Software Requirements Specification
Team Wakati
5.1.14

Adrian Chambers | Anthony Cristiano | James Doan | Daniel Green | Jesse Rosato

TABLE OF CONTENTS

1 INTRODUCTION

The Software Requirements Specification establishes criteria for successfully completing the project to the satisfaction of the sponsor. The remainder of this introduction gives a brief overview of this document.

1.1 Purpose

The purpose of this Software Requirements Specification document is to:

1 Clearly define all user functionality and quality attributes the product must exhibit in order to satisfy the sponsor's stated need, based on prior requirements elicitation.

2 Create a baseline for the design specification of the product.

3 Solidify and validate the understanding of requirements between the development team and the project sponsor to conclude the requirements engineering phase.

1.2 Scope

This document will detail all requirements that needed to complete this project. The SRS details product features, uses, assumptions, and dependencies. The SRS describes plans for implementation of the product, as well as system and design constraints. Models detailing the use cases and mock-ups of the intended design are also included.

1.3 Definitions, Acronyms, and Abbreviations

- Android: An operating system based on Linux, and designed primarily for touchscreen mobile devices such as smartphones and tablet computers.
- Asynchronous JavaScript and XML (AJAX): The practice of sending and receiving data (using XML or a similar format) to and from a server without loading a new page or preventing user interactions on the current page.
- Application Programming Interface (API): Implemented declarations of how a software component will interact with other software components. A common example of an API is a web service that provides data via a collection of resource addresses. This project will use APIs for preexisting tools to increase efficiency and avoid reinventing the wheel.
- Continuous Delivery (CD): The practice of "pushing" updates to a software system without causing system unavailability.

- <u>Favicon</u>**:** The small icon that appears in the browser's address bar when visiting a website, usually a small version of the site's logo.
- <u>Frequency of Use:</u> Denotes how often a given use case is expected to be activated: "high" describes use cases that will be activated during every application usage, "medium" describes use cases that will be used during most usages, and "low" describes use cases that be used rarely.
- <u>Geo-Location</u>: The practice of accessing the geographic location of a mobile phone or internet-connected computer terminal.
- <u>Hypertext</u>: Text that contains traversable references to other text. Internet "links" are the most common form of hypertext.
- <u>Hypertext Transfer Protocol (HTTP)</u>: "An application-level protocol for distributed, collaborative, hypermedia information systems" [1]. HTTP is one of the foundational protocols of the web, and is generally used to retrieve hypertext from a website (e.g. *http*://facebook.com).
- <u>JavaScript Object Notation (JSON)</u>: A data-interchange format, commonly used to pass data between the client and server sides of a web application. Many web service APIs deliver data in JSON format.
- <u>LAMP Stack</u>: LAMP is an abbreviation for Linux (operating system), Apache HTTP Server (web server software), MySQL (or an alternative relational database management system) and PHP (or an alternative scripting language). "Stack" describes the combination of that software into a solution set. The LAMP stack is a common solution for serving dynamic hypertext on the web.
- <u>MySQL:</u> An open source relational database management system.
- <u>Open Eligibility Project (OEP):</u> An open source taxonomy to categorize human services and human situations [2].
- <u>PHP: Hypertext Preprocessor (PHP):</u> A scripting language commonly used to dynamically produce HTML to be served to a web browser. In this project, it will also be used to receive, validate and process user-submitted data.
- <u>Priority:</u> Denotes the importance of a given use case: "high" describes use cases that are essential to the basic functionality of the application, "medium" describes non-essential use cases that model expected, but non-essential, functionality, and "low" describes use cases that model functionality that is desired, but that may be omitted from the application if time or resource shortages occur.
- <u>Relational Database</u>: A database built on the relational model, wherein data is organized into tables, with the data in each table representing a "relation", and the data in various tables being interconnected in relationships.

- Representational State Transfer (REST): A network-based architectural style that "define[s] a uniform connector interface" [3]. In modern practice, REST principles are commonly used to guide design decisions when building APIs for web services.
- RESTful: Jargon to describe software implementations that closely adhere to REST principles.
- Stale Data: Records and data that no longer reflects the current actual data of an organization.
- Software Development Life Cycle (SDLC): An iterative model describing the phases of a software development project: planning, analysis, design, implementation, and maintenance.
- SQL Injection: A technique used to attack data-driven applications. It involves a user exploiting security vulnerability by entering malicious SQL statements into an input field. If that input is not filtered, or filtered incorrectly, before being inserted into an SQL database, it could result in corruption or loss of data.
- User Interface (UI): The system by which users interact with software. This term is commonly used to describe the on-screen elements with which a user can interact to operate a piece of software.
- Viewport: A region of the screen used to display a portion of an image, akin to a photo mat.
- Extensible Markup Language (XML): A markup language that allows users to create their own markup elements (i.e. "extend" the language).

1.4 References

[1]   Fielding, R., Gettys, J., Mogul, J.C., Frystyk, H., Masinter, L., Leach, P., Berners-Lee, T. *Hypertext Transfer Protocol – HTTP/1.1*, Network Working Group RFC 2616, June 1999; http://tools.ietf.org/html/rfc2616.

[2]   "Open Eligibility Project", [online], http://openeligibility.org/, [April 14, 2014]

[3]   Fielding, R., "Architectural Styles and the Design of Network-based Software Architectures", Ph.D., ICS, Univ. California, Irvine, 2000.

[4]   Team Wakati Project Charter. Chambers, A., Cristiano, A., Doan, J., Green, D., and Rosato, J., "Project Charter", Team Wakati, Sacramento, CA, Mar. 13, 2014.

[5] "Open Street Map Project", [online],
     http://wiki.openstreetmap.org/wiki/Main_Page, [April 20, 2014]

[6] "Code4Sac activities statement", [online], http://code4sac.org/what-we-
     do/, [April 20, 2014]

[7] "About", [online] http://openreferral.org/about/, [April 20, 2014]

[8] Nielsen, J., "Website Response Times", [online],
     http://www.nngroup.com/articles/website-response-times/, June 21,
     2010, [April 4, 2014].

[9] "211Sacramento.org", [online],
     http://www.alexa.com/siteinfo/http%3A%2F%2Fwww.211sacramento.org,
     Alexa, [April 11, 2014].

[10] Luchaninov, V., "Alexa Rank – A Thorough Examination", [online],
      http://netberry.co.uk/alexa-rank-explained.htm, NetBerry, May 4, 2013,
      [April 11, 2014].

[11] "Coding Style Guide", [online], PHP Framework Interop Group,
      http://www.php-fig.org/psr/psr-2/, [April 14, 2014].

[12] "JavaScript Style Guide", [online], jQuery Foundation,
      https://contribute.jquery.org/style-guide/js/, [April 14, 2014].

[13] "PHP Documentation - PDO Drivers", [online],
      http://www.php.net/manual/en/pdo.drivers.php

1.5 Overview of Contents of Document

2 General Description. This section provides a refined view of the system's assumptions, constraints, requirements, and user characteristics. Use case models and interaction specifications are also presented here.

3 Specific Requirements. Details the functionality and qualities of Project Wakati, as requested by the sponsor and negotiated with Team Wakati. This section includes use case specifications, performance requirements, design constraints, and quality attributes.

4   **Document Approval Page.** This section contains the list of the key
    signatures necessary to approve the SRS, thereby agreeing to the scope and
    content of the requirements specified within the document.

2 GENERAL DESCRIPTION

This section provides a refined view of the system's assumptions, constraints,
requirements, and user characteristics. Use case models and interaction
specifications are also presented here.

2.1 Product Perspective

The product perspective specifies the software and hardware with which Project
Wakati will interact.

- **Data Sources.** This project is intended to showcase the work of the
  sponsor organization in improving access to public data. That work includes
  efforts to convert public data available in human-readable formats on the
  web into machine-readable formats, and to expose that machine-readable
  data for use by apps like Project Wakati. Project Wakati will interact with one
  such project (currently under development by the sponsor organization) via
  either an HTTP-based RESTful API or direct SQL queries, depending on the
  progress of that project.

- **Third-Party API.** Project Wakati will use map data from the open-source
  OpenStreetMap project [5]. Project Wakati will interact with that map data
  using the Leaflet.js open-source JavaScript library [5].

- **Architecture.** Project Wakati depends on the LAMP software stack. That
  dependency is described in detail in "Transferability and Conversion" (3.5,
  section 5) and "Site Adaptations" (3.5, section 8).

- **Hardware.** Project Wakati depends on hardware (to be chosen by the
  sponsor) capable of running the LAMP stack. See "Site Adaptations" (3.5,
  section 8), for details about specific LAMP stack requirements.

2.2 Use Case Models

These models show how the user will interact with the system. The system will
allow desktop and mobile Users to search for public services (Figure 1).
Searching can be done in a number of different ways (Figure 2), such as:
searching by a need (Figure 3), searching by a situation (Figure 4), searching by

