

Group: Mighty Group G, LLC

Members: Michael Catanzaro, Miguel Guzman, Katie Isbell, Ryan Schmitt, and Ethan Wells

Due Date: Thursday, October 10, 2012

Objective:

The main objective for this assignment was to create a work breakdown structure and gantt chart using our list of requirements. These documents appear below.

Work Breakdown Structure:

The work breakdown structure was created by reorganizing our requirements and writing steps for each individual task. The tasks were divided as evenly as possible amongst the team.

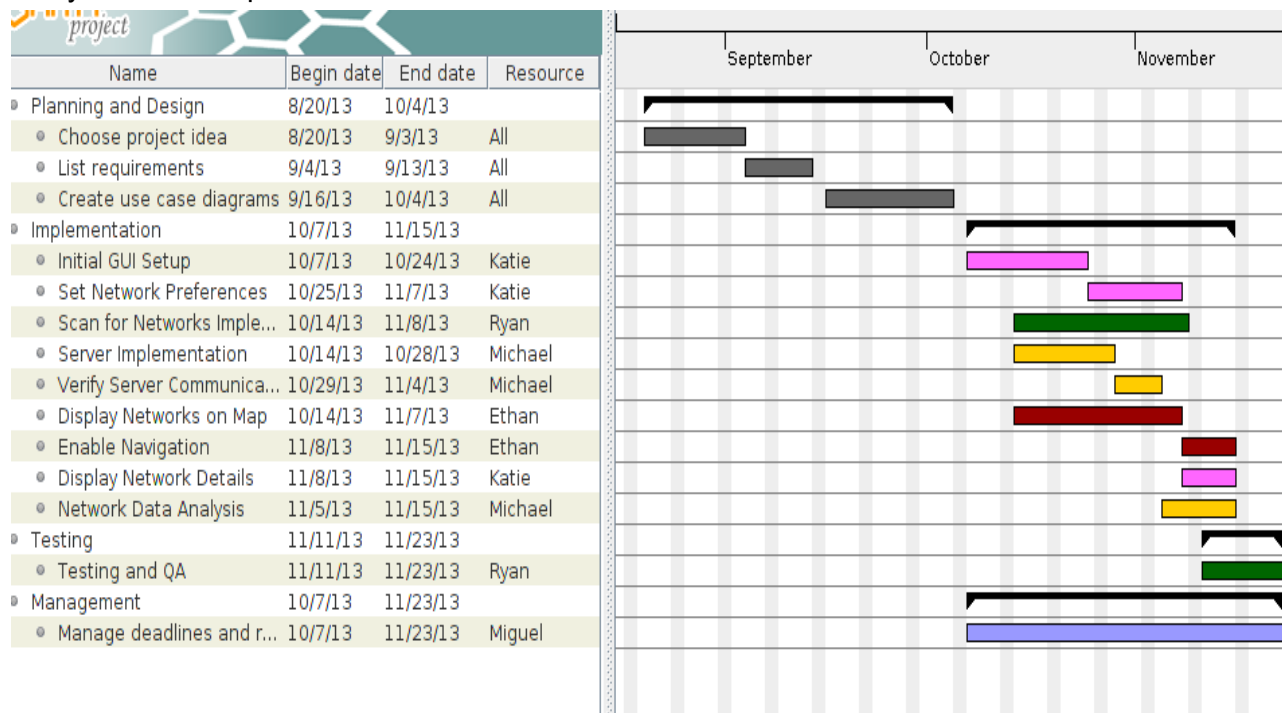
Task	Steps	Assigned to
Initial GUI Setup	Create a button on the main page that leads to a settings page	Katie
	Add buttons on the settings page to display all the application settings options.	
	Add a check box which will eventually be used for selecting the option to continually scan for networks	
	On the settings page, create a button that will bring us to the "About" page. Create the page containing information about our Application and credits to contributors.	
Set Network Preferences Implementation	Create the popup window that allows users to set the maximum distance of nearby networks. When the user sets a new distance value, be sure to update a variable that can be used in filtering out certain networks from being displayed on the map.	Katie
	Add the feature for the user to set the minimum signal strength.	
	Add the feature for the user to set the maximum number of networks to be displayed at one time.	
Scan for Networks Implementation	Create a popup window that appears when the user decides to do a custom scan	Ryan
	Within this popup window, display the names of all of the networks for which the user should attempt to make a connection. Check boxes should appear next to the name of each network.	
	Add a feature such that when the user chooses to scan for these networks, it will only attempt to connect to those networks that are selected (checked).	
	Add feature to allow users to decide whether to continually scan for networks within range. If continual	

	scan is selected, the application should attempt to connect to all networks every few seconds.	
Server Implementation	Write a hello world server.	Michael
	Add an autotools build system.	
	Write a simple test client.	
	Figure out how to store the WiFi network data on the server.	
	Write a real server.	
	Make the real server pass the test client's tests.	
	Make the server socket-activated by systemd (optional)	
	Hook up a systemd timer unit to purge old networks once per day (optional)	
Verify Server Communication	Develop a protocol that determines the rules of interaction between the client application and the server.	Ethan, Michael, Miguel
	Determine a format to save and parse GPS data, signal strength, and other phone information on the server.	
	Write and test functionality to retrieve and push information to the server from the client.	
Display Networks on Map	Write initial Java code to display a map on the Android device using Google Maps	Ethan
	Implement client-server interaction and retrieve map data.	
	Write code to parse file retrieved from server and store the network data.	
	Read through the stored data and plot the networks using Google Maps.	
	Add functionality to retrieve current location and move the map to center at that location.	
Display Network Details	Create the popup window that is displayed when a user clicks on a plotted WiFi spot on the map.	Katie
	Within the window, display the details of the specific WiFi spot that was selected, including the SID, signal strength, and GPS coordinates.	
Network Data Analysis	Write method to update a WiFi network. It will delete the old network from the server and insert the updated network with the new lat lon values and signal strength.	Ethan, Michael, Miguel
	Write method to get all of the networks pertaining to user-defined constraints, such as distance and signal strength.	
Testing and QA	Test individual components as they are created	Ryan

	Test the interaction between individual components	
	Test application to make sure it meets functional requirements	
	Try to break application - creatively	
	Test application's ability to approximate wifi origin (within a specified range)	
	Ask other students to test App and gather feedback on functionality and UI	
	Report feedback to group, discuss and correct project accordingly	
	Final Test of completed and corrected project, test for functionality, completeness, and design	
Management	Receive weekly reports from team members of current progress	Miguel
	Perform monthly evaluations of team members	
	Submit assignments in due time	

Gantt Chart:

Our Gantt Chart was created using Gantt Project and designed such that group members are working in parallel on different parts of the project. The deadline to complete the project is set for November 23, 2013, so that all group members will complete before Thanksgiving break and be ready for the final presentations.



Resources Chart:

Each task within our Gantt Chart has been assigned one resource, which implies that the group member assigned to a particular task is responsible for the completion of that task. The resources chart also shows that each group member has been assigned a particular role.

