Project Documentation

Android Based Situational Awareness: Moving Map

Tom Atnip, Susi Cisneros, Sam Kim, and Seth Troisi

February 3, 2013

Contents

Changes

Date	Description
September 13, 2012	Document started
September 13, 2012	Wrote Requirements and Questions to ask Raytheon
September 20, 2012	Created outline
September 20, 2012	Wrote Users/Stakeholders section
September 23, 2012	Wrote Key Needs, Alternatives, Risks, Documentation Metrics, and Code Metrics
September 24, 2012	Updated Requirements as per Gate 5 visit with Raytheon
October 8, 2012	Revised Requirements, added Assumptions and Opportunities, and updated Risks
October 15, 2012	Added Introduction and made Requirements into shall statements
October 25, 2012	Updated Requirement, added Performance Requirements
November 7, 2012	Updated Performance placeholder
January 26, 2013	Project Document Audit

1 Introduction

This document details all documentation associated with this project. It includes the problem statement, requirements, project plan, and metrics.

2 Problem Statement

2.1 User/Stakeholder Descriptions

2.1.1 Users

Soldier, Police Officers, and other Ground Personnel
The users of our program are seeking to maintain their situational awareness in locations which may not have connectivity to the Internet. Many of them use voice guided situational awareness technology, but in light of advances in mobile devices, they could receive this information in a visual manner.

2.1.2 Stakeholders

Raytheon
Raytheon's customers are mainly military organizations, many of which are using Raytheon's current situational awareness technologies. Raytheon is looking to update these technologies to keep their position as a leading provider of military systems.

JD Hill
JD is the client who proposed this solution. He is a major proponent of using mobile devices in a military application.

Doug Dusseau
Doug is the acting Technical Lead for this project.

Development Team
The development team on this project are graduating seniors who wish to learn more about the software development process and interaction with a client. They are very interested in learning more about developing on the Android platform.

2.2 Key Needs

ID	Need
N0	View map of surrounding area
N1	View points of interest on the map
N2	View current location on the map
N3	Map must not require internet access
N4	Map must be Android based
N5	Application must work on any size android device

2.3 Current Solution

The chosen mapping engine was OSMDroid. This engine came pre-built with offline map support, which handled the key functional requirement. The engine is also part of the Open source community which allows the code to be used free of charge, as long as licensing requests from the owner are met. There is also built in support for overlays.

2.4 Alternatives

All considered solutions to the proposed system require Internet access. Most other mapping engines explored were very domain specific (ie hiking or biking) and lacked significant amounts of documentation.

3 Requirements

3.1 Functional

ID	Requirement	Priority
FR0	The system shall let the user pan the map by a dragging gesture	Objective
FR1	The system shall let the user zoom using an on-screen button	Threshold
FR2	The system shall let the user zoom using pinch gestures	Objective
FR3	The system shall let the user zoom using double tap	Objective
FR4	The system shall store map tiles on the device	Threshold
FR5	The system shall display map tiles which are stored on the device	Threshold
FR6	The system shall be able to pull map tiles which are stored on a local server and store them on the device	Objective
FR7	The system shall georeference the location of the device	Threshold
FR8	The system shall let the user center on current location	Objective
FR9	The system shall store multiple map types	Threshold
FR10	The system shall let the user choose the map type	Objective
FR11	The system shall store points of interest as a map overlay	Objective
FR12	The system shall display points of interest overlays	Objective
FR13	The system shall let the user choose which overlays are displayed	Objective
FR14	The system shall let the user add custom points of interest	Objective
FR15	The system shall let the user choose which overlay the custom point of interest is added to	Objective
FR16	The system shall let the user create new overlays	Objective
FR17	The system shall display a compass	Threshold
FR18	The system shall let the user toggle heading/north up	Threshold
FR19	The system shall let the user change default settings via a settings menu found in the menu bar	Threshold
FR20	The system shall let the user access a help menu via the menu bar	Objective

3.2 Non-functional

ID	Requirement
NR0	The system shall run on Android platforms running at least version 3.0 (Honeycomb)
NR1	The system shall receive GPS data from a local server or the device
NR2	The system shall display properly on either mobile phones or tablets
NR3	The system shall use modular code
NR3	The system shall use the Android usability conventions

