Software Requirements Specification

for

Red Team Health Inspection Software

Version 1.0 prepared: 10/22/2015

Prepared by Red Team

CS320

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of the Data Management System for the daily tasks of the health inspector. In this document, we will focus on high level descriptions of the data entry and mapping modules within the desktop application, and the report printing module for the mobile application.

Ultimately, we intend to create a product which is able to facilitate the daily tasks of a health inspector. The client is looking for an application which will be able to replace his current physical filing system, which is hard to navigate and does not give a complete view of the current state of the town's public health. It would allow him to search the food code for violations based on keywords and save information about restaurants, water wells and septic tanks within the towns of Sunderland and Leverett. It would also be able to store the information he writes in the food inspection forms and search for that information based on keywords or restaurant name.

A similar system would be put in place for the wells and septic tanks of both towns, because their filing system is equally unnavigable, and data frequently gets lost. As part of a long term plan, the client would also like to be able to see information about well locations and water quality laid out on a map. The application needs to be user friendly enough to be used by computer beginners, secure enough that sensitive data can be stored on it and reliable enough that it can be trusted to hold records of legal documents and serve them up correctly.

1.2 Document Conventions

We will be referring to following terms by their equivalents on the right side of the table.

"health inspector"	"the user" (restricted to use cases)
"health inspector"	"the client"
"students developing this system"	"the developers"
"a health inspection report, water well report, or septic tank report"	"a report" (Refer to Section 1.5 for report details.)
"the data of health inspection reports, water well reports and septic tank reports"	"the data" (Refer to Section 1.5 for report details.)
"the towns of Leverett and Sunderland"	"the towns"

We will be referring to the mobile application and the desktop applications as “the mobile application” and “the application” respectively.

1.3 Intended Audience and Reading Suggestions

The intended audience of this document is directed at those who are trying to familiarize themselves with the functionality that this system will be required to produce. In particular this document should be of use to both the client and to the developers.

1.4 Product Scope

The scope of this software is contained within the practical uses of the client. This means that the software will be required to serve the client’s needs. The scope of this system entails the managing data described in **section 1.1 “Purpose”**, and therefore entails a system capable of handling information about restaurants, water wells, and septic tanks within the towns.

1.5 References

Food Establishment Inspection Form:
http://people.cs.umass.edu/~ridgway/compsci320/customer/FoodInspectionForm.pdf

Septic System Forms:
http://www.mass.gov/eea/agencies/massdep/water/approvals/title-5-septic-system-forms.html

SRS template:
We will be using a template prepared by Karl E. Wiegers covering IEEE 830 specifications. Permission was granted to use, modify and distribute the template.

2. Overall Description

2.1 Product Perspective

This will be a new product, created specifically for the client. This product will consist of two parts: desktop application and mobile application. The desktop application will be used for managing the data of the reports on a local machine when the client is in the office, while the mobile application will be used for saving report data and printing forms during the inspection.

Since this product will be used to manage the data, it will need a database to store the data. Both the desktop application and the mobile application will communicate with the same database. However, since we assume that there is no Internet access for the mobile device, the mobile application will have somewhere to store the data temporarily. The data on the mobile device will be transferable to the main database on a local machine.

2.2 Product Functions

- Data Entry:
  - Enter a new health inspection report into the database
  - Enter a new water well report into the database
  - Enter a new septic tank report into the database
  - Display the data of a preexisting health inspection report
  - Display the data of a preexisting water well report
  - Display the data of a preexisting septic tank report
  - Update the data of a health inspection report in the database
  - Update the data of a water well report in the database
  - Update the data of a septic tank report in the database
  - Delete a health inspection report from the database
  - Delete a water well report from the database
  - Delete a septic tank report from the database

- Mapping:
  - Provides a user interface for browsing the data and lot boundaries
  - Populate a map of the towns
  - Retrieve data from database (includes lot boundaries)
  - Display locations by marking on map. Each marker contains data on that location and information on lot boundaries
  - Display data associated with a marker on map

- Report Printing (mobile application):
  - Print a completed report to give to any necessary recipients.
  - Print a specific page of a completed report.
  - Enter any necessary additional information to a report if needed before printing the report.

2.3 User Classes and Characteristics

Health inspector
The primary user of this product. The health inspector will use both the desktop application and the mobile application to manage the data. He will need a user-friendly interface.

Secretary
The secondary user of this product who help the health inspector manage the data. The secretory will also use both the desktop application and the mobile application.

