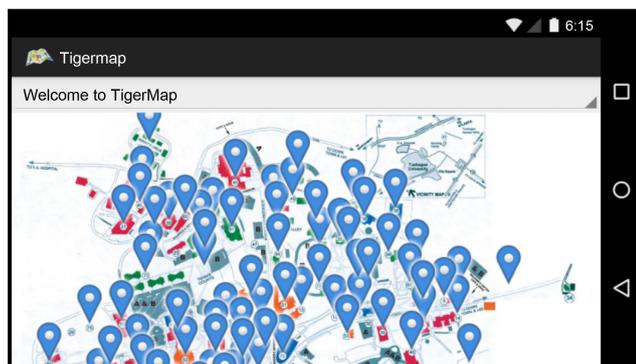


Figure 2. Java. layout for map.

2.5. Feedback Layout

The Feedback Layout helps the creator on how to improve or enhance the application. It is the backbone of the application's function. The design of this layout focuses on helpful information to improve the application.

Similar to the video XML file, the Feedback Layout uses a WebView that has the ability to view web pages. The WebView will send the user to a website with a feedback survey page to rate the android mobile's application.

3. System Architecture

3.1. Android Eclipse

Eclipse is a community for individuals and organizations who wish to collaborate on commercially friendly open source software. The projects are focused on building an open development platform comprised of extensible frameworks, tools and runtimes for building, deploying and managing software across the lifecycle [7]. Android Development Tools (ADT) is an extension of Eclipse in creating Android projects or applications. In other words, users can create Android projects, code and test them on virtual devices to see how the project or application works.

Moreover, ADT is a good way to get started with an Android project or application. ADT comes with many guides, tools, debuggers, and custom XML editors to give the users an extra boost in developing Android projects and applications.

3.2. Programming Languages

Android Eclipse is written in many different computer languages from C++, C, FORTRAN, Python, Perl, Scala, Ruby, JavaScript, Scheme, and Java to create an android application. The leading language for Android Eclipse is Java.

Java is a programming language expressly designed for use in the distributed environment of the Internet. It was designed to have the "look and feel" of the C++ language, but it is simpler to use than C++ and enforces an object-oriented programming model. Java can be used to create complete applications that may run on a single computer or be distributed among servers and clients in a network. It can build a small application module or applet for use as part of a Web page. Applets make it possible for a Web page user to interact with the page [8].

Moreover, Java is an object-oriented program. An object can use a class of objects and receive the code for a normal class. Moreover, Java executes on a Java applet to make the whole process run even faster than before.

3.3. Google Maps

Google Maps is a Web-based service that provides detailed information about geographical regions and sites around the world. In addition to conventional road maps, Google Maps offers aerial and satellite views of many places. In some cities, Google Maps offers street views comprising photographs taken from vehicles [9].

Some of the main features of Google Maps are directions for walking, bicycling, driving, and public transportation. Furthermore, Google Maps for Mobile can use GPS location of the mobile device, and the Google Street View allows users to view a location from the street level. Google Maps was used for our application to obtain directions from the user's current position to the particular building on campus that was selected.

3.4. Mobizen

Mobizen is an Android app run and developed by R-support that allow users to display their mobile device on a computer screen. Some of the features of Mobizen are to record, take screenshots, and mirror mobile device to a computer screen. Mobizen was selected for our application to create video tutorials.

3.5. YouTube

YouTube is a free video sharing and streaming website. This makes it easy for users to create, upload, and edit their own videos to be shared with others. Some of the features of YouTube are to search, watch, playback, upload, and change the quality of a video. We use YouTube for our application to upload our video tutorials.

3.6. Survey Monkey

Survey Monkey is an online survey service where users can customizable surveys and gain feedback information. We use Survey Monkey to obtain feedback to see what improvements are needed to create a better application.

3.7. Mobile Device

A Mobile device is a small or portable computing device that can normally fit in someone's hand or pocket. Some mobile devices can do similar or the same applications as a laptop or even a desktop computer can provide. These mobile devices include smartphones and tablets.

