Software Requirements Specification

for

Red Team Health Inspection Software

Version 1.1 prepared: 11/08/2015

Prepared by Red Team

CS320

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Red Team	10/22/15	First submission	1.0
Red Team	11/9/15	Final submission	2.0

1. Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of the health inspection software which will be used 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 records 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”
“the following data : water-well : location, water quality, water quality trends lot : location, lot number, lot boundaries septic tank : location, tank size, pump information, last pump information”	“the mapping data”
“desktop application”	“the application”

1.3 Intended Audience and Reading Suggestions

The intended audience of this document is users 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 requires that the data management system described in **section 1.1 “Purpose”** is 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.

Types of Software Testing - Definitions:
http://www.softwaretestinghelp.com/types-of-software-testing/
http://searchsoftwarequality.techtarget.com/definition/conformance-testing

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 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 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 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 secretary will also use both the desktop application and the mobile application.

2.4 Operating Environment

Minimum performance specifications:

Desktop Application:

- Operating System: Windows, Mac OS X, or Linux
- Processor: Intel® Core™ i3 @ 2.4GHz
- RAM: 2GB DDR3 SDRAM
- Disk Space: 160GB
- Network Access

Mobile Application:

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

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. We will need to make sure that the printer in question is compatible with our software.

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.
    - Submit Button: Submits the entered data into the database as a new record.
    - Edit Button: Enables editing of information
    - Delete Button: Deletes the specific record 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 Log-in Button : Redirects the user to the log-in 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. Applying this to our own modules, the data entry software and mapping software will have to be written with the structure of the database in mind.

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

The mobile application will need to be able to communicate with the database via a networking protocol, to be able to sync data which has been cached locally.

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 a record of the information in the form to the database.
Post-condition	The data is saved to the database.

Exception Scenarios:	1. The client does not fill in a required field in the data entry form. 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. 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 record to the database. The data 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 a record of the information in the form is submitted to the database.
Post-condition	The data is changed and saved to the database.
Exception Scenarios:	1. The client does not fill in a required field in the data entry form: The application does not submit the form and asks the client to complete all fields before proceeding with submission. Return to step 5.

	2. The client enters bad input into the data entry form: The application does not submit the form and asks the client to correct the input before proceeding with submission. Return to step 5. 3. The client clicks cancel when the application asks for confirmation: The application doesn't submit the form and return to the Edit mode.

Use Case 3: Deleting a report or inspection.

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.
Post-condition	The data of the target report is removed from the database.
Exception Scenarios:	1. Report is currently being edited or viewed by another user when client attempts to view and delete report. Client is prevented from viewing report until it is no longer accessed by a different user. 2. Client accidentally performs action to delete report. 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. 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

Map allows to the user to view the location of wells and septic systems, view the associated data with wells and septic systems, and view trends over time. See Appendix A for the data flow chart. 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. The user has a version of the map available.
Main Success Scenario:	1. Client navigates to the map form within the application. 2. Client clicks on map. If the client clicked on a well, go to alternate scenario 1 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 Alert the user to download the latest version of the map.

	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 Alert the user to relaunch the mapping application to see the latest 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. b. Alert the user to reload the application in order to view the latest data. 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 Map” button and attempts to turn the database back on reconnect to it. c. If “Retry Map” button works, the user is reconnected to the database and the information is displayed on the map d. If “Retry Map” 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. Prompt the user to relaunch the mapping application to see the latest version of the map. 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. Prompt the user to relaunch the mapping application to see the latest version of the map.
Post-condition	The client is viewing a pop-up display that shows appropriate well data.

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 Prompt the user to relaunch the mapping application to see the latest version of the map. 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.
Post-condition	The client is viewing a pop-up display that shows appropriate septic data.

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 viewing a previously entered report.
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 record 4. Application sends the formatted report to the printer 5. Application redirects the user to the main view
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
Post-condition:	The report has been printed and the application has been returned to the main view.

Use Case 2: Print specific page of a report

Actors:	Client
Precondition:	Client has logged into the application and is viewing a previously entered report.
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 4. Application redirects the user to the main view
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
Post-condition:	A page of the report has been printed and the application has been returned to the main view.