2.4 Operating Environment

Minimum performance specifications:

Desktop Application:

- Operating System: Microsoft Windows 7 or higher
- Processor: Intel® Core™ i3 @ 2.4GHz
- RAM: 2GB DDR3 SDRAM
- Disk Space: 160GB

Mobile Application:

- Operating System: Android 4.4 or higher
- Processor: 1GHz
- RAM: 1GB
- Storage Capacity: 8GB

2.5 Design and Implementation Constraints

The mobile application may not always have Internet access. The data on the mobile device might need to be stored temporarily on the device before being transferred to a local machine.

The client will be using a printer to print certain documents from the mobile application. The make and model of the printer will determine the design and implementation of the feature.

The application needs to be accessible on a commercial desktop or laptop computer.

2.6 User Documentation

A user manual will be created which describes and illustrates all system functions for both desktop and mobile applications.

2.7 Assumptions and Dependencies

We will be assuming that the desktop application has reliable Internet access.
We will be assuming that the database is going to kept in a secure, reliable location.
We will be assuming that the mobile application is running the Android operating system.
We will be assuming Internet access is minimal for the mobile device when used outside the office.

3. External Interface Requirements

3.1 User Interfaces

3.1.1 Data Entry:

- Screen Layout Constraints:
  - Application will be used on a laptop or desktop monitor.
- Standard Buttons and Functions:
  - Home: Redirects user to the application home page.
  - Enter Form: Submits the entered data into the database as a new form.
  - Edit Button: Enables editing of information
  - Save Button: Saves the updated form
  - Delete Button: Deletes the specific form from the database
- Error Message Display Standards:
  - Invalid Data: Displays “Invalid data: Please check your input”
  - Missing Data: Displays “Missing data: Please check all the fields”
  - Submit Confirmation: Displays “Do you want to submit the form?”
  - Delete Confirmation: Displays “Do you want to delete the data?”
  - Save Confirmation: Displays “Do you want to save the changes you made?”

3.1.2 Mapping

- Screen Layout Constraints:
  - Application will be used on a laptop or desktop monitor.
- Standard Buttons and Functions:
  - Marker Button: Displays data retrieved from the database
  - Retry Server Button: Attempt to reconnect to the database
  - Go to Login Button : Redirects the user to the login page
- Error Message Display Standards:
  - Invalid Click : “Mouse input invalid: Please use mouse left-click input”
  - Connection Unavailable : “The database is unavailable right now please press ‘Retry Server’ to attempt to reconnect”
  - User Not Logged In : “You are not logged in, log in to view this page”
- Screen Images:
  - Background Image : A visual that shows a bird’s eye view of the towns.

3.1.3 Printing

- Buttons:
  - Print Report: Prints the report that is currently selected.
  - Print Page: Prints a specific page of the report that is currently selected.

3.2 Hardware Interfaces

The desktop application requires that the client has access to a screen, mouse and keyboard.
To print reports, the mobile application requires a printer that can connect to the mobile device.

3.3 Software Interfaces

This application will have to interface with many other modules to create the final application. The data entry and mapping modules we are implementing will have to interface with the user authentication and search modules which are being implemented by the blue team in order to create a functioning desktop application. They will also have to interface with the orange team when they need to query the database for information.

The mapping portion of this application will need to directly interface with the database provided by orange team. The application will retrieve data from the database.

The mobile application report printing module will need to interface with the green team to obtain information about data entry on the mobile application, and with the orange team to get information from the database.

Project	Sub-project	Team Assigned
Application	User Authentication	Blue
	Data Entry	**Red**
	Reporting and Search	Blue
	Mapping	**Red**
Mobile application	User Authentication	Orange
	Data Entry (Forms)	Green
	Data Entry (GIS)	Blue
	Report Printing	**Red**
	Browsing and Search	Green
	Delayed Upload	Green
Database	Basic	Orange
	Federation	Orange

A lot of data will be shared between the database and the rest of the desktop and mobile applications. Everything else will rely on the database for information. The data entry portion of the desktop application will have to interface with the database directly, inserting information in. The user authentication module which is being developed by the blue team will need to use the database to determine if a user is allowed to use the service. Reporting and search will need to be able to query from the database and go by the same conventions used by the data entry team when they first put the information in the database. Finally, the

mapping portion of the application will need to gather information from the database to be able to display the locations of wells and septic tanks accurately.

The printing module for the mobile application will need to interface with the database to acquire the information it needs to print. This means that it will have to go by the same conventions used by the data entry teams. The printing portion of the application will also need to be able to interface with the software powering the printer, to tell it what needs to be printed.