For our android application, we use several devices to ensure that android application works properly. These devices include Motorola Moto X, Samsung Galaxy Tab 2, and Samsung Galaxy Tab 4 mobile device.

1) Motorola Moto X

The Motorola Moto X is a smartphone device that runs on Android 4.2.2 Jelly Bean Operating System, which updates to Android 5.1 lollipop, and has the energy of a 1.7 GHZ dual core CPU. Some are the feature Android 5.1 lollipop are quick setting, HD voice, new API, screen pinning, device protection, volume, and new animations.

2) Samsung Galaxy Tab 2

The Samsung Galaxy Tab 2 is a smart device that runs on Android 4.0 Ice Cream Sandwich Operating System and has the energy of a 1.2 GHZ dual core CPU. Some are the feature multitasking, instant voice, unified API for Tablets and Phones, quick responses, and screen shoot.

3) Samsung Galaxy Tab 4

The Motorola Moto X is a smartphone device that runs on Android 4.4 Jelly KitKat Operating System, which update to Android 5.1 lollipop. In addition, it includes a MicroSD card jack that can add even more memory to the device and has the energy of a 1.2 GHZ dual core CPU.

4. Tigermap Algorithm

Tigermap Algorithm is divided into countless classes that cooperate with one another in order to create a successful application. To implement and create this application, some information is needed to make the application work. This section will describe the activity.

4.1. Main Activity

The Main Activity is the class that will connect with all of the other classes. This is a most important class for the Tigermap application. Without the Main Activity class, it will be very difficult or impossible to connect to all of the other classes.

The Main Activity includes three custom buttons or icons that communicate what action happens when a button is clicked or pressed by a user. For our android application, we use this button to connect with other classes in our application by using intent, OnClickListener, OnClickListener.

Intent is an abstract description of an operation being performed. In other words, intent is an action or a message to want something to occur. Intent is used with activity, service, and broadcast to connect with a service. The Activity can start by event of an Activity given by start Activity. In our case, Intent carries needed data. Service is a section that performs operations behind the scenes. It starts a service to be performed at one time, which can pass start Service or Intent. Broadcast is a message to be received from the application. The broadcast will send a message or messages to other applications by a given Intent to be sent to the right broadcast.

OnClick is a method that the code is run to a listener that detects a click on the button, icon, or image. The actual tap or click activates the method. Therefore, when users unclick on the button, it will trigger the code to execute and the method when it is called.

OnClickListener is a callback function that invokes when a button is pressed. When OnClickListener is created, it will override the onClick method. This method will perform every time the user press or tap the button. The Algorithm 1 labeled, "Tiger Map", displays the code of the Main Activity class presented in [Figure 3](#).

4.2. Map Activity

The Map Activity is the class that search and locate buildings around campus. It is one of the most important

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    final Context context = this;
    setContentView(R.layout.activity_main);

    //Declare the button for map, tutorial, and feedback
    final Button map = (Button) findViewById(R.id.map_button);
    final Button tutorial = (Button) findViewById(R.id.tutorial_button);
    final Button feedback = (Button) findViewById(R.id.feedback_button);

    //Create a button event for map button
    map.setOnClickListener(new OnClickListener ()
    {
        public void onClick(View v)
        {
            //Create a intent to go to Map class
            Intent intent = new Intent(context, Map.class);
            startActivity(intent);
        }
    });

    //Create a button event for tutorial button
    tutorial.setOnClickListener(new OnClickListener()
    {
    public void onClick(View v)
    {
        //Create a intent to go to Tutorial class
        Intent intent = new Intent(context, Tutorial.class);
        startActivity(intent);
    }
    });