both (Figure 5), or browsing the map for services (Figure 6). The user can then interact with the results (Figure 7).
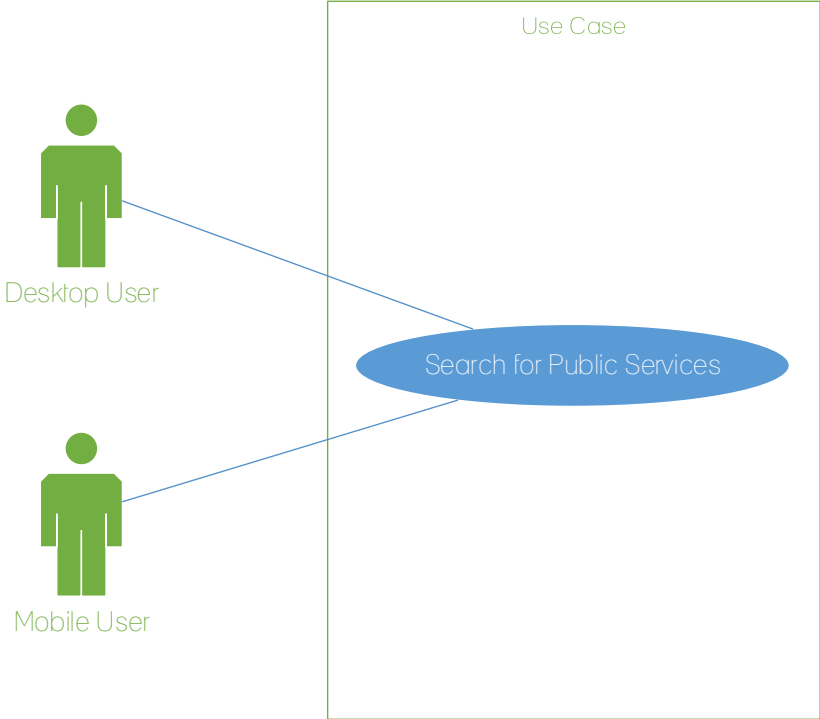
Figure 1. Use Case Overview
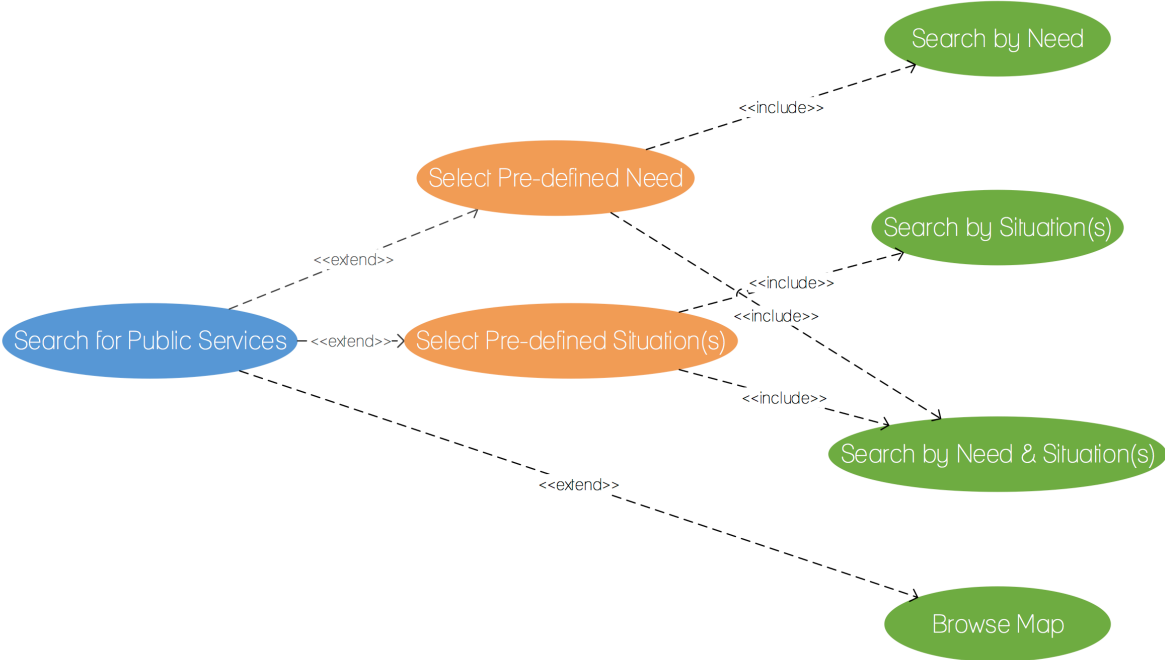
Figure 2. Search for Public Services
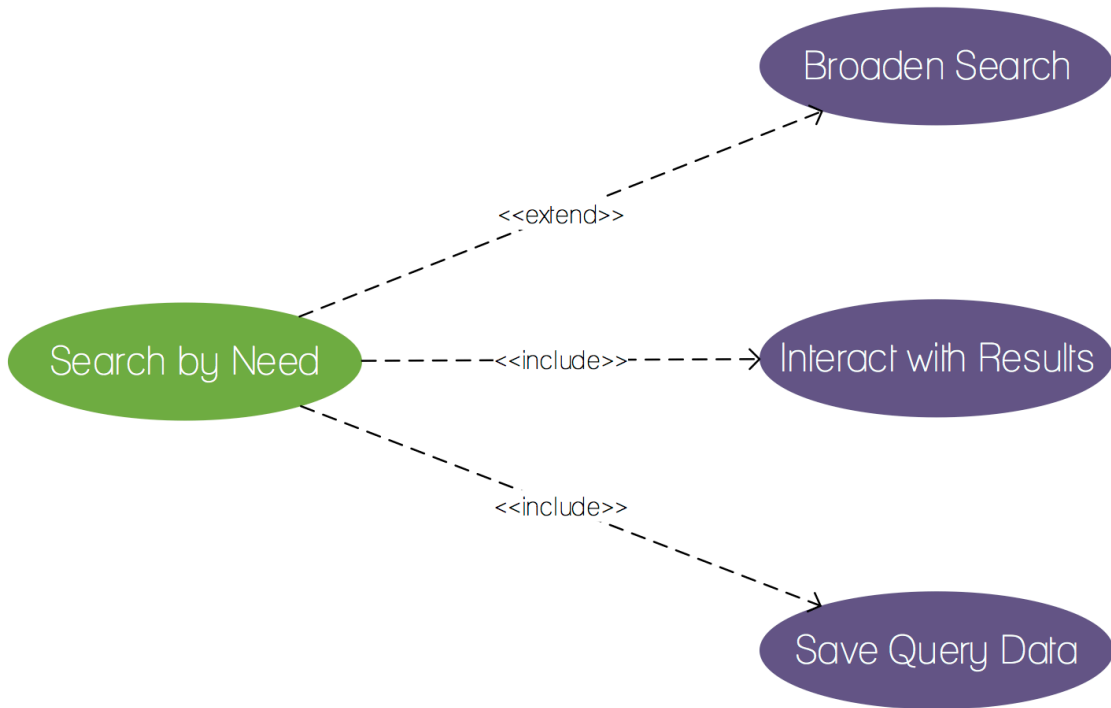
Figure 3. Search by Need
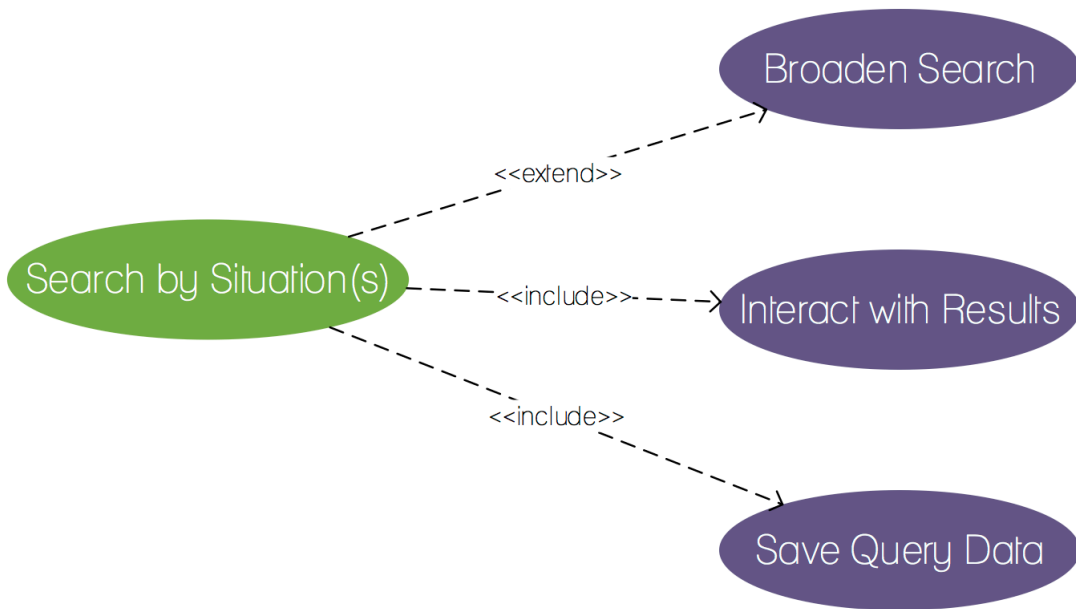
Figure 4. Search by Situation
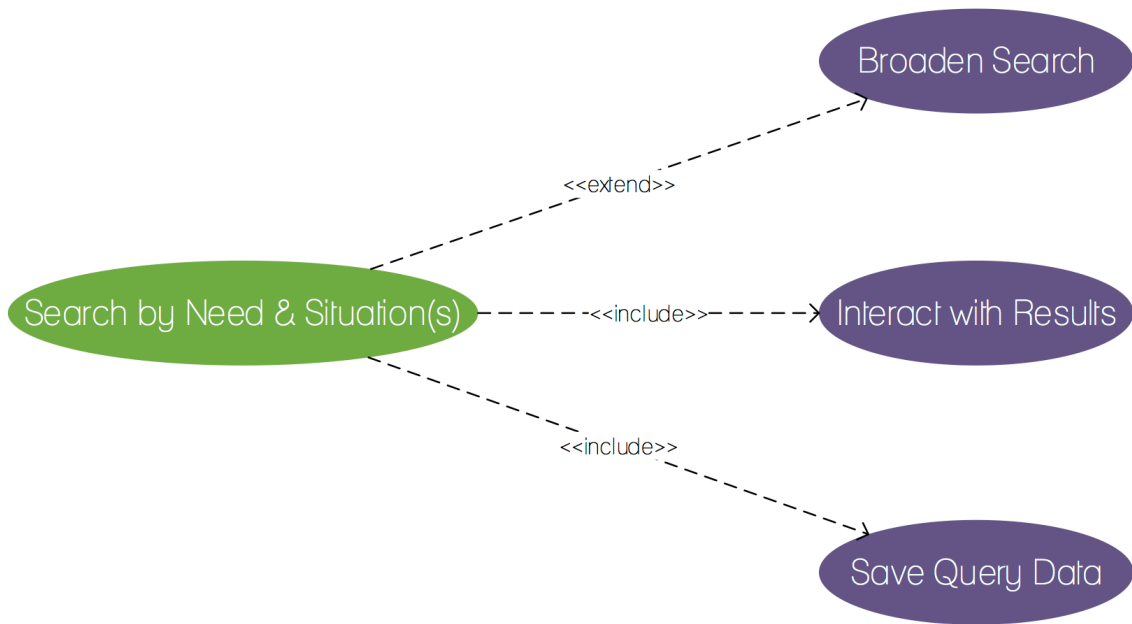
Figure 5. Search by Need & Situation
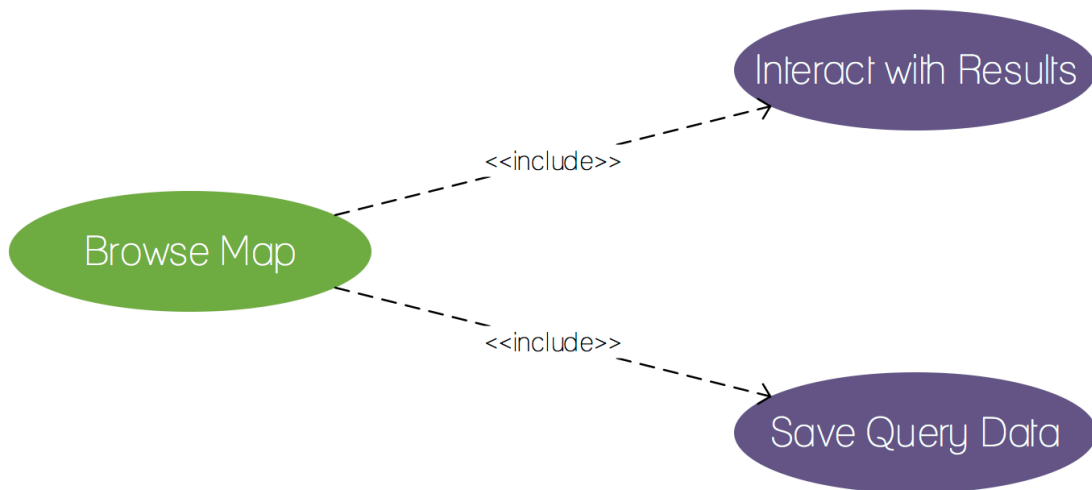
Figure 6. Browse Map
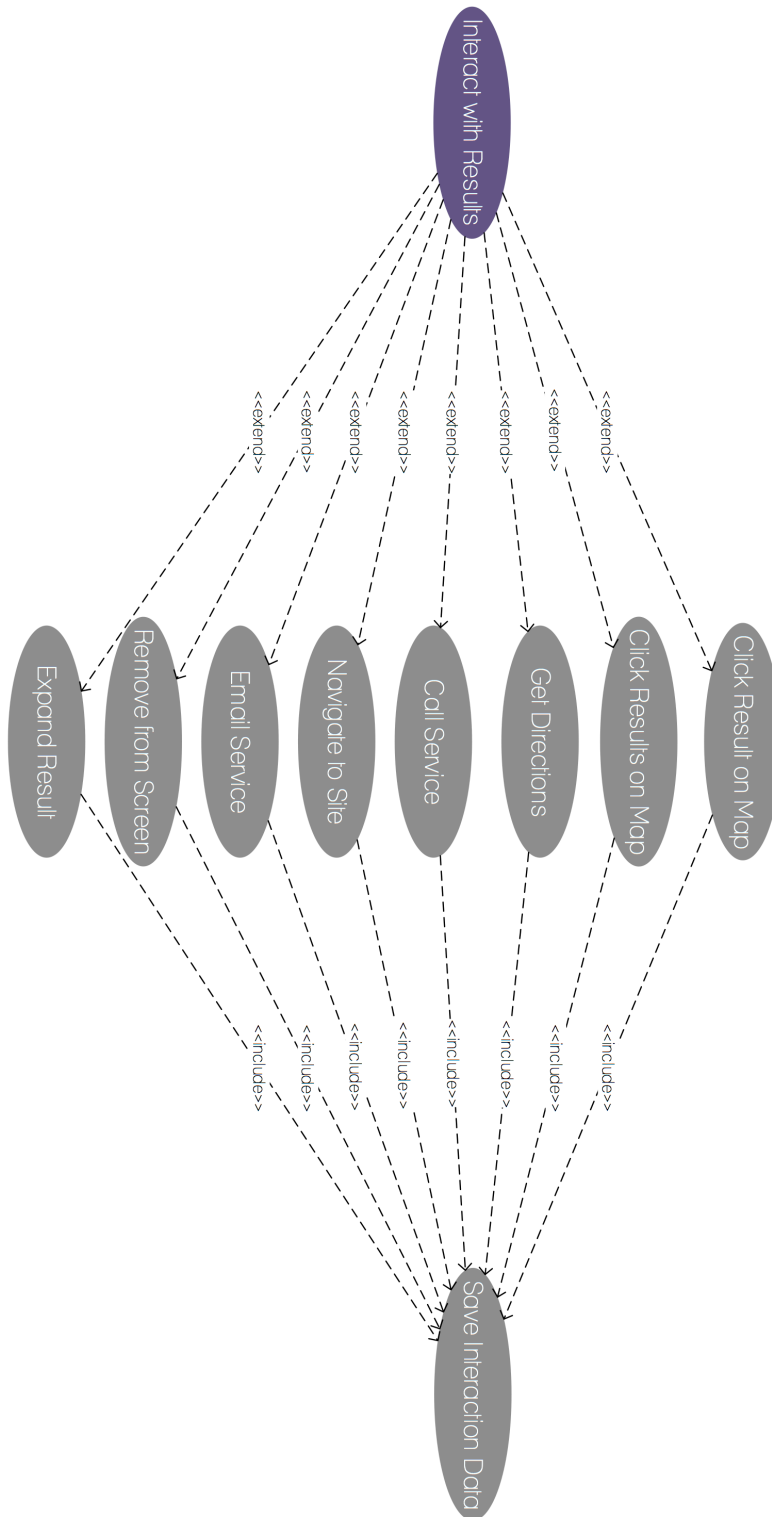
Figure 7. Interact with Results

