Wi-Finder
Use Case Specification

Submitted to:

Asst. Prof. Ma. Rowena C. Solamo
Faculty Member
Department of Computer Science
College of Engineering
University of the Philippines, Diliman

Submitted by:
Miguel Luis G. Posadas
Juan Gabriel C. Tamayo
Zechariah B. Jimenez

In partial fulfillment of academic requirements
for the course
CS 191 Software Engineering I
of the
1st Semester, AY 2014-2015

Unique Reference:

The documents are stored in *https://github.com/191GPZ1617A/wifinder*.

Document Purpose:

This document serves as a foreword to the project that the group attempts to build throughout the course proper. Stated below are the plans and specifications of the Wi-finder project and its feature to check available Wi-Fi hotspots and other internet places.

Target Audience:

The software is primarily targeted towards mobile users looking for a temporary public internet connection that will suit their purposes (messaging, email, etc.) while on-the-go. The initial release will be focused towards the UP community, and may be expanded if the project is received well.

Revision Control

History Revision:

Revision Date	***Person Responsible***	***Version Number***	***Modification***
09/29/16	All Members	1.0	Initial Document; Version number should match the one found in the footer.

Use-Case Name: 1.0 Check Available Spots

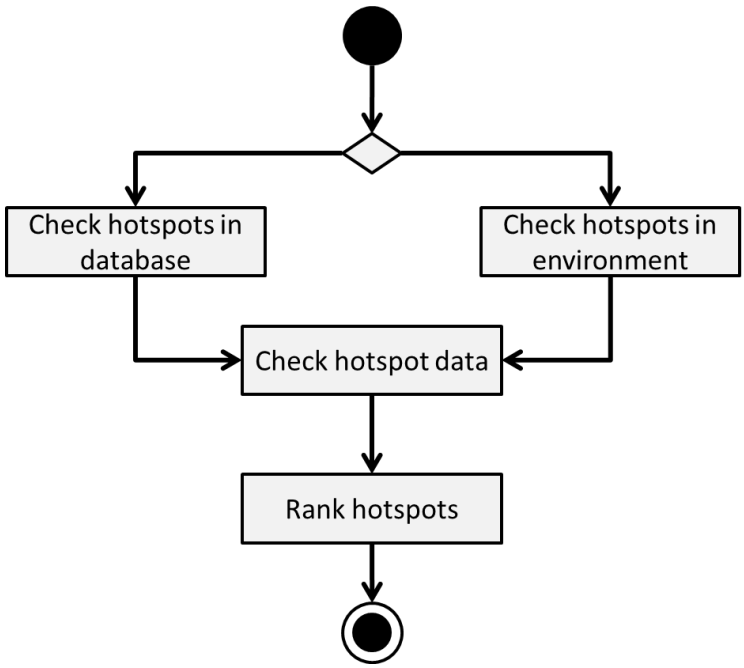
Description: Look for surrounding Wi-Fi hotspots and internet places, and gather all necessary info regarding the spots (e.g. signal strength given time of day). This can be done by checking either the environment or the device's last updated version of the database.

Preconditions: NONE

Flow of Events:

Scenario Name	***Description***
Scenario 1 (Basic Flow) Hotspots are searched in the environment	1. Environment is checked for hotspots 2. Information about hotspots is checked (e.g. connection speeds/quality) 3. Hotspots are shown and ranked appropriately
Scenario 2 Hotspots are searched in client-side database	1. Client-side database is checked for hotspots 2. Information about hotspots is checked (e.g. connection speeds/quality) 3. Hotspots are shown and ranked appropriately

Activity Diagram of the Flow of Events:



Postcondition: NONE

Relationships: User to system: System must be updated regularly via ratings or by adjusting the database in order to be accurate for the users. Users in turn must rate the system in order to keep the system running properly.

Special Requirements:

- Must have a working database for any build.
- Must have users rating the system to keep accuracy