3.4 Communications Interfaces

Communications protocols will be decided at a later date.

4. System Features

4.1 Data Entry

4.1.1 Description and Priority

The application will allow the user to enter, save, edit and remove the data on a local machine. The ability for the user to enter data with this form is high priority.

4.1.2 Application Data Entry Use Cases

1. Entering data into the application
2. Edit health inspection report from the application.
3. Deleting a report

Use Case 1: Entering data into the application.

Actors:	Client(Primary), Application
Precondition:	Client has logged into the application.
Main Success Scenario:	1. Client navigates to the respective data entry form within the application. 2. Client enters data into the form. 3. Client submits form. 4. Application checks for bad data or missing input. 5. Application asks if the client would like to make any changes before submitting. 6. Client confirms that they wish to submit the form. 7. Application saves the form to the database.

Exception Scenarios:	1. The client does not fill in a required field in the data entry form. a. The application does not submit the form and asks the client to complete all fields before proceeding with submission. 2. The client enters bad input into the data entry form. a. The application does not submit the form and asks the client to correct the input before proceeding with submission. 3. The application is unable to save the form to the database. a. The form is saved to local storage for later entry to the database.

Use Case 2: Edit health inspection report from the application.

Actors:	Client
Precondition:	Client has logged into the application and the target health inspection report is in the database.
Main Success Scenario:	1. Client navigates to the health inspection report list within the application. 2. Client clicks and opens the target health inspection report. 3. Application retrieves data from database and shows the detail of the report. 4. Client clicks “Edit” button on the application. 5. Application switches to edit mode and all the fields are able to be edited. 6. Client changes data in the health inspection form. 7. Client clicks “Submit” button to submit the form. 8. Application checks for bad data or missing input. 9. Application asks if the client would like to make any changes before submitting. 10. Client confirms they wish to submit the form and the form is submitted.
Exception Scenarios:	1. The client does not fill in a required field in the data entry form: a. The application does not submit the form and asks the client to complete all fields before proceeding with submission. 2. The client enters bad input into the data entry form: a. The application does not submit the form and asks the client to correct the input before proceeding with submission.

	3. The client clicks cancel when the application asks for confirmation: a. The application doesn't submit the form and return to the Edit mode.

Use Case 3: Deleting a report.

Actors:	Client
Precondition:	Client has logged into the application and is able to view previously entered health inspection reports.
Main Success Scenario:	1. Client views report which they want to delete. 2. Client performs action to delete report. 3. Application verifies that user wishes to complete this action. 4. Application deletes report. 5. Application notifies user that the report has been deleted.
Exception Scenarios:	1. Report is currently being edited or viewed by another user when client attempts to view and delete report. a. Client is prevented from viewing report until it is no longer accessed by a different user. 2. Client accidentally performs action to delete report. a. Client chooses that they do not wish to complete this action when the application requests verification. 3. Application attempts to verify delete action but receives no response from the client. a. Deletion request times out and the report is not deleted. 4. Client disconnects after verifying delete action with application. a. Application deletes report. b. Notification of deletion is not rendered.

4.1.3 Functional Requirements

REQ-1: The data should save to the database completely.
REQ-2: User authentication must be complete for this feature to work.
REQ-3: The database must be complete and connected to the application for this feature to work.
REQ-4: Missing required fields on all forms after an attempted form submission will result in a notification to the user to complete the form.
REQ-5: Invalid input of required fields on all forms after an attempted form submission will result in a notification to the user to correct their input.

4.2 Mapping

4.2.1 Description and Priority

The application will allow the user to navigate, enter, save, edit and remove the data on a local machine. Priority: Medium.

4.2.2 Application Mapping Use Cases

1. Clicking on the map
2. Clicking on a well
3. Clicking on a septic tank

Use Case 1: Clicking on the map

Actors:	Client
Precondition:	Client has logged into the application and is able to view previously entered health inspection reports.
Main Success Scenario:	1. Client navigates to the map form within the application. 2. Client clicks on map. a. If the client clicked on a well, go to alternate scenario 1 b. If the client clicked on a septic tank, go to alternate scenario 2
Exception Scenarios:	1. If the client has not clicked on an object, nothing will happen because the map is static 2. The client is looking at an outdated version of the map or map does not load 3. The client has not refreshed the map for a long period of time or the map has not loaded, making it so the user can't see the most relevant version of the map 4. The map does not load a. The application does not correctly load the map so the user can not see the well that he or she is clicking on and cannot read the information about that well. 5. The client is not logged in and is trying to access the map a. The application notifies the client that he/she is not currently logged in b. A pop up box appears prompting the client to redirect to the User Authentication Page

	6. The database is offline a. The database cannot send data to the map because is offline and not connected to the application b. User presses "Retry Database" button and attempts to turn the database back on reconnect to it. c. If "Retry Database" button works, the user is reconnected to the database and the information is displayed on the map d. If "Retry Database" button doesn't work, notify user so they can retry again.

