Juan Toledo
Faculty Advisor: Dr. Feng Wang
School of Mathematical and Natural Sciences
New College of Interdisciplinary Arts and Sciences

Wi-Fi Troubleshooting App Development in Android

Introduction

Mobile devices have become a substantial tool in the day-to-day operations of government, enterprises, businesses, and private owners. Wireless networking is an ever evolving technology to provide communication among mobile devices. More and more private owners are incorporating Wi-Fi into their homes and businesses to take advantage of the easy-of-us and freedom that these technologies offer. While the technology is spreading and being adapted widely, the tools to monitor and maintain Wi-Fi networks and the knowledge on using these tools is seemingly out-of-reach to people who had not devoted to studying these technologies.

Existing Tools

Wi-Fi diagnostic applications do exist on the Android platform for paid and free download. We studied two popular ones: *Wifi Analyzer* by Farproc and *Netgear Wifi Analytics* by Netgear.

There are two problems that are common between these two applications:

1) The organization of the tools and user-interface are confusing and difficult to use.

2) The omission of instructions on how to use the tools make the application unusable to those who do not have prior knowledge, or unwilling to conduct their own research on each tool.

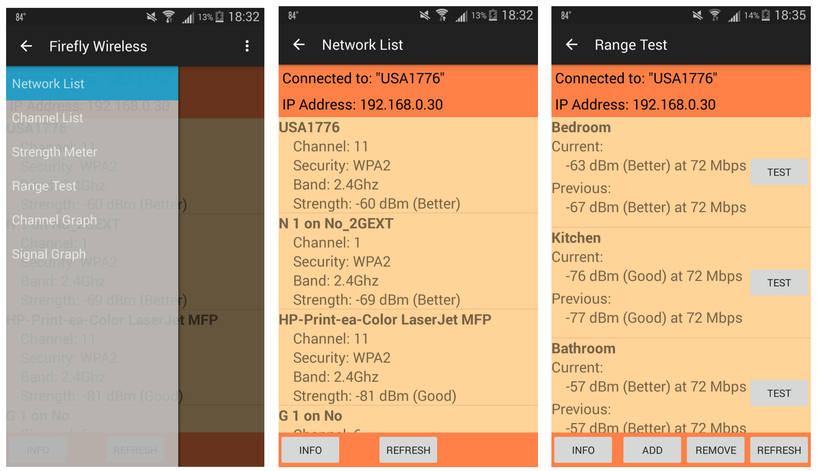

Figure 1: Screenshot from the application: The navigation drawer (left), Network List tool (center), Range Test tool (right)

Purpose of the Project

In this project, we propose to build a diagnostic tool, called *Firefly Wireless*, to keep Wi-Fi maintained and operational. The tool needs not only to be robust, but also be intuitive so that every user, regardless of background, could find the tool useful. We adopt the Android development platform due to the wide-spread use of Android mobile devices and the fact that Android development is intuitive and well-supported. We can release our tool to Google Play Store so anyone could potentially manage their own Wi-Fi network.

Design

Firefly Wireless is organized with the end-user in mind, keeping every page as consistent with each other as possible. We use a *Navigation Drawer* (Figure 1, left) as the back bone of navigation for Firefly Wireless because it is one of the most common and intuitive navigation structures in mobile applications; it is simple and familiar to mobile users.

Outlined in Figure 2, each page adheres to a simple set of layout rules that keep the application uniform.

Each tool features an “Information” button that, when clicked, displays a dialog box with an introduction and basic instruction about the tool.

The functions of each tool is kept simple by utilizing the base Wi-Fi utilities and displaying organized information that is simple to understand.

This design addresses the problems found in the existing tools by using familiar and consistent patterns and giving the user a basic explanation of the tools.

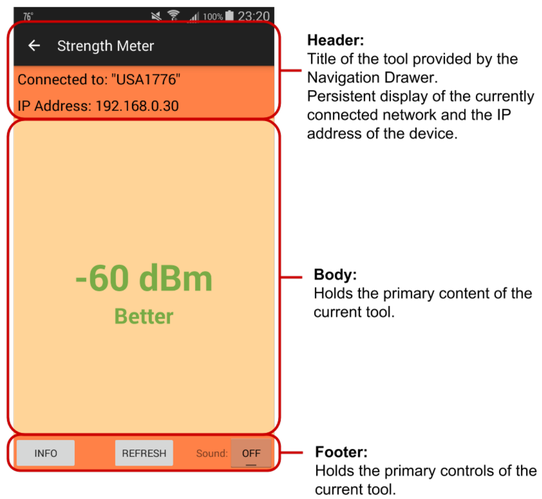

Figure 2: The Strength Meter tool. Each tool is designed to remain consistent with three rules: header, body, and footer.

List of Tools

- **Network List** – Scans and displays details of nearby access points.
- **Channel List** – Scans and displays the condition of the channels used by nearby devices.
- **Strength Meter** – Measures the live signal strength of the current connection.
- **Range Test** – Measures and records the signal strength and connection speed.
- **Channel Graph** – A graphical representation of the channel condition.
- **Signal Graph** – A graphical representation of the signal strength of nearby access points.

Implementation

Firefly Wireless primarily incorporates fragments, modularized code that represent behaviors or components of the user interface, and interchanging between them by making *FragmentTransaction* calls to the main activity (Figure 3). Individual fragments have a java file that defines the functionality separate from other fragments.

Each fragment that defines a tool in the application instantiate their own *WifiManager* differently to achieve the desired functionality.

AlertDialog and *AlertDialog.Builder* on all fragments to control windows that display information about the tools and ask for user input.

Scenarios

- If a user needs additional information about access points in their area that is not provided by their default utility, they can use the detailed descriptions from Network List in Firefly Wireless.
- If a user is setting up a new access point and needs to determine the wireless channel that is least conjected, they can use the Channel List in Firefly Wireless.
- If a user needs to map out the wireless coverage of their access point, they can use the Strength Meter in Firefly Wireless and read the live result as they walk around.

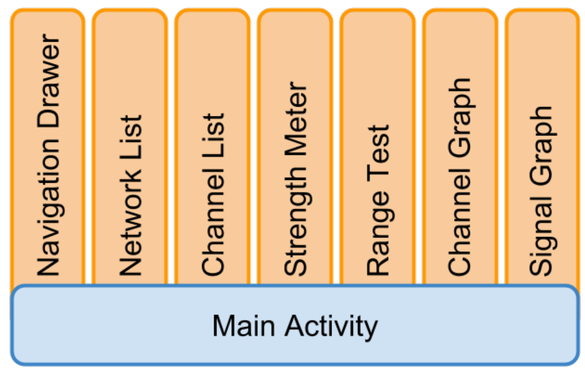

Figure 3: The typical Activity (blue) and Fragment (orange) structure in Android.