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 A: Analysis Models and Visuals

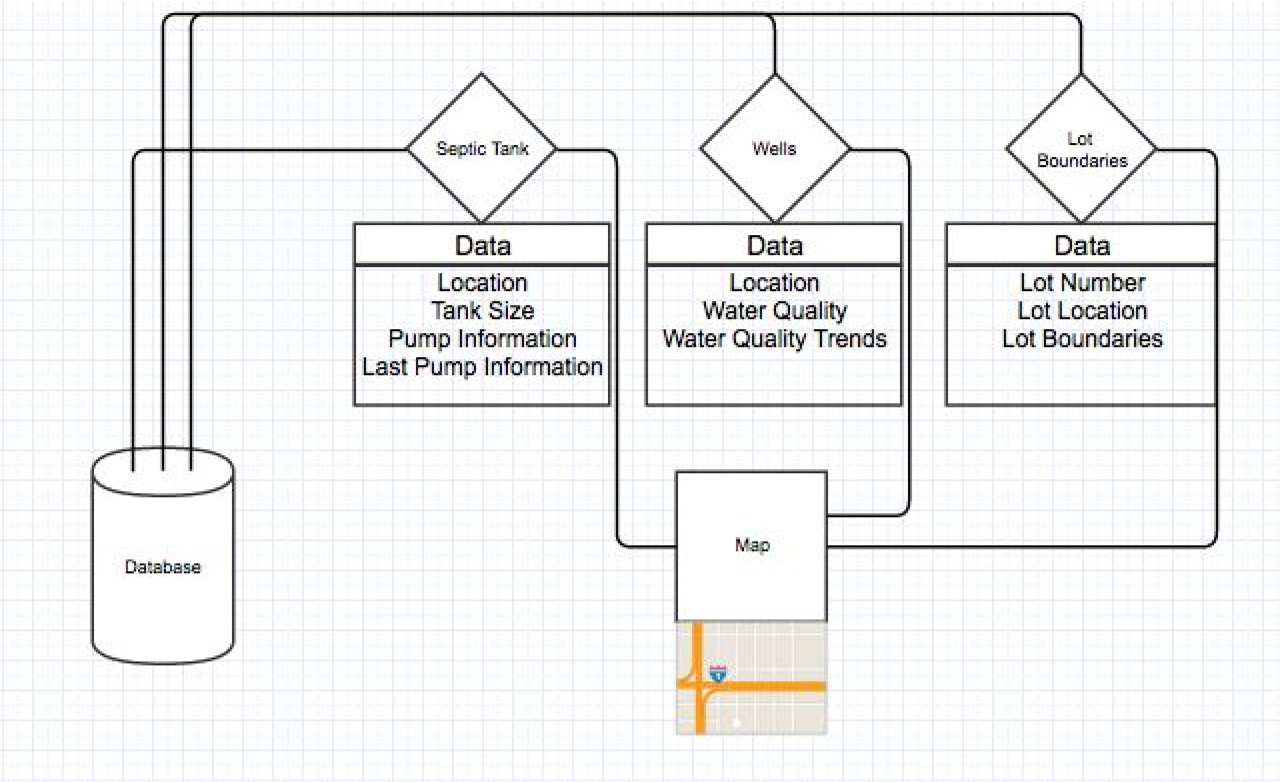


Figure 1. Map data flow chart

The figure above represents data flow between the map application and the database. Provides an in-depth description of the specific types of data that are used by the mapping application.