3.3 Performance

ID	Requirement
PR0	To be determined

4 Project Plan

4.1 Schedule

Week 0	27-Aug	31-Aug
	Kickoff meeting	
	Requirements Definition	
Week 1	3-Sep	7-Sep
	Requirements Definition	
	Architecture Development	
Week 2	10-Sep	14-Sep
	Requirements Definition	
	Architecture Development	
Week 3	17-Sep	21-Sep
	Requirements Definition	
	Architecture Development	
	Raytheon Site Visit	
Week 4	24-Sep	28-Sep
	Requirements Definition	
	Architecture Development	
	OSMDroid Demo	
	Android Local Network Demo	
Week 5	1-Oct	5-Oct
	Requirements Definition	
	Architecture Development	
	OSMDroid Demo	
	Android Local Network Demo	
Week 1	26-Nov	30-Nov
	Alpha	
Week 2	3-Dec	7-Dec
	Alpha	
Week 3	10-Dec	14-Dec
	Alpha	
Week 4	17-Dec	21-Dec
	Alpha	
	Alpha Demo	
Week 5	7-Jan	11-Jan
	Beta	

Week 6	8-Oct	10-Oct
	Requirements Definition	
	Architecture Development	
	OSMDroid Demo	
	Android Local Network Demo	
	System Requirements Review	
	Senior Project Presentations	
Week 7	15-Oct	19-Oct
	Preliminary Design Development	
	Environment Setup	
	Use Cases	
Week 8	22-Oct	26-Oct
	Preliminary Design Development	
	Environment Setup	
	Use Cases	
Week 9	29-Oct	2-Nov
	Sprint Establishment	
	Guided Coding	
	Use Cases	
Week 10	5-Nov	9-Nov
	Coding	
	Use Cases	
Week 6	14-Jan	18-Jan
	Beta	
Week 7	21-Jan	25-Jan
	Beta	
Week 8	28-Jan	1-Feb
	Beta	
Week 9	4-Feb	8-Feb
	Beta	
Week 10	11-Feb	15-Feb
	Beta	
	Senior Project Exposition	

Key
Activity
Sofware
Documentation

4.2 Assumptions

ID	Assumption
A0	There exists an open source mapping engine for Android devices
A1	The mapping engine does not require an internet connection to run
A2	Android devices can connect to a local server

4.3 Risks

ID	Risk
R0	Performance of the system
R1	Organizing data in the correct format in a timely manner

4.4 Opportunities

ID	Opportunity
O0	Finding a feature complete mapping engine

5 Metrics

5.1 Project

5.1.1 Documentation

The progress of the documentation will be tracked by breaking it down into three parts: the percent written and ready for review, the percent that has been reviewed, and the percent that is ready for delivery. The initial portion will encompass the percentage of the requirements, features, and other material that have been documented according to our currently known goals. A portion of the documentation will be considered in the reviewed stage once Dr. Wollowski and/or JD Hill have provided feedback and approval. Once a section of the documentation is in its final state (written, reviewed, and stable), it will be considered complete.

Percent Written: 100
Percent Reviewed: 100
Percent Complete: 100

5.1.2 Code

During the coding phase of this project, progress will be tracked by the features scheduled during an iteration and the number of features completed. Code will belong to one of five phases unwritten, written, peer reviewed, tested, or complete. Once code has been written and passes the required unit tests, it will undergo a peer review to check for good coding practices, clarity, and errors. After a peer review the functionality will then be required to pass integration tests. Once it has passed system integration, it will be considered complete and will be merged into the main branch of code.

Percent Written: 30
Percent Reviewed: 15
Percent Tested: 0
Percent Complete: 0

5.1.3 Testing

For the final phase of the project, progress will be measured by how many tests are passing. The tests that the software will be subjected to will be more thorough than the tests required for code to join the main branch. Most tests will be automated, but there will also be human factor tests.

Percent Passing: 0

5.2 Process

The process that the project is following will be measured by due dates met versus missed due dates.

Milestone Dates Kept: 0

5.3 Communication

Communication will be measured by how well the team feels that their needs are being heard and being taken care of, along with efficiency of meetings.

Team Confidence:	85
Meetings:	95

6 Questions

6.1