2.3 Interaction Design Specifications

While Team Wakati will continue to hone the appearance of the Project Wakati application during the design phase of SDLC, these mockups depict the most important aspects of user interaction with the application.

Figure 8. Mobile (Portrait Orientation)
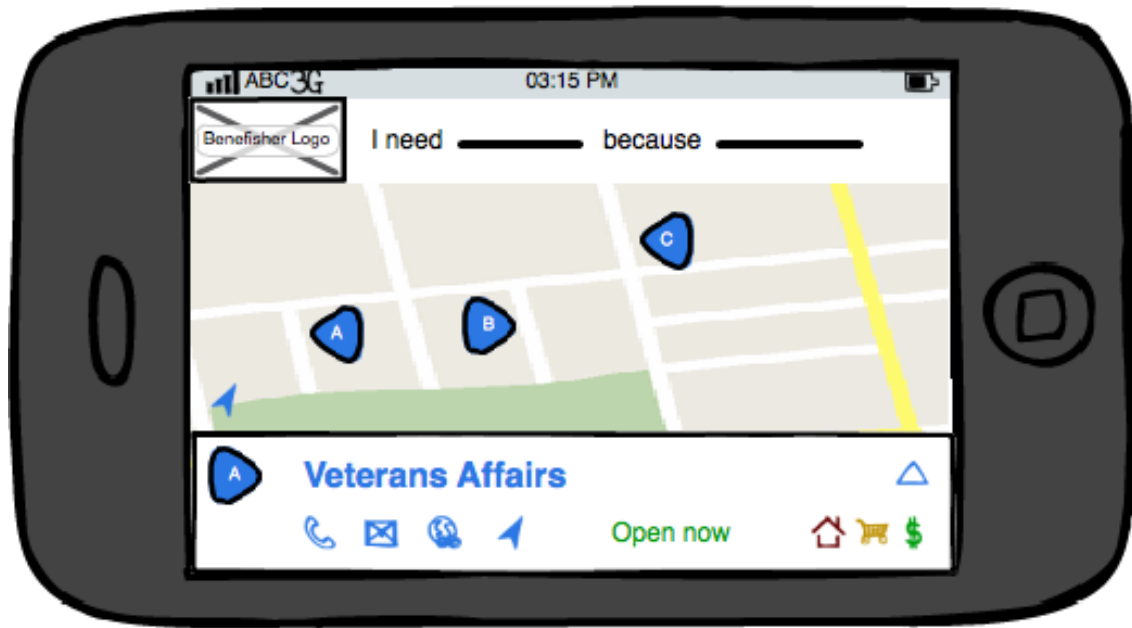
Figure 9. Mobile (Landscape Orientation)
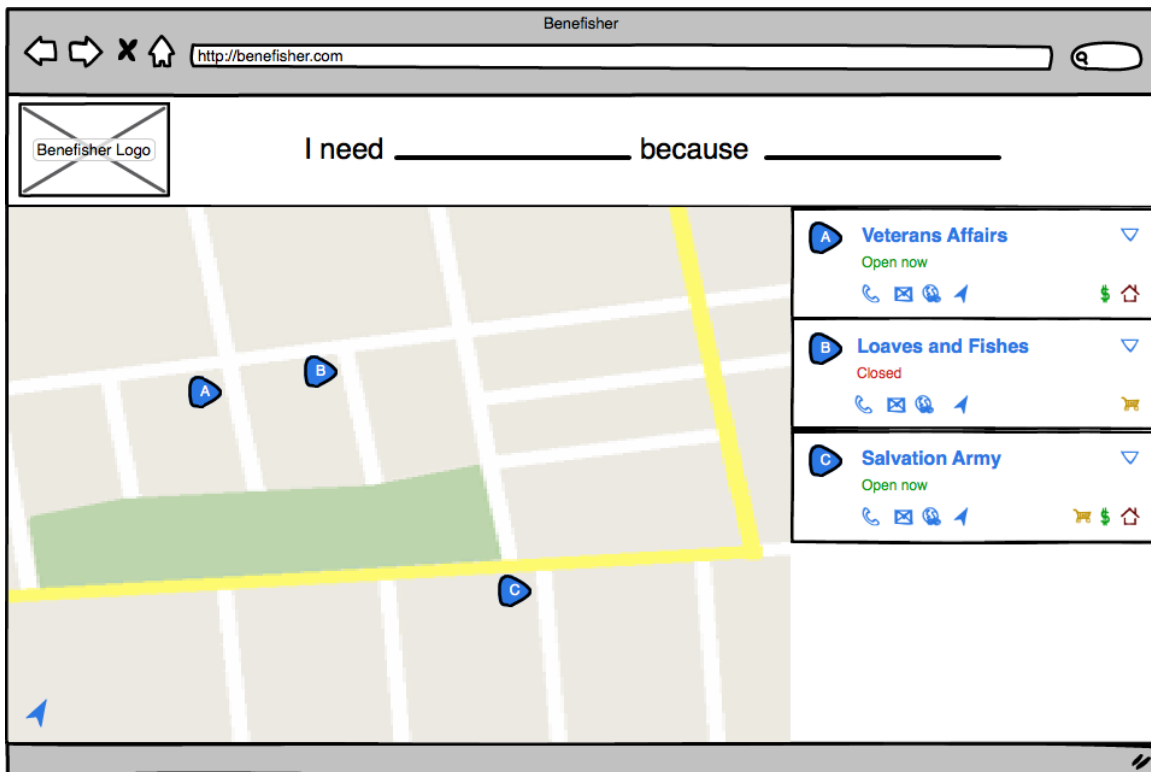
Figure 10. Desktop
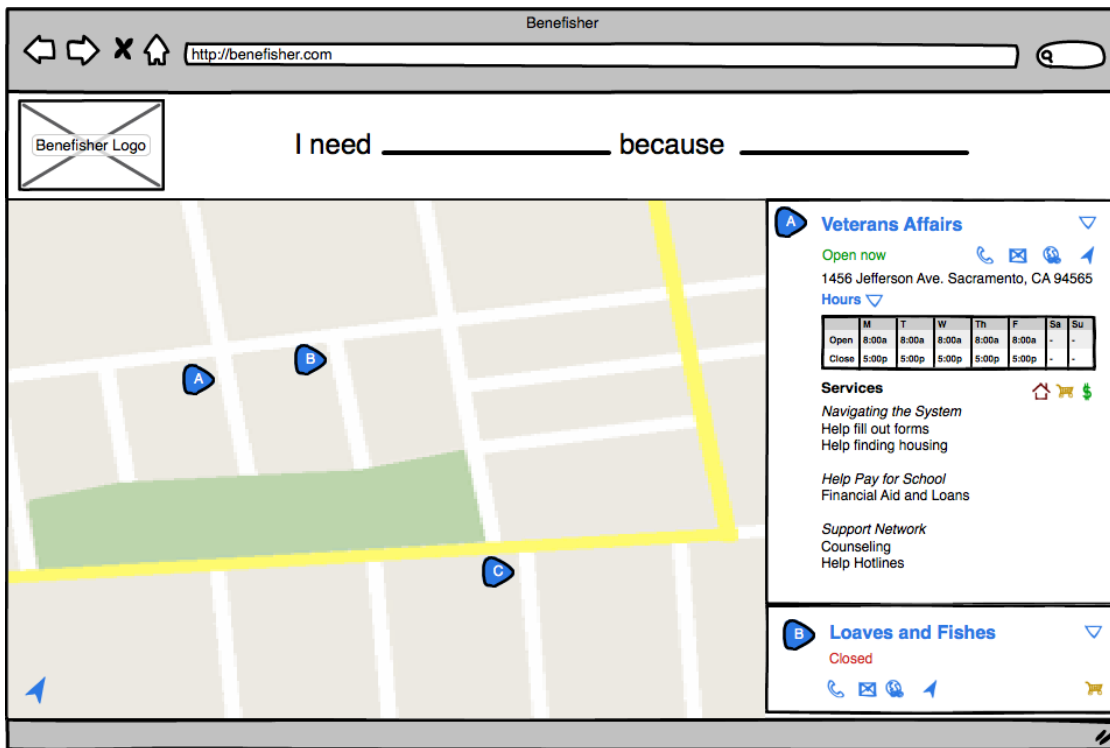
Figure 11. Desktop – Expanded Result
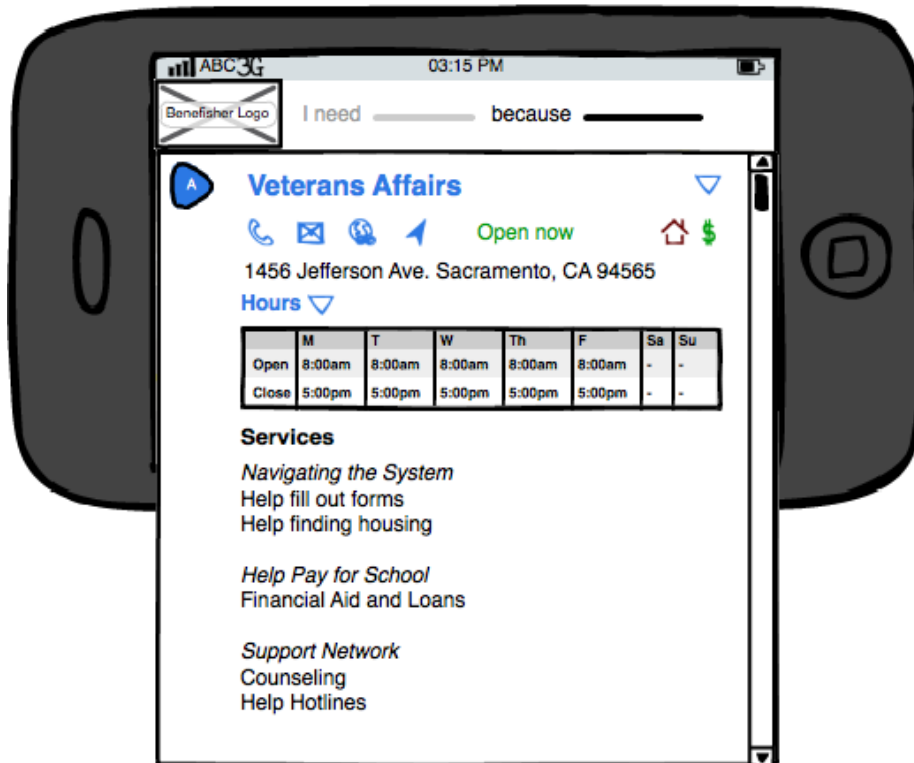
Figure 12. Mobile – Expanded Result (Landscape Orientation)