    //Create a button event for feedback button
    feedback.setOnClickListener(new OnClickListener()
    {
    public void onClick(View v)
    {
        //Create a intent to go to Feedback class
        Intent intent = new Intent(context, Feedback.class);
        startActivity(intent);
    }
    });
}

```

Figure 3. Main activity code of algorithm 1.

classes in the Tigermap application. The Map Activity includes Tile-view, Linear Layout, Spinner, Coordinates, Icon Marker, and Zoom Function.

Tile-view, which we mentioned earlier, displays tile-based images. Linear Layout is a view that aligns in one direction vertically or horizontally. In other words, it is classified as the controller of the linear style; whether, it is horizontal or vertical. The spinner is a way to pick or select one value from a list. The Spinner displays a dropdown menu with other selections or values that users can select. Coordinates can locate buildings on the map. The Icon markers indicate a single location or identify locations on the map. Developers can also customize colors or replace a marker icon with a custom image. The Zoom expands an image, or, in our case, the map from the normal screen to fill the screen.

A Map Activity starts by an embed of the campus map to android assets from there using Java coding to enable to ability to zoom in or out. With this feature, we added the zoom function for android project by generating 4 zoom levels. Once completed, we added tile-view in order to display the campus map properly in order to use the image tool called, Magick. It helped to generate all the needed content for the map image. The final feature we added for that Map Activity is creating a spinner and icon markers for all of the buildings for campus, in order for the users to select the building from the spinner.

4.3. Dialog Activity

The Dialog Activity is the class that creates a dialog box showing the particular building the user selected. This is a significant class in the Tigermap application. The Dialog Activity includes a dialog box fragment, title for dialog box, and a negative and positive button.

Dialog Box is a fragment that displays a dialog window or box that appear at the top of the active window or box. The fragment decides what to show, and dismiss in the dialog box. The Title for Dialog Box displays the title text in the dialog box. The Negative button cancels the action or event. Moreover, the Positive button accepts the action or event.

The Dialog Activity is implemented by adding a dialog box window function to Tigermap and integrates with Google Maps. After adding dialog window function for Tigermap, we create Title and Message for Alert Dialog Windows. In our case, the title of the dialog box is Tiger Map and message is the name and information about the building the user selected. Once completed, we added the image of the building. Lastly, we added two buttons; a negative and positive button, as shown in [Figure 4](#). The Negative button is the cancel button in our application, and the positive button opens an URL to Google Maps from your location to the location of the building the user selected.

4.4. Tutorial Activity

The Tutorial Activity is the class with the purpose to teach users that the Tigermap works. This is a key class in the Tigermap application. The Tutorial Activity includes List, VideoView, WebView, and Adapter for this class.

A List is a set of elements that maintains in an order. VideoView displays video files by loading images from many sources. Some of the features are scaling and tinting. As mentioned earlier, WebView gives the user the ability to view web pages. Adapter access to data items and makes a view for each item of data.

The Tutorial Activity starts by creating a list which contains ImageView, and TextView for each of the tutorial videos. Once completed, we upload our videos to YouTube. This can be achieved by using a WebView to link the application with YouTube.

4.5. Feedback Activity

The Dialog Activity is the class that creates a dialog box showing the particular building the user selected.