Alternate Scenario 1: Clicking on a well.

Actors:	Client
Precondition:	Client has navigated to the map form within the application, and has clicked on a well.
Main Success Scenario:	1. Information for the current selected well is displayed in a pop-up box by the well. 2. Client can close the pop-up box after reading all the well information 3. Map is now in the main state, refer to use case 1 for further actions.
Exception Scenarios:	1. The client is looking at an outdated version of the map or map does not load. 2. The client has not refreshed the map for a long period of time or the map has not loaded, making it so the user can't see the most relevant version of the map.

Alternate Scenario 2: Clicking on a septic tank.

Actors:	Client
Precondition:	Client has navigated to the map form within the application, and has clicked on a septic tank.
Main Success Scenario:	1. Information for the current selected septic tank is displayed in a pop-up box by the septic tank.

	2. Client can close the pop-up box after reading all the septic tank information 3. Map is now in the main state, refer to use case 1 for further actions.
Exception Scenarios:	1. The client is looking at an outdated version of the map or map does not load. 2. The client has not refreshed the map for a long period of time or the map has not loaded, making it so the user can't see the most relevant version of the map.

4.2.3 Functional Requirements

REQ-1: The map must be able to query the database.
REQ-2: The map must be able to display applicable information in the database

4.3 Printing

4.3.1 Description and Priority

The mobile application will allow the client to print reports to give to any necessary recipients.
Priority: High.

4.3.2 Mobile Application Printing Use Cases

1. Print report
2. Print specific page of a report

Use Case 1: Print report

Actors:	Client
Precondition:	Client has logged into the application and is able to view previously entered reports.
Main Success Scenario:	1. User clicks 'print report' button for a completed report in the mobile application 2. Application ensures that a printer is connected 3. Application formats the data of the report into an image resembling the official form 4. Application sends the formatted report to the printer
Exception Scenarios:	1. Mobile device is not connected to a printer a. Application informs the user that no printer is connected

	b. Application returns the user to the main view 2. Report is missing necessary information a. Application informs the user that necessary information is missing b. Application redirects the user to the data entry page

Use Case 2: Print specific page of a report

Actors:	Client
Precondition:	Client has logged into the application and is able to view previously entered reports.
Main Success Scenario:	1. User selects “Print This Page” button 2. Application ensures that a printer is connected 3. Application prints the current page that is displayed on the mobile device
Exception Scenarios:	1. Mobile device is not connected to a printer a. Application informs the user that no printer is connected b. Application returns the user to the main view 2. Report is missing necessary information a. Application informs the user that necessary information is missing b. Application redirects the user to the data entry page

4.3.3 Functional Requirements

REQ-1: A printer is connected to the mobile device. If not, the mobile application should alert the user.

REQ-2: The report to be printed is completed. If not, the mobile application should prompt the user for additional information.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

No specific performance requirements have been identified.
We will be able to stress test the system based on these requirements when we have more information.

5.2 Safety Requirements

The whole application needs to be reasonably reliable, and it should have the capacity for redundancy. It is not a mission critical system, but it will be used daily by health officials and store important town information. The client will have to be trained in the use of the system to ensure that no data is lost due to user error.

5.3 Security Requirements

The client will need to log into the system for all operations on local machine and mobile device, because the system will be dealing with sensitive data. Only users who have been authorized can access the data. This will prevent files from being deleted unintentionally or maliciously, which could impact the public health and cleanliness of houses and restaurants in the town.

5.4 Software Quality Attributes

This application must be reliable, correct, robust and usable. This system will be used to store important information about town resources, so it must always be available and accurate. Since this system is going to be used by clients who are used to doing this process manually on paper, they will need to have a user interface which will be intuitive and easy to transition into.

5.5 Business Rules

Since this is going to be a closed system with only two main users, users in this system will be given administrative access by default. All users will be able to modify the database.

Appendix: References to other modules

Appendix: Analysis Models and Visuals