Figure 13. Mobile – Search by Situation (Landscape Orientation)

2.4 User Characteristics

Project Wakati is intended for anyone in need of public services. This includes a diverse swath of the population. OEP [2] defines 71 mostly non-exclusive "situations" to describe someone who could be helped by public services. These 71 situations are divided into 21 categories. For example, the "Income" category provides two situations: "Benefit Recipients" and "Low-Income" [2]. The OEP situations, and the nature of the public services they relate to, are important to understanding the needs of Project Wakati users. Using the OEP situations, Team Wakati has identified the following potential user constraints as being particularly relevant to this project:

- **Financial Constraints.** The OEP situations suggest that many users will be economically disadvantaged. For that reason, users are unlikely to own their own personal computers, and many may have limited technical literacy. Team Wakati imagines such users accessing the Project Wakati application from public computers (e.g. computers at the public library) or from low-cost mobile devices (e.g. prepaid Android smartphones).

- **Transportation Constraints.** A consequence of limited economic resources is that users will likely have limited access to transportation resources. Users will likely be most interested in geographically proximate services.

- **Physical Constraints.** The OEP situation categories include "Disability", which means that Project Wakati users may require screen readers or other assistive technologies to access the web.

- **Privacy Constraints.** Several of the OEP situations describe survivors of crime or abuse. These users may have serious and legitimate concerns about the privacy and security of any information they provide.

2.5 General Constraints

Constraints in this project refer to the conditions Team Wakati must work within. These limits apply to aspects of the project such as scope, design, implementation and testing priority.

1 Team Wakati consists of five members with a sixteen-week schedule for design, implementation and testing of the project. Team Wakati will be limited by total work hours in both the number and depth of features associated with the project.

2 Each team member has different levels of experience and knowledge with the tools required for the project.

3 The sponsor has offered volunteers to help evaluate the project as end users during implementation and testing. As such, scheduling volunteers and experience of individual volunteers may affect the time Team Wakati will have to effectively use data from the volunteers.

4 A large number of browsers exist currently that the project team will not be able to fully support, given the project schedule. Team Wakati must limit testing to browsers with a large user base such as recent versions of Internet Explorer, Firefox, Chrome and Safari.

5 As a web based application, the project will not function without a connection to the Internet.

6 Depth of the content provided by the project is limited by the depth of information available in the data source maintained by the sponsor organization and its partners.

7 In keeping with the open source spirit of the sponsor organization, the team will be using open source components to build the project.

8   Each function of the project has been estimated as to completion time. The depth and complexity of these functions may not be fully apparent until implementation phase.

2.6 Assumptions and Dependencies

This section lists the assumptions Team Wakati must make about the project, as well as the dependencies for the project.

1   At least once a week, the team will meet with the sponsor to validate project progress.

2   Servers managed by the sponsor organization will host the data describing public services, required by the search features of the project.

3   Data managed by the sponsor organization and its partners is assumed to be complete and accurate.

4   All languages, software, and protocols used will be compatible with both desktop and mobile devices.

5   Requests for additional features or changes in features will be submitted via the change request process described in the Project Charter [4].

6   Team Wakati will complete the implementation of the project on or before December 12, 2014.

3 SPECIFIC REQUIREMENTS

Details the functionality and qualities of Project Wakati, as requested by the sponsor and negotiated with Team Wakati. This section includes use case specifications, performance requirements, design constraints, and quality attributes.

3.1 Use Case Specifications

Use cases model the interactions of the user with the system, and describe the functionality expected from the system.

Search for Services (UC1)			
Created By	Team Wakati	Last Updated By	Jesse
Created On	4/11/14	Last Updated On	4/29/14

Search for Services (UC1)		
Priority	High	**Frequency of Use**	High
Actors	Desktop User, Mobile User	
Preconditions	1 User has accessed the application from either a mobile or desktop web browser. 2 User has chosen to give permission to access geo-location, or provided a zip code of the area that they would like to search. 3 Map is displayed with all services in the area determined by user's zip code or geo-location.	
Postconditions	1 Relevant results are displayed on map.	
Normal Course	**Desktop User, Mobile User** 1 User decides to search by need, situation, a combination of the two, or browse the map containing all results in the area.	
Alternative Course	The alternative flow of UC1 is handled within its subcases: UC2 – UC5	
Exceptions	**E1 Unable to connect to external data source** 1 User is shown a critical error, asking them to come back at a later time. 2 Error is logged, system admin is notified via email.	
Includes	N/A	
Business Rules	See Team Wakati Project Charter [4].	
DB Requirements	Retrieve services data from external source (see 2.1).	
Special Requirements	**Privacy.** No personal information (except for zip code or geo-location, with explicit user permission, will be collected).	
Assumptions	The user has JavaScript enabled.	
Notes & Issues	User must consent to either geo-location access or zip code in order to move forward in the application.	

Table 1. Search for Services (UC1)

Search by Need (UC 2)	
Created By	Team Wakati
Last Updated By	Anthony
Created On	4/11/14
Last Updated On	4/17/14
Priority	High
Frequency of Use	Medium
Actors	Desktop User, Mobile User

Search by Need (UC 2)	
Created By	Team Wakati **Last Updated By** Anthony
Created On	4/11/14 **Last Updated On** 4/17/14
Priority	High **Frequency of Use** Medium
Actors	Desktop User, Mobile User
Preconditions	1. User has accessed the application from either a mobile or desktop web browser. 2. Map is displayed indicating all services available in the user-provided area. 3. User selects from list of OEP needs [2].
Postconditions	1. Relevant results are displayed on a map.
Normal Course	**Desktop User, Mobile User** 1. The system produces a map and a list of results containing matching service information.
Alternative Course	**Desktop User, Mobile User** 1. The system is unable to find relevant results. 2. E2 is triggered.
Exceptions	**E2 No services match search criteria** 1. User is shown a map of results for similar services in the area. 2. A message is displayed, indicating that the system was unable to find any relevant results. 3. User is suggested to revise or broaden their search.
Includes	UC6, UC9, UC18
Business Rules	N/A
DB Requirements	N/A
Special Requirements	N/A
Assumptions	N/A
Notes & Issues	Design should minimize occurrences of E2.

Table 2. Search by Need (UC2)

Search by Situation (UC 3)			
Created By	Team Wakati	Last Updated By	Anthony
Created On	4/11/14	Last Updated On	4/17/14
Priority	High	Frequency of Use	Medium
Actors	Desktop User, Mobile User		
Preconditions	1 User has accessed the application from either a mobile or desktop web browser. 2 Map is displayed indicating all services available in the user-provided area. 3 User selects situation(s) from OEP list [2].		
Postconditions	1 Relevant results are displayed on a map.		
Normal Course	**Desktop User, Mobile User** 1 The system produces a map and a list of results containing matching service information.		
Alternative Course	**Desktop User, Mobile User** 1 The system is unable to find results for the specific situation, or combination of situations. 2 E2 is triggered.		
Exceptions	E2 (see UC2)		
Includes	UC7, UC9, UC18		
Business Rules	N/A		
DB Requirements	**N/A**		
Special Requirements	N/A		
Assumptions	N/A		
Notes & Issues	Design should minimize occurrences of E2.		

Table 3. Search by Situation (UC3)

Search by Need & Situation (UC 4)			
Created By	Team Wakati	Last Updated By	Anthony
Created On	4/11/14	Last Updated On	4/17/14
Priority	High	Frequency of Use	High
Actors	Desktop User, Mobile User		
Preconditions	1 User has accessed the application from either a mobile or desktop web browser. 2 Map is displayed indicating all services available in the user-provided area. 3 User selects situation(s) from OEP list [2]. 4 User selects a need from OEP list [2].		
Postconditions	1 Relevant results are displayed on a map.		
Normal Course	Desktop User, Mobile User 1 The system produces a map and a list of results containing matching service information.		
Alternative Course	Desktop User, Mobile User 1 The system is unable to find results for the specific situation, or combination of situations. 2 E2 is triggered.		
Exceptions	E2 (see UC2)		
Includes	UC6, UC7, UC8, UC9, UC18		
Business Rules	N/A		
DB Requirements	N/A		
Special Requirements	N/A		
Assumptions	N/A		
Notes & Issues	Design should minimize occurrences of E2.		