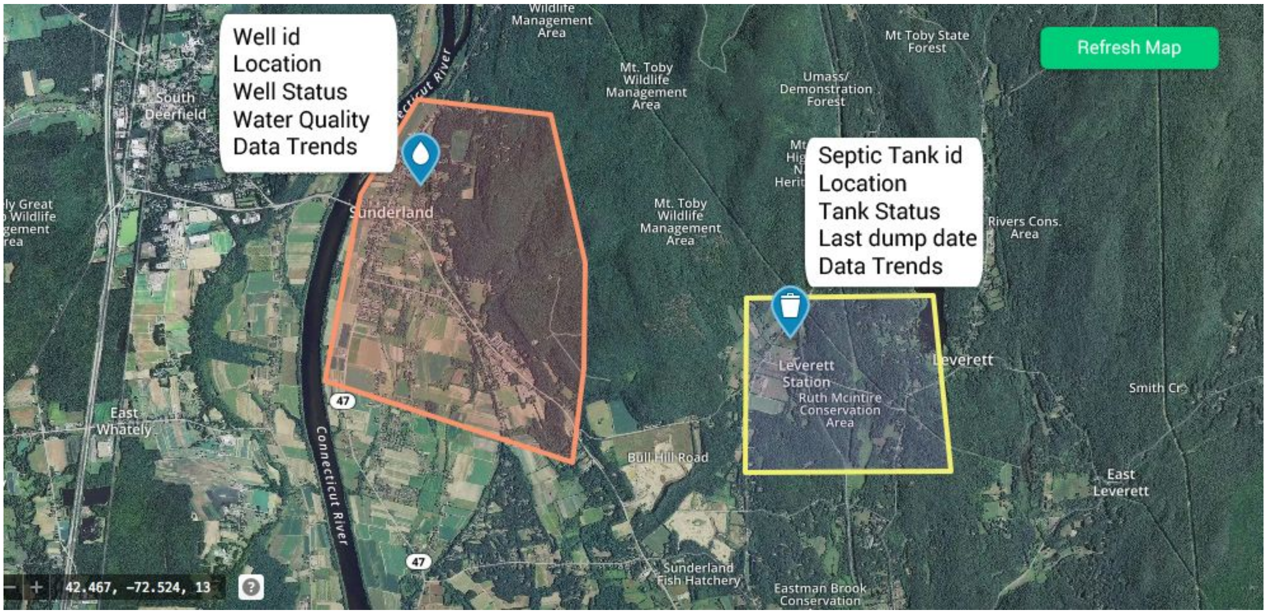


Figure 2. Map Wireframe

The figure above represents a very basic representation of the map with some key features.

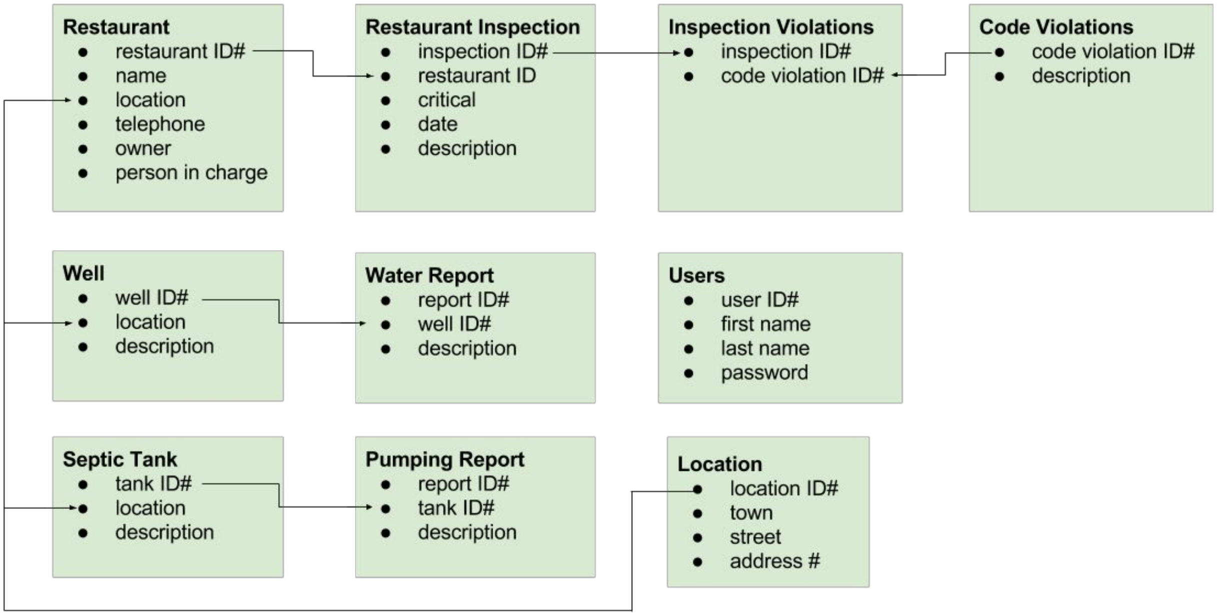


Figure 3. Data Diagram

The figure above represents the tables which will exist in the database for the application.