The Feedback Activity is the class that allows user feedback on the application. This is a basic class in the Tigermap application. The Feedback Activity uses a WebView and WebSettings. WebView not only gives the user the ability to view web pages, but also open pages without opening the web browser. WebSettings is an objective that maintains the settings of WebView. The Feedback Activity uses WebView loads the URL link to survey website, Survey Monkey, to give feedback on the application.

5. Tigermap Algorithm Experimental Results

We used android mobile devices to test our algorithm, and to make sure that our Tigermap application worked correctly. We used three android mobile devices, which are the Motorola Moto X, Samsung Galaxy Tab 2, and Samsung Galaxy Tab 4 to test all the layouts, classes, and functions of Tigermap. This experiment was created and tested our application to see if it is a practical and consistent tool for parents, visitors, and students of Tuskegee University. Various trials and experiments were completed to establish best practices for users to search and find campus buildings, video tutorials, and give user feedback on the application. This application is best fitted to achieve these tasks on newer android devices where TileView widget can be used. Currently, the application can work if the user connects to the Internet or use cellular data, and allow Google Maps to share the user's location. The experimental results reveal that our mobile application can be used for parents, visitors, and students of Tuskegee University to search and locate buildings on campus, find video tutorials, and give feedback, as illustrated in [Figure 5](#).

6. Conclusion and Future Work

In this paper, we designed, created, programmed, and implemented a mobile application using a wide variety of classes and layouts to search, locate buildings on campus, find video tutorials, and give feedback. Originally, it

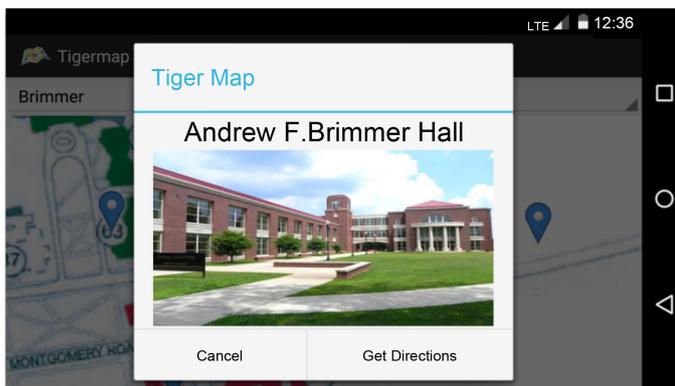


Figure 4. Map from the class to Brimmer Class Dialog Window.

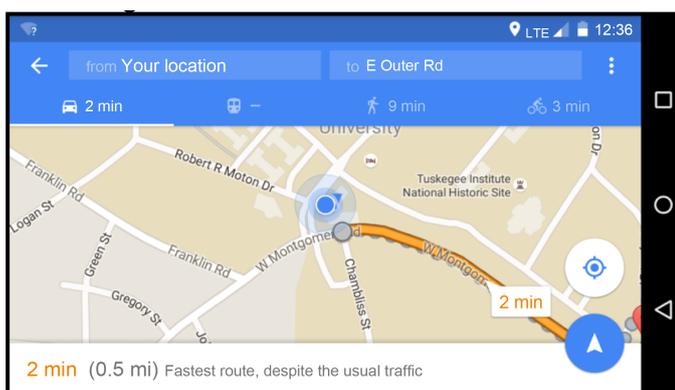


Figure 5. Result of Tigermap Map activity.

was a bit of a struggle, but the application was successful. The application should function very well on many different types of android mobile devices with Internet or use a cellular data connection. However, it may present future challenges in a real world scenario, since, not all the locations on campus have Wi-Fi or a good cellular connection. At times, users may not be able to use the application. Moreover, the video tutorials do not give the user the ability to make the videos go full screen. Further research and application testing with Tigermap will involve improving or enhancing the design which makes the application even more user-friendly, embeds the ability to make the videos go full screen, and creates a database that allow organizations on campus to add their cultural, educational or social events.

Acknowledgements

This work has been supported in part by US NSF grant # DUE-1241670 and US Department of Homeland Security Scientific Leadership Award grant # 2012-ST-062-000055.

References

- [1] (2016) The Future of Mobile Application. University of Alabama, Birmingham. <http://businessdegrees.uab.edu/resources/infographics/the-future-of-mobile-application/>
- [2] Lynch, L. (2012) 5 Colleges with Great Mobile Apps. <http://college.usatoday.com/2012/08/15/5-colleges-with-great-mobile-apps/>
- [3] The University of Alabama (2011) The University of Alabama Mobile Application. <http://app.ua.edu/>
- [4] Vadari, M. and Burck, M. (2006) Why Electronic Maps Are Essential to Electric Operations. http://www.elp.com/articles/powergrid_international/print/volume-11/issue-9/features/why-electronic-maps-are-essential-to-electric-operations.html
- [5] Beal, V. (2016) Android SDK. http://www.webopedia.com/TERM/A/Android_SDK.html

- [6] Moagrus, M. (2016) Moagrus/TileView. <https://github.com/moagrus/TileView>
- [7] The Eclipse Foundation (2016) About the Eclipse Foundation. <http://www.eclipse.org/org/>
- [8] Rouse, M. (2007) Java. <http://searchsoa.techtarget.com/definition/Java>
- [9] Rouse, M. and Gibilisco, S. (2013) Google Maps. <http://whatis.techtarget.com/definition/Google-Maps>