Table 4. Search for Services by Need & Situation (UC4)

Browse Map (UC5)			
Created By	Team Wakati	Last Updated By	Anthony
Created On	4/11/14	Last Updated On	4/17/14
Priority	High	Frequency of Use	High
Actors	Desktop User, Mobile User		
Preconditions	1 User has accessed the application from either a mobile or desktop web browser. 2 Map is displayed indicating all services available in the user-provided area.		
Postconditions	1 Results are updated, based on current viewport.		
Normal Course	**Desktop User, Mobile User** 1 Navigate the map using a click and drag method. 2 Based on current viewport, relevant services are displayed as markers on the map, and as a list.		
Alternative Course	**Desktop User, Mobile User** 1 User navigates to a part of the map with no services available within the viewport. 2 E3 is triggered.		
Exceptions	**E3 No services within current viewport** 1 User is shown a list of nearby results nearby, while the map is empty. 2 User is suggested to perform a search outlined in UC2, UC3, or UC4.		
Includes	UC8, UC9		
Business Rules	N/A		
DB Requirements	N/A		
Special Requirements	N/A		
Assumptions	N/A		
Notes & Issues	N/A		

Table 5. Browse Map (UC5)

Select Pre-defined Need (UC6)			
Created By	Team Wakati	Last Updated By	Anthony
Created On	4/11/14	Last Updated On	4/17/14
Priority	High	Frequency of Use	High
Actors	Desktop User, Mobile User		
Preconditions	1 User has accessed the application from either a mobile or desktop web browser. 2 Map is displayed indicating all services available in the user-provided area.		
Postconditions	1 Selected need added to search criteria.		
Normal Course	**Desktop User, Mobile User** 1 A list of OEP needs [2] is retrieved from a data store. 2 User selects one of pre-defined need. 3 Selected need displayed as search criteria.		
Alternative Course	**Desktop User, Mobile User** 1 System is unable to retrieve list of OEP needs from data store. 2 E4 is triggered.		
Exceptions	**E4 Unable to read from internal data store** 1 User is shown a critical error, asking them to come back at a later time. 2 Error is logged, system admin is notified via email.		
Includes	N/A		
Business Rules	N/A		
DB Requirements	Retrieve list of OEP needs [2].		
Special Requirements	N/A		
Assumptions	N/A		
Notes & Issues	Design should minimize occurrences of E4.		

Table 6. Select Pre-defined Need (UC6)

Select Pre-defined Situation(s) (UC7)			
Created By	Team Wakati	Last Updated By	Anthony
Created On	4/11/14	Last Updated On	4/17/14
Priority	High	Frequency of Use	High
Actors	Desktop User, Mobile User		
Preconditions	1 User has accessed the application from either a mobile or desktop web browser. 2 Map is displayed indicating all services available in the user-provided area.		
Postconditions	1 Selected situation(s) added to search criteria.		
Normal Course	**Desktop User, Mobile User** 1 A list of OEP situations [2] is retrieved from a data store. 2 User selects one or more of pre-defined situation(s). 3 Selected situation(s) displayed as search criteria.		
Alternative Course	**Desktop User, Mobile User** 1 System is unable to retrieve list of OEP situations from data store. 2 E4 is triggered.		
Exceptions	E4 (see UC6)		
Includes	N/A		
Business Rules	N/A		
DB Requirements	Retrieve list of OEP situations [2].		
Special Requirements	N/A		
Assumptions	N/A		
Notes & Issues	Design should minimize occurrences of E4.		

Table 7. Select Pre-defined Situations (UC7)

Save Interaction Data (UC8)			
Created By	Team Wakati	Last Updated By	Anthony
Created On	4/11/14	Last Updated On	4/17/14
Priority	High	Frequency of Use	High
Actors	Desktop User, Mobile User		
Preconditions	1 User has performed a search or has browsed the map, as outlined in UC2 – UC5. 2 Relevant results are shown on a map and in a sidebar (desktop) or on the bottom of the screen (mobile). 3 User has performed an interaction with the data, as outlined in UC9 and subcases.		
Postconditions	1 Interaction data is saved to a data store.		
Normal Course	**Desktop User, Mobile User** 1 Relevant interaction data (including search query and interaction type) is recorded.		
Alternative Course	**Desktop User, Mobile User** 1 System is unable to connect to database. 2 E4 is triggered.		
Exceptions	**E5 Unable to connect/write to database** 1 Error is logged, system admin is notified via email.		
Includes	N/A		
Business Rules	N/A		
DB Requirements	Write interaction data to database.		
Special Requirements	N/A		
Assumptions	N/A		
Notes & Issues	Design should minimize occurrences of E5, and occurrences of E5 should be invisible to user.		

Table 8. Save Interaction Data (UC8)

Interact With Results (UC9)			
Created By	Team Wakati	Last Updated By	Anthony
Created On	4/11/14	Last Updated On	4/24/14
Priority	High	Frequency of Use	High
Actors	Desktop User, Mobile User		
Preconditions	1 User has performed a search or has browsed the map, as outlined in UC2 – UC5. 2 Relevant results are shown on a map and in a sidebar (desktop) or on the bottom of the screen (mobile).		
Postconditions	1 Interaction is recorded (see UC8).		
Normal Course	**Desktop User, Mobile User** 1 A user interacts with a result on the page, as outlined by UC 10 – 16.		
Alternative Course	The alternative flow of UC9 is handled within its various subcases.		
Exceptions	The exceptions of UC9 are handled within its various subcases.		
Includes	N/A		
Business Rules	N/A		
DB Requirements	N/A		
Special Requirements	N/A		
Assumptions	N/A		
Notes & Issues	N/A		

Table 9. Interact with Results (UC9)

Expand Results (UC10)			
Created By	Team Wakati	Last Updated By	Anthony
Created On	4/11/14	Last Updated On	4/17/14
Priority	Medium	Frequency of Use	Medium
Actors	Desktop User, Mobile User		
Preconditions	1 User has performed a search or has browsed the map, as outlined in UC2 – UC5. 2 Relevant results are shown on a map and in a sidebar (desktop) or on the bottom of the screen (mobile). 3 User clicks the 'expand' icon in a result, indicating they would like more information.		
Postconditions	1 Result is expanded, showing more information about the clicked service. 2 Button to return result to normal size appears.		
Normal Course	**Desktop User, Mobile User** 1 When a result's 'expand' icon is clicked upon, the targeted result expands. 2 When result is expanded, the user can choose to return it to it's normal state, or interact with it further.		
Alternative Course	N/A		
Exceptions	N/A		
Includes	N/A		
Business Rules	N/A		
DB Requirements	N/A		
Special Requirements	N/A		
Assumptions	N/A		
Notes & Issues	N/A		

Table 10. Expand Results (UC10)

Get Directions (UC11)	
Created By	Team Wakati	Last Updated By	Anthony
Created On	4/11/14	Last Updated On	4/17/14
Priority	Medium	Frequency of Use	Medium
Actors	Desktop User, Mobile User
Preconditions	1 User has performed a search or has browsed the map, as outlined in UC2 – UC5. 2 Relevant results are shown on a map and in a sidebar (desktop) or on the bottom of the screen (mobile). 3 User clicks the 'directions' icon on a result, indicating that they would like directions to the service location.
Postconditions	1 User is redirected to default directions app.
Normal Course	**Desktop User, Mobile User** 1 User clicks result's 'get directions' icon. 2 User is redirected to their default directions app.
Alternative Course	**Desktop User, Mobile User** 1 User clicks result's 'get directions' icon. 2 Project Wakati is unable to redirect the user to a native directions app. 3 E6 is triggered.
Exceptions	**E6 Unable to redirect to native directions app** 1 An error (with a link to the OpenStreetMap project) is displayed to the user. 2 User client's "app not found" flow is triggered.
Includes	N/A
Business Rules	N/A
DB Requirements	N/A
Special Requirements	N/A
Assumptions	N/A
Notes & Issues	N/A

Table 11. Get Directions (UC11)

Call Service (UC12)			
Created By	Team Wakati	Last Updated By	Anthony
Created On	4/11/14	Last Updated On	4/17/14
Priority	Medium	Frequency of Use	Medium
Actors	Mobile User		
Preconditions	1 User has performed a search or has browsed the map, as outlined in UC2 – UC5. 2 Relevant results are shown on a map and in a sidebar (desktop) or on the bottom of the screen (mobile). 3 User has clicked the 'call service' icon in the sidebar, indicating that they would like to contact the service.		
Postconditions	1 The call is cancelled or the user places the call directly from their mobile device, without having to manually dial.		
Normal Course	**Mobile User** 1 When a service's 'call service' icon is clicked upon, the user is asked to confirm whether they would like to make a phone call. 2 The mobile phone's capabilities are leveraged to make the phone call directly, without manual dialing.		
Alternative Course	N/A		
Exceptions	N/A		
Includes	N/A		
Business Rules	N/A		
DB Requirements	N/A		
Special Requirements	N/A		
Assumptions	Team Wakati assumes that the user's mobile device is a smart device, capable making phone calls and receiving remote requests to do so.		
Notes & Issues	N/A		

Table 12. Call Service (UC12)

Navigate to Site (UC13)			
Created By	Team Wakati	Last Updated By	Anthony
Created On	4/11/14	Last Updated On	4/17/14
Priority	Medium	Frequency of Use	Medium
Actors	Desktop User, Mobile User		
Preconditions	1 User has performed a search or has browsed the map, as outlined in UC2 – UC5. 2 Relevant results are shown on a map and in a sidebar (desktop) or on the bottom of the screen (mobile). 3 User has clicked the 'website' icon in a result.		
Postconditions	1 The user is redirected to the website provided in the result.		
Normal Course	**Desktop User, Mobile User** 1 The user clicks the 'website' icon in a result. 2 The website is opened in a new tab/window.		
Alternative Course	N/A		
Exceptions	N/A		
Includes	N/A		
Business Rules	N/A		
DB Requirements	N/A		
Special Requirements	N/A		
Assumptions	N/A		
Notes & Issues	N/A		

Table 13. Navigate to Site (UC13)

Email Service (UC14)			
Created By	Team Wakati	Last Updated By	Anthony
Created On	4/11/14	Last Updated On	4/17/14
Priority	Low	Frequency of Use	Low
Actors	Desktop User, Mobile User		
Preconditions	1 User has performed a search or has browsed the map, as outlined in UC2 – UC5. 2 Relevant results are shown on a map and in a sidebar (desktop) or on the bottom of the screen (mobile). 3 User has clicked the 'email' icon in a result.		
Postconditions	1 The user is directed to their default email application on mobile or desktop platform.		
Normal Course	**Desktop User, Mobile User** 1 When a result's 'email' icon is clicked upon, the user's default email application opens. 2 A new message is started. 3 The recipient field is be pre-filled.		
Alternative Course	N/A		
Exceptions	N/A		
Includes	N/A		
Business Rules	N/A		
DB Requirements	N/A		
Special Requirements	N/A		
Assumptions	N/A		
Notes & Issues	N/A		

Table 14. Email Service (UC14)

Remove from Screen (UC15)			
Created By	Team Wakati	Last Updated By	Anthony
Created On	4/11/14	Last Updated On	4/17/14
Priority	Medium	Frequency of Use	Low
Actors	Desktop User, Mobile User		
Preconditions	1 User has performed a search or has browsed the map, as outlined in UC2 – UC5. 2 Relevant results are shown on a map and in a sidebar (desktop) or on the bottom of the screen (mobile). 3 User has clicked 'remove' icon in a result.		
Postconditions	1 Information about the result is removed from the list of results, and the map.		
Normal Course	**Desktop User, Mobile User** 1 User selects to remove the result from the screen. 2 Result is removed.		
Alternative Course	N/A		
Exceptions	N/A		
Includes	N/A		
Business Rules	N/A		
DB Requirements	N/A		
Special Requirements	N/A		
Assumptions	N/A		
Notes & Issues	N/A		

Table 15. Remove from Screen (UC15)

Click Result on Map (UC16)			
Created By	Team Wakati	Last Updated By	Anthony
Created On	4/11/14	Last Updated On	4/17/14
Priority	High	**Frequency of Use**	High
Actors	Desktop User, Mobile User		
Preconditions	1 User has performed a search or has browsed the map, as outlined in UC 2 – UC 5. 2 Relevant results are shown on a map and in a sidebar (desktop) or on the bottom of the screen (mobile). 3 User has clicked a 'result' icon on the map.		
Postconditions	1 The result will expand and UC10 will take effect.		
Normal Course	**Desktop User** 1 User clicks a 'result' icon on the map. 2 Result is highlighted on the results list. **Mobile User** 1 User clicks a 'result' icon on the map. 2 Related result appears at the bottom of the screen.		
Alternative Course	N/A		
Exceptions	N/A		
Includes	N/A		
Business Rules	N/A		
DB Requirements	N/A		
Special Requirements	N/A		
Notes & Issues	N/A		

Table 16. Click Result on Map (UC16)

Broaden Search (UC17)			
Created By	Team Wakati	Last Updated By	Anthony
Created On	4/11/14	Last Updated On	4/17/14
Priortiy	High	Frequency of Use	Medium
Actors	Desktop User, Mobile User		
Preconditions	1 User has performed a search or has browsed the map, as outlined in UC2 – UC5. 2 User modifies the search by removing tags from the situation or the need.		
Postconditions	1 New results populate the map and the results list.		
Normal Course	**Desktop, Mobile User** 1 Results are shown on screen. 2 User modifies search criteria. 3 Originating use case is retriggered.		
Alternative Course	**Desktop, Mobile User** 1 No results are given. 2 User is suggested a search similar to current search. 3 User accepts the suggestion. 4 Results are displayed.		
Exceptions	N/A		
Includes	N/A		
Business Rules	N/A		
DB Requirements	N/A		
Special Requirements	N/A		
Assumptions	N/A		
Notes & Issues	N/A		

Table 17. Broaden Search (UC17)

Save Query Data (UC18)			
Created By	Team Wakati	Last Updated By	Anthony
Created On	4/11/14	Last Updated On	4/17/14
Priority	High	Frequency of Use	High
Actors	Desktop User, Mobile User.		
Preconditions	1 User has performed a search or has browsed the map, as outlined in UC2 – UC5.		
Postconditions	1 Query data is saved to a data store, where it will be used to improve search accuracy.		
Normal Course	**Desktop User, Mobile User** 1 Relevant query data (including: zip code, need, and/or situation searched) is saved to a database, where it will be used to improve search accuracy.		
Alternative Course	N/A		
Exceptions	E5 (see UC8)		
Includes	N/A		
Business Rules	N/A		
DB Requirements	Write search query data to database.		
Special Requirements	N/A		
Assumptions	N/A		
Notes & Issues	Design should minimize occurrences of E5, and o occurrences of E4 should be invisible to user.		

Table 18. Save Query Data (UC18)

3.2 Informational Model

Project Wakati's search features will access a data source owned and managed by the sponsor. The design of that data source is ongoing, but its structure will be derived from the OpenReferral data standard [7]. The data managed by the Project Wakati application is limited to usage data recording users' queries and interactions with search results, which will be used to improve the accuracy of search results.

Please see Appendix A for the detailed data dictionary, Appendix B for a simplified ERD of the data entities managed by Project Wakati (and their interaction with data managed by the sponsor organization), and Appendix C for the OpenReferral data model.

3.3 Performance Requirements

As a public-facing web application, performance is crucial to the success of Project Wakati. Nielsen [8] identifies long response times as a primary driver of user dissatisfaction with web applications. To avoid excessive response times, server architecture must be capable of meeting performance requirements. Server performance requirements are a function of the volume of HTTP traffic to the server per user, and the volume of users.

HTTP Requests. The majority of website response time (and of server workload) is generally due to HTTP request volume. For an initial page load, the HTTP requests to, and responses from, the server retrieve the following resources (in approximate ascending order of file size):

1   **Text-based content.** HTML (and JSON or XML) content incurs some additional delay while the server executes the scripts required to generate the content, which may include additional requests to an external data source. This estimation ignores these implementation- and page-specific delays. For the purpose of estimating the performance requirements on the server, it is sufficient to say that such content will be delivered in a single request/response cycle.
    HTTP Requests per page: 1

2   **JavaScript files.** Project Wakati will minimize the overhead of JavaScript and CSS file requests by combining and minimizing those files into a single file of each type.
    HTTP Requests per page: 1

3 **CSS files.** See "JavaScript files" above.
   HTTP Requests per page: 1

4 **Font files.** Font files, used to display non-native fonts on the web, can dwarf JavaScript and CSS file sizes, and multiple font formats may be needed to account for variations in browser support. To minimize the performance penalty associated with serving web fonts, Project Wakati will use fonts hosted by Google. Google's infrastructure is far superior to that available to Project Wakati, so fonts will travel "over the wire" as fast as possible. It's also possible that users will have a given Google-provided font cached locally from visits to other sites. If nothing else, using Google's font delivery system reduces the load on the Project Wakati server.
   HTTP Requests per page: 0

5 **Media files.** The only traditional media files served via the Project Wakati server will be one small image for the project's logo, and the site's favicon. As these will be very small files that will be cached after the first page load, they are not counted against the application's requests per page.

   Of greater concern is Project Wakati's heavy reliance on geographic maps. Such maps use "tiles" to display data to users in a meaningful way (i.e. as maps). Retrieving these tiles will incur a significant and unavoidable performance penalty, but because the tiles will be provided by a third-party service, they will not increase the workload of the Project Wakati server.
   HTTP Requests per page: 0

User Traffic. The user load for the Project Wakati application is estimated at 14,914 monthly visitors. This estimate is derived via comparison to the traffic seen by the most similar existing product, the 211 Sacramento website (http://211sacramento.org). Monthly visitors for the 211 Sacramento website were calculated using the site's Alexa global ranking [9] (12,349,303), and the formula described in [10]:

$$monthly\ visitors = 104{,}943{,}144{,}672 \times (Alexa\ rank)^{-1.008}$$

This yields a figure of 7,457 monthly visitors to the 211 Sacramento website. Since the goal of this project is to improve upon existing solutions such as 211 Sacramento, that figure of 7,457 was doubled (i.e. scaled by a 'traffic multiplier' of two) to produce the estimated monthly visitor figure of 14,914. Sources of

imprecision in this estimate include any imprecision in the Alexa ranking system, any errors or imprecision in the formula described above, and the mostly arbitrary choice of two as a traffic multiplier.

This workload, distributed evenly across an entire month is less than 0.35 users per minute. Given the requests per page estimated in the "HTTP Requests" section above, the Project Wakati server can expect just over one HTTP request per minute (1.05 requests per minute). However, the Project Wakati application will also feature AJAX interactions, which increase the number of HTTP requests to the server. Given the search-based nature of the application (as opposed to a real-time communication application, for example), the number of AJAX requests is estimated to be relatively low, in the range of one to ten requests per user per minute. The distributed volume of HTTP requests to the server is thus estimated to range from 2.10 to 10.5 requests per minute.

Clearly, web traffic is not evenly distributed over time, so variations in traffic must be anticipated. Such variations can result from normal usage patterns (e.g. traffic is likely to be higher at midday than at midnight), or from external forces (e.g. the application is featured on a local news program). Given the figures estimated above, a traffic spike of 100 times the evenly distributed load would result in less than twenty requests per second (17.5 requests per second).

Implications. At these relatively low traffic volumes, it is the judgment of Team Wakati that extensive performance optimization would likely be premature. Team Wakati will use practices, such as the combining and minimizing of JavaScript and CSS mentioned above, in coding to reduce the number of HTTP requests made to the server, and to avoid slow response times for client-side interactions. Beyond these basic, Team Wakati recommends that the sponsor organization monitor server usage patterns and consider server optimization once performance or application availability is affected.

3.4 Design Constraints

Describes constraints on the system due to standards and hardware limitations.

- The product will be web based, so users will need access to the Internet to use it.

- The system will be designed in such a way that other organizations can modify the system for their own use.

- Mobile users will be supported, and presented with an experience tailored for mobile devices.

3.5 Quality Attributes.

1 **Reliability.** Team Wakati aims to produce a system that can operate continually for two years. Team Wakati expects that improvements or overhauls will be made to the project after that time. Team Wakati will be creating an interface for the user, which depends on an external data source managed by the sponsor organization. Project Wakati will only be as reliable as that data source.

2 **Maintainability.** Team Wakati intends to create an understandable and maintainable system that can be managed by the sponsor organization after Team Wakati's obligation ends, as outlined in the Project Charter [4].

- **Integrity of Data.** The sponsor organization and its partners are responsible for maintaining up-to-date information to avoid 'stale' data.

- **Additions/Changes to Source Code.** After Team Wakati's obligation to this system ends, responsibility for maintenance of, and improvements to, source code are the responsibility of the sponsor, as outlined by the Project Charter [4]. The system will be designed to make such maintenance and improvements easy (see "Program Quality Attributes" below).

3 **Program Quality Attributes.** The following requirements concern the quality of code to be created during the implementation phase.

- **Readability.** The readability of the software's source code has a direct impact on how easily Team Wakati and future developers can maintain and improve the system. To ensure readable and maintainable code is developed Team Wakati will use well-supported programming languages and frameworks, widely known design patterns, and clearly defined coding standards

- **Coding Standards.** In order to maintain quality code, Team Wakati will adhere to a coding standard, which adherence will be enforced during routine code reviews throughout the SDLC. For PHP code, Team Wakati will follow the PHP Framework Interop Group's PSR-2 standard

[11]. For JavaScript code, Team Wakati will use the jQuery Foundation's coding style [12].

- **Code Comments.** In addition to external documentation to aid future maintenances of the software, the developers will also maintain readability of code by writing comments within the source code during development. Comments will be included at the beginning of each routine and class, indicating the routine or class's purpose, and any applicable limitations, assumptions, or "gotchas".

4 **Security.** The following details the information the user will have to provide in order to use the system, as well as methods to prevent corruption of application data.

- The system is aimed towards helping members of the public locate a particular service. As such, the system will need to know the user's location, but the user may not want their exact location known. The system will allow a user to search services by zip code instead (see "Privacy Constraints" in section 2.4).

- The system will not need any specific information about the user other than a description of their situation, their current need, and their location or zip code.

- Identifying user data will not be saved, however, aggregate data will be collected to monitor how the system is being used. This data will not be sufficient to identify any specific user. This data will be used to improve search accuracy.

- The system will protect against SQL Injection by filtering input data before queries are executed to prevent corruption or loss of application data.

5 **Transferability and Conversion.** Project Wakati is dependent on the LAMP stack, of which certain components are fungible.

- **Software.** The Apache HTTP server may be substituted for another HTTP server, such as Nginx. The application will be database agnostic (among the solutions listed in [13]), so the MySQL component of the stack can be easily replaced with another relational database solution.

The remaining components in the stack (Linux and PHP) are fixed dependencies of Project Wakati.

- **Hardware.** The application must be deployed to a server architecture that supports the LAMP stack.

6 **Operational Quality Attributes.** Here we refer to "operational quality attributes" as those attributes related to "usability" of the software:

- Ergonomic concerns are especially important to mobile applications. Most mobile users have experienced the discomfort of contorting one's thumb to interact with a GUI component in the lower right or left corner of the screen. Project Wakati's map interface will encourage mobile users to use their pointer fingers to select results, instead of their thumbs.

- **Ergonomics.** Project Wakati will be accessible via large screens, (e.g. desktops, laptops and large tablets), and via small screens, (e.g. smartphones and small tablets). Project Wakati will identify users' screen sizes in order to appropriately tailor the user experience for a given device.

- **Multilanguage Support.** Project Wakati will use the Google Translate plugin to provide translation of text strings on the site into over sixty languages without increasing development costs. It should be noted that Google Translate is not always accurate in its translations, but it should suffice until a better alternative is available.

- **Help System.** The goal of the design of the Project Wakati application is to produce an interface that is intuitive enough that even inexperienced users will be able to find the services they're seeking without needing any sort of explicit help system.

7 **Operations.** This section describes the operational states of Project Wakati.

- The system will require a period of initial configuration, where static data, such as the list of OEP terms, is entered into the database.

- As a web application, Project Wakati will be continually available via a web server.

- Team Wakati recommends that the sponsor organization employ a continuous delivery strategy to avoid maintenance-related system downtime.

8 **Site Adaptations.** The following requirements are necessary for the software to properly.

A Web Server running a typical LAMP stack configuration is required. The following lists specific versions of each component to be used in developing the software. Versions and variations of Linux that support the remaining components of the stack are highly fungible, so a specific version is not included here.

Apache HTTP Server: Version 2.4.9
MySQL Community Server: Version 5.6.17
PHP: Version 5.4.27

4 DOCUMENT APPROVAL PAGE

This section contains the list of the key signatures necessary to approve the SRS, thereby agreeing to the scope and content of the requirements specified within the document. Any future changes in this baseline specification will be made through the project's defined change process.

Name	Signature	Date
Adrian Chambers		
Anthony Cristiano		
Daniel Green		
James Doan		
Jesse Rosato		
Sponsor Representative		
Meiliu Lu (Advisor)		

APPENDIX A

1 Data Dictionary: Data Elements

category			
Type	Varchar(255)	Set By	Initial configuration
Data Structure	OEP Term		
Description	A category for an OEP term, see [2] for a list of possible values.		
Uses	1 Divide list of available needs and situations for user search. 2 Describes services or user situations.		
Validation	N/A		

Table B.1 Data Element "category" (OEP Term)

classification			
Type	Enum	Set By	Initial configuration
Data Structure	OEP Term		
Description	The classification of OEP term (currently limited to "need" and "situation") [2].		
Uses	1 Determine whether a term describes a need or a situation.		
Validation	N/A		

Table B.2 Data Element "classification" (OEP Term)

date_time			
Type	DateTime	Set By	Database default
Data Structure	Search Result Interaction		
Description	The date and time a user interacted with a search result.		
Uses	1 Used to track interactions over time.		
Validation	N/A		

Table B.3 Data Element "date_time" (Search Result Interaction)

date_time			
Type	DateTime	Set By	Database default
Data Structure	Search Query		
Description	The date and time a user search query was executed..		
Uses	1　Used to track search queries over time.		
Validation	N/A		

Table B.4 Data Element "date_time" (Search Query)

description			
Type	Varchar(255)	Set By	Initial configuration
Data Structure	Search Result Interaction Target		
Description	A description of the target (e.g. "call", or "get directions") of a user's interaction with a search result.		
Uses	1　Used to track types of interactions with search results.		
Validation	N/A		

Table B.5 Data Element "description"
(Search Result Interaction Target)

oep_term_id			
Type	Integer	Set By	Database default
Data Structure	OEP Term		
Description	Unique identifier for an OEP term.		
Uses	1　Primary key when retrieving OEP terms.		
Validation	N/A		

Table B.6 Data Element "oep_term_id" (OEP Term)

oep_term_oep_term_id			
Type	Integer	Set By	Initial configuration
Data Structure	Search Query Has OEP Term		
Description	Unique identifier of an OEP term related to a search query.		
Uses	1 Foreign key when retrieving an OEP term from a search query to OEP term relationship.		
Validation	N/A		

Table B.7 Data Element "oep_term_oep_term_id"
(Search Query Has OEP Term)

oep_term_oep_term_id			
Type	Integer	Set By	Initial configuration
Data Structure	OEP Term Applies to Service		
Description	Unique identifier of an OEP term related to a service.		
Uses	1 Foreign key when retrieving an OEP term from a service to OEP term relationship.		
Validation	N/A		

Table B.8 Data Element "oep_term_oep_term_id"
(OEP Term Applies to Service)

phrasing			
Type	Varchar(255)	Set By	Initial configuration
Data Structure	OEP Term		
Description	A non-offensive first-person phrasing for applying a term to an individual (e.g. "I have a" for the OEP term "mental disability").		
Uses	1 Used to avoid insensitive default phrasings (e.g. "I am mentally disabled.").		
Validation	N/A		

Table B.9 Data Element "phrasing" (OEP Term)

search_query_has_oep_term_id			
Type	Integer	Set By	Initial configuration
Data Structure	Search Query Has OEP Term		
Description	Unique identifier for a search query to OEP term relationship.		
Uses	1　Primary key when retrieving search query to OEP term relationships.		
Validation	N/A		

Table B.10 Data Element "search_query_has_oep_term_id"
(Search Query Has OEP Term)

search_query_id			
Type	Integer	Set By	Database default
Data Structure	Search Query		
Description	Unique identifier for a user search query.		
Uses	1　Primary key when retrieving search queries.		
Validation	N/A		

Table B.11 Data Element "search_query_id" (Search Query)

search_query_search_query_id			
Type	Integer	Set By	Initial configuration
Data Structure	Search Query Has OEP Term		
Description	Unique identifier of a search query related to a search query to OEP term relationship.		
Uses	1　Foreign key for retrieving a search query from a search query to OEP term relationship.		
Validation	N/A		

Table B.12 Data Element "search_query_search_query_id"
(Search Query Has OEP Term)

search_query_search_query_id			
Type	Integer	Set By	Initial configuration
Data Structure	Search Result		
Description	Unique identifier for a search query related to a search result.		
Uses	1 Foreign key for retrieving a search query from a search result.		
Validation	N/A		

Table B.13 Data Element "search_query_search_query_id" (Search Result)

search_result_id			
Type	Integer	Set By	Database default
Data Structure	Search Result		
Description	Unique identifier for a search result.		
Uses	1 Primary key when retrieving search results.		
Validation	N/A		

Table B.14 Data Element "search_result_id" (Search Result)

search_result_search_result_id			
Type	Integer	Set By	Initial configuration
Data Structure	Search Result Interaction		
Description	Unique identifier for a search result related to a search result interaction.		
Uses	1 Foreign key when retrieving a search result from a search result search interaction.		
Validation	N/A		

Table B.15 Data Element "search_result_search_result_id"
(Search Result Interaction)

search_result_interaction_id			
Type	Integer	Set By	Database default
Data Structure	Search Result Interaction		
Description	Unique identifier for a user interaction with a search result.		
Uses	1 Primary key when retrieving a search result interaction.		
Validation	N/A		

Table B.16 Data Element "search_result_interaction_id"
(Search Result Interaction)

search_result_interaction_target_id			
Type	Integer	Set By	Database default
Data Structure	Search Result Interaction Target		
Description	Unique identifier for the target of a user's interaction with a search result.		
Uses	1 Primary key when retrieving a search result interaction target.		
Validation	N/A		

Table B.17 Data Element "search_result_interaction_target_id" (Search Result Interaction Target)

search_result_interaction_target_search_result_interaction_target_id			
Type	Integer	Set By	Database default
Data Structure	Search Result Interaction Target		
Description	Unique identifier for the target of a user's interaction with a search result.		
Uses	1 Foreign key when retrieving a search result interaction target from a search result interaction.		
Validation	N/A		

Table B.18 Data Element
"search_result_interaction_target_search_result_interaction_target _id"
(Search Result Interaction Target)

service_service_id			
Type	Integer	Set By	Application calculation
Data Structure	Search Result		
Description	Unique identifier for a service related to a search result.		
Uses	1 Foreign key when retrieving a service from a search result.		
Validation	N/A		

Table B.19 Data Element "service_service_id" (Search Result)

term			
Type	Varchar(255)	Set By	Initial configuration
Data Structure	OEP Term		
Description	An OEP term, see [2] for a list of possible values.		
Uses	1 Item in list of preselected needs and situations for user search. 2 Describes services or user situations.		
Validation	N/A		

Table B.20 Data Element "term" (OEP Term)

user_postal_code			
Type	Varchar(10)	Set By	User input
Data Structure	Search Query		
Description	The postal code associated with a user search query.		
Uses	1 Used to track queries by geographic location.		
Validation	Must be a valid postal code.		

Table B.21 Data Element "user_postal_code" (Search Query)

2 Data Dictionary: Data Structures

OEP Term			
Data Table	oep_term		
Uses	1 Provide predefined user needs and situations to simplify search. 2 Provide index for service, location or organization.		
Contained By			
Search Query Has OEP Term			
OEP Term Applies to Service			
Contains Structures		**Multiplicity**	**Optional**
N/A		-	-
Contains Elements		**Multiplicity**	**Optional**
oep_term_id		1	No
category		1	No
classification		1	No
phrasing		1	No
term		1	No

Table B.22 OEP Term

OEP Term Applies to Service		
Data Table	oep_term_applies_to_service	
Uses	1 Associates an OEP term with the services described by that term and vice-versa.	
Contained By		
N/A		
Contains Structures	Multiplicity	Optional
OEP Term		No
Service (external)	*	No
Contains Elements	Multiplicity	Optional
oep_term_applies_to_service_id	1	No
oep_term_oep_term_id	1	No
service_service_id	1	No

Table B.23 OEP Term Applies to Service

Search Query		
Data Table	search_query	
Uses	1 Used to track search queries to improve search accuracy.	
Contained By		
Search Query Has OEP Term		
Search Result		
Contains Structures	Multiplicity	Optional
N/A	-	-
Contains Elements	Multiplicity	Optional
search_query_id	1	No
date_time	1	No
user_postal_code	1	No

Table B.24 Search Query

Search Query Has OEP Term		
Data Table	search_query_has_oep_term	
Uses	1 Associates a search query to the OEP terms it contains.	
Contained By		
N/A		
Contains Structures	Multiplicity	Optional
OEP Term	1	No
Search Query	1	No
Contains Elements	Multiplicity	Optional
search_query_has_oep_term_id	1	No
<u>oep_term_oep_term_id</u>	1	No
search_query_search_query_id	1	No

Table B.25 Search Query Has OEP Term

Search Result		
Data Table	search_result	
Uses	1 Track search results generated by search queries to improve search accuracy.	
Contained By		
Search Result Interaction		
Contains Structures	Multiplicity	Optional
Search Query	1	No
Service (external)	1	Yes
Contains Elements	Multiplicity	Optional
search_result_id	1	No
<u>search_query_search_query_id</u>	1	No
service_service_id	1	Yes

Table B.26 Search Result

Search Result Interaction		
Data Table	search_result_interaction	
Uses	1	Track user interactions with search results to improve search accuracy.
Contained By		
N/A		
Contains Structures	**Multiplicity**	**Optional**
Search Result	1	No
Search Result Interaction Target	1	No
Contains Elements	**Multiplicity**	**Optional**
search_result_interaction_id	1	No
search_result_search_result_id	1	No
search_result_interaction_target_search_result_interaction_target_id	1	No

Table B.27 Search Result Interaction

Search Result Interaction Target		
Data Table	search_result_interaction	
Uses	2	Track user interactions with search results to improve search accuracy.
Contained By		
Search Result Interaction		
Contains Structures	**Multiplicity**	**Optional**
N/A	-	-
Contains Elements	**Multiplicity**	**Optional**
search_result_interaction_target_id	1	No
description	1	No

Table B.28 Search Result Interaction Target

3 Data Dictionary: Data Tables

oep_term		
Uses	1 Provide index for service, location or organization. 2 Provide predefined user needs and situations to simplify search.	
Records	One for each OEP term (~200).	
Growth	Extremely low (only when new terms are added to the OEP taxonomy).	
Record Size	770 Bytes	
Contains Elements	**Type**	**Size**
<u>oep_term_id</u>	Integer	4B
category	Varchar	255B
classification	Enum	1B
phrasing	Varchar	255B
term	Varchar	255B

Table B.29 Data Table "oep_term"

oep_term_applies_to_service		
Uses	1 Associates an OEP term with the services described by that term and vice-versa.	
Records	One for each term/service combination (~200 x number of services).	
Growth	Low (only when new services are added).	
Record Size	12 Bytes	
Contains Elements	**Type**	**Size**
<u>oep_term_applies_to_service_id</u>	Integer	4B
<u>oep_term_oep_term_id</u>	Integer	4B
<u>service_service_id</u>	Integer	4B

Table B.30 Data Table "oep_term_applies_to_service"

search_query		
Uses	1 Used to track search queries to improve search accuracy.	
Records	One for each search query.	
Growth	Extremely high (one per application use).	
Record Size	22 Bytes	
Contains Elements	Type	Size
search_query_id	Integer	4B
date_time	DateTime	8B
user_postal_code	Varchar	10B

Table B.31 Data Table "search_query"

search_query_has_oep_term		
Uses	1 Associates a search query to the OEP terms the query contains.	
Records	One for each OEP term in each search query.	
Growth	Extremely high (multiple per application use).	
Record Size	12 Bytes	
Contains Elements	Type	Size
search_query_has_oep_term_id	Integer	4B
search_query_search_query_id	Integer	4B
oep_term_oep_term_id	Integer	4B

Table B.32 Data Table "search_query_has_oep_term"

search_result		
Uses	1	Track search results related to search queries to improve search accuracy.
Records	Multiple for every search query.	
Growth	Extremely high (multiple per application use).	
Record Size	12 Bytes	
Contains Elements	Type	Size
search_result_id	Integer	4B
search_query_search_query_id	Integer	4B
service_service_id	Integer	4B

Table B.33 Data Table "search_result"

search_result_interaction		
Uses	1	Track user interactions with search results to improve search accuracy.
Records	One for each interaction per search result.	
Growth	Extremely high (multiple per application use).	
Record Size	20 Bytes	
Contains Elements	Type	Size
search_result_interaction_id	Integer	4B
date_time	DateTime	8B
search_result_id	Integer	4B
search_result_search_result_id	Integer	4B

Table B.34 Data Table "search_result_interaction"

search_result_interaction_target		
Uses	2 Track types of user interactions with search results to improve search accuracy.		
Records	One for each search result interaction type (currently 6).		
Growth	Extremely low (only when new interaction types are added).		
Record Size	259 Bytes		
Contains Elements	Type	Size
search_result_interaction_target_id	Integer	4B
description	Varchar	255B

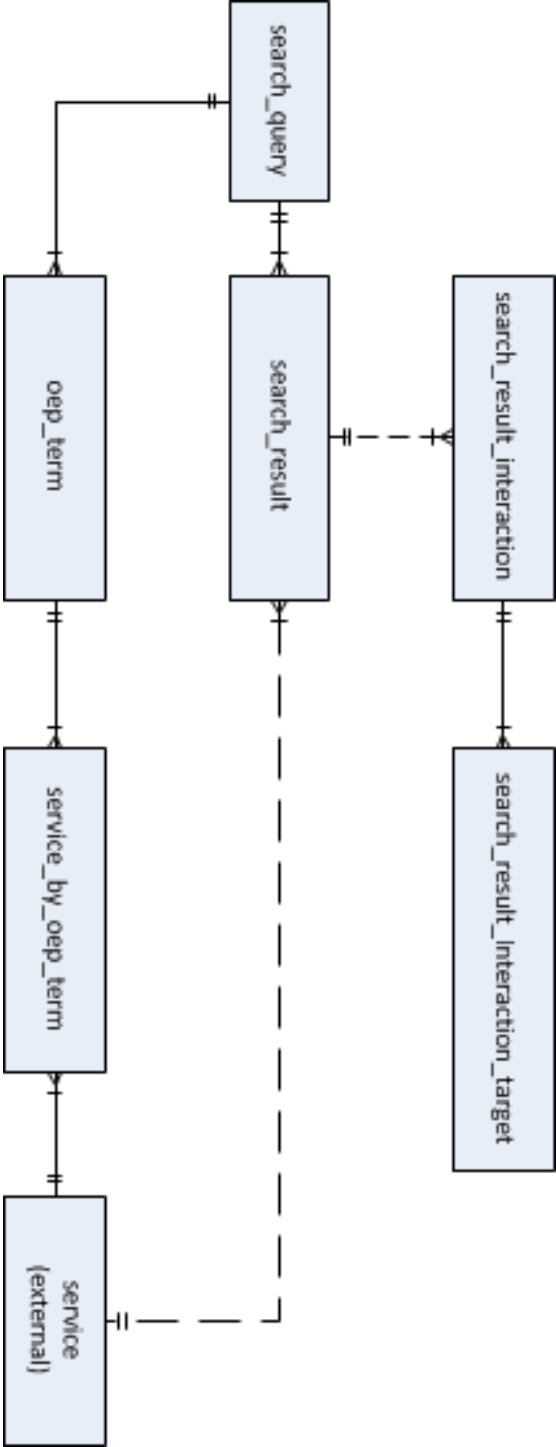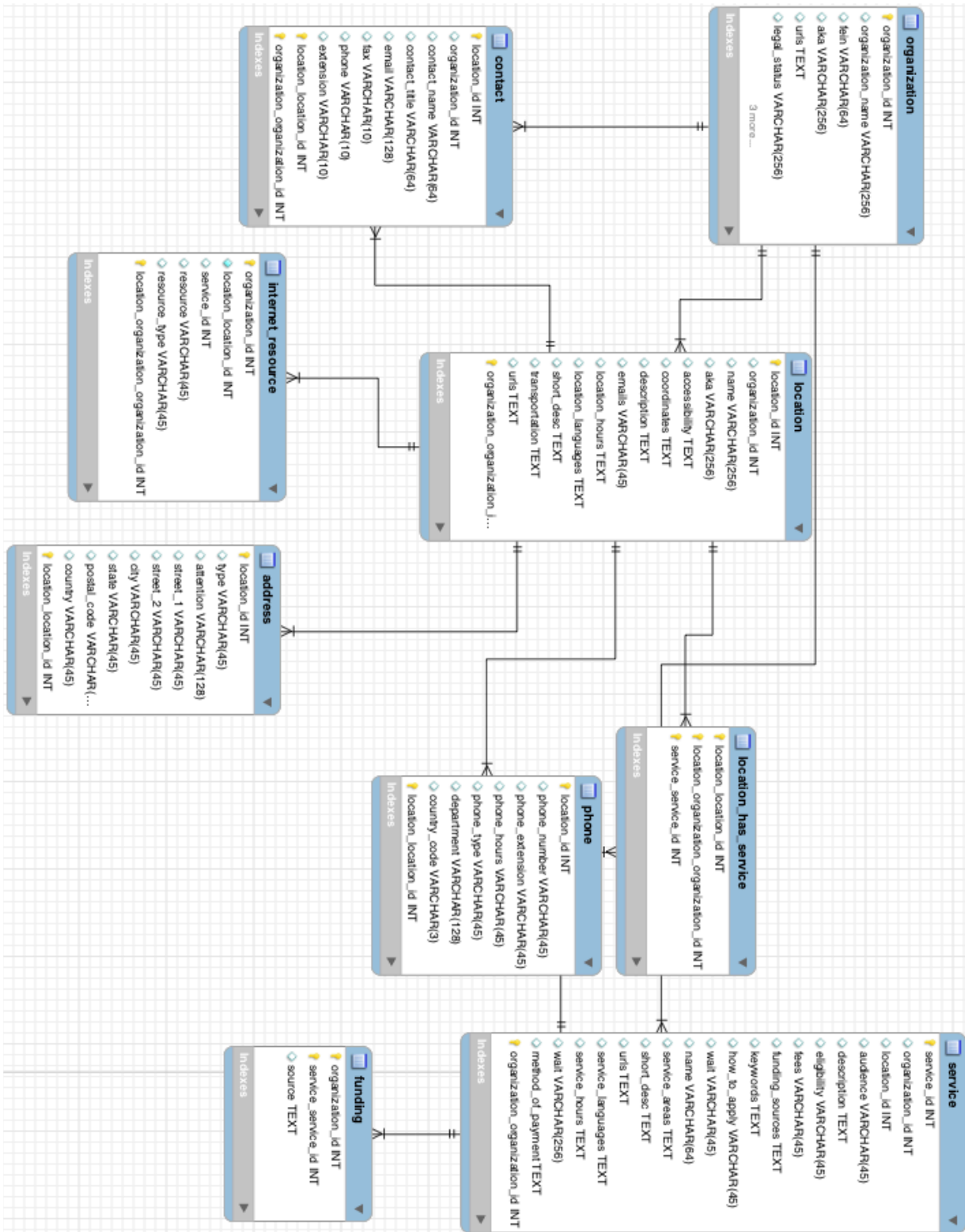*Table B.35 Data Table "search_result_interaction_target"*

APPENDIX B

Figure C.1. Simplified ERD

Figure D.1. OpenReferral ERD