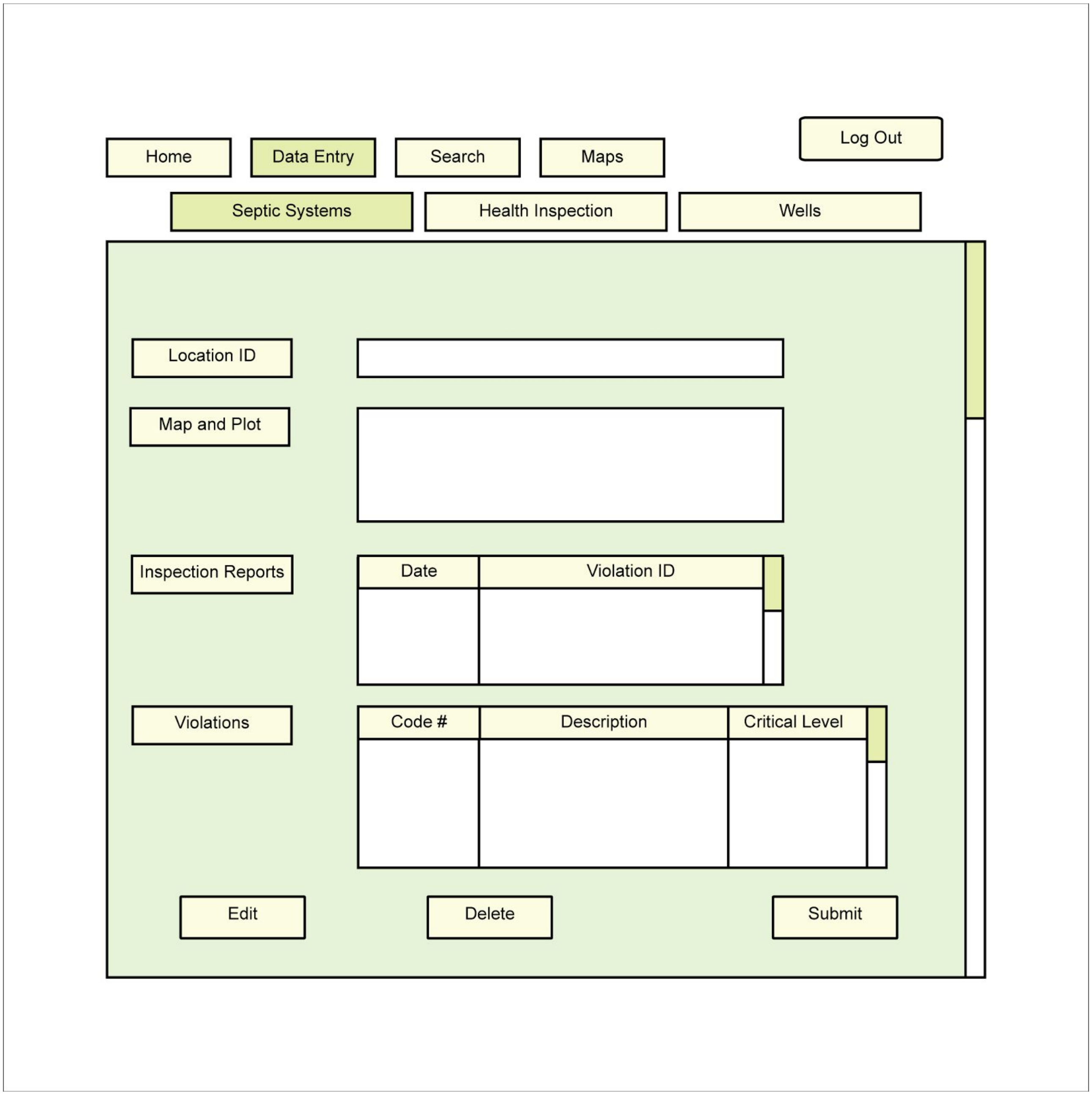
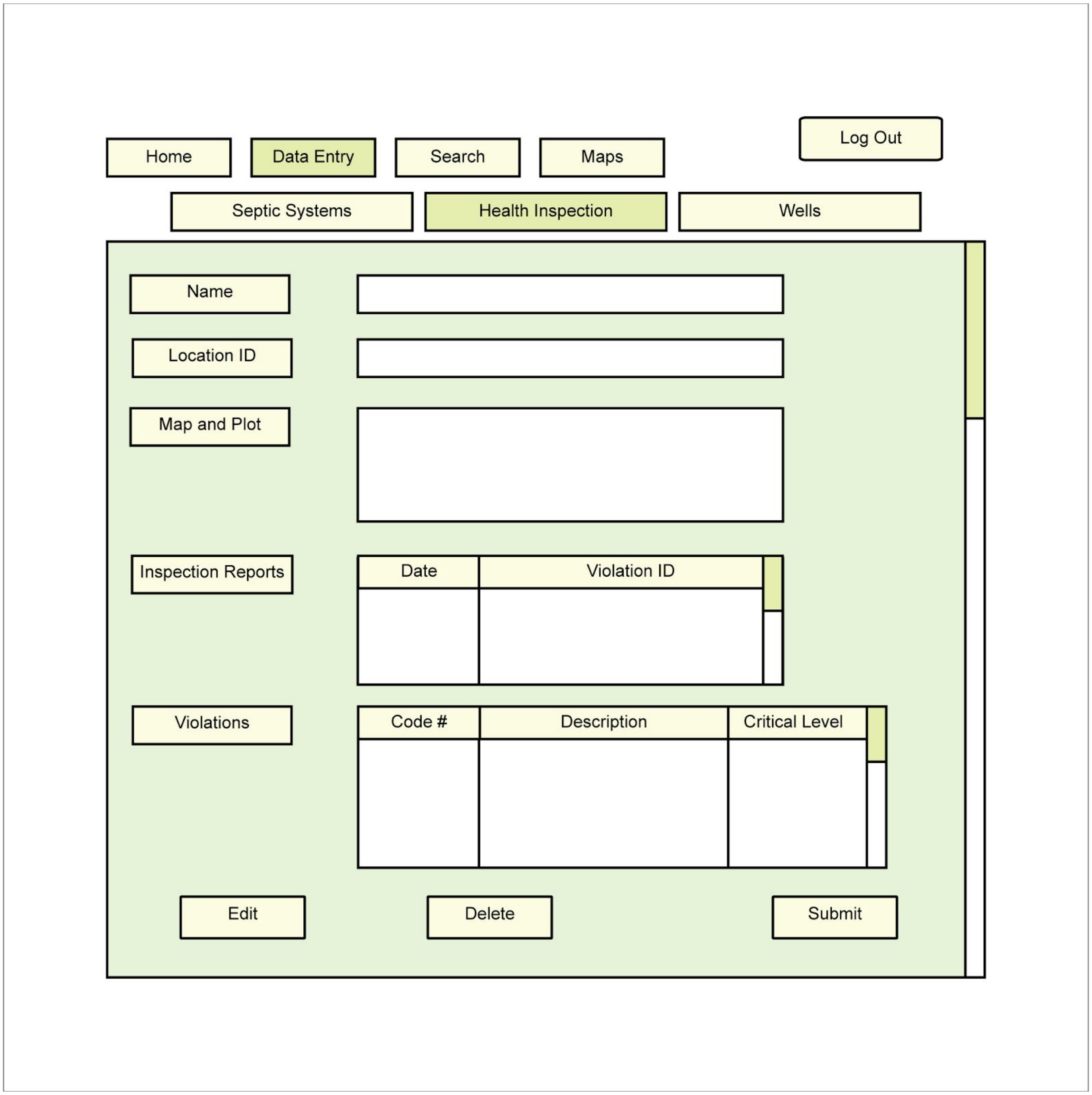


Figure 4. Septic Systems Data Entry UI

The figure above is a mock-up of what the user interface could possibly look like.



Date	Violation ID

Code #	Description	Critical Level

Figure 5. Health Inspection Data Entry UI

The figure above is a mock-up of what the user interface could possibly look like.

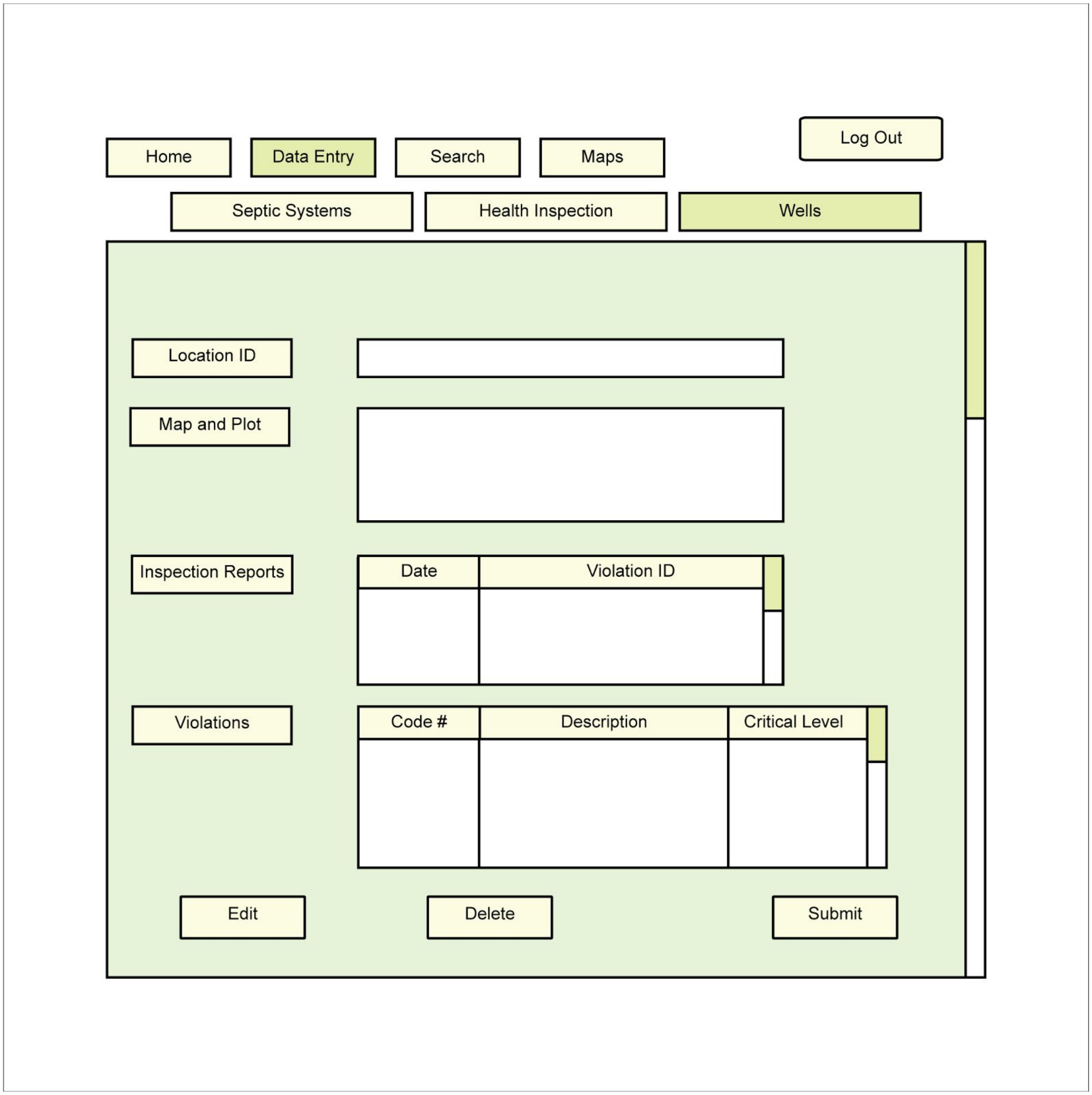


Figure 6. Well Data Entry UI

The figure above is a mock-up of what the user interface could possibly look like.

Appendix B: TEST PLAN

TABLE OF CONTENTS

VERSION HISTORY

Version #	Implemented By	Revision Date	Reason
1.0	Red Team	11/9/15	Integration into SRS

UP Template Version: 12/31/07

1. INTRODUCTION

PURPOSE OF THE TEST PLAN DOCUMENT

The purpose of this document is to form a testing plan for the Health Inspection Software project. This document will focus on the testing of the three sub-projects Mapping, Application Data Entry, and Tablet printing. The document will serve as a guide for how testing will be implemented after work on implementation begins.

2 FORMS OF TESTING

2.1 Compatibility Testing
2.2 Conformance Testing
2.3 Functional Testing
2.4 Stress & Load Testing
2.5 Performance Testing
2.6 Regression Testing
2.7 System Testing
2.8 Unit Testing
2.9 User Acceptance Testing

2.1 Compatibility Testing:

Testing how well software performs in a particular hardware, software, operating system, or network environment and different combinations of the above.

2.2 Conformance Testing:

Conformance testing, also known as compliance testing, is a methodology used in engineering to ensure that a product, process, computer program or system meets a defined set of standards. These standards are commonly defined by large, independent entities such as the Institute of Electrical and Electronics Engineers (IEEE), the World Wide Web Consortium (W3C) or the European Telecommunications Standards Institute (ETSI).

2.3 Functional Testing:

This type of testing ignores the internal parts and focus on the output is as per requirement or not. Black-box type testing geared to functional requirements of an application.

2.4 Stress & Load Testing:

System is stressed beyond its specifications to check how and when it fails. Performed under heavy load like putting large number beyond storage capacity, complex database queries, continuous input to system or database load. Its a performance testing to check system behavior under load. Testing an application under heavy loads, such as testing of a web site under a range of loads to determine at what point the system’s response time degrades or fails.

2.5 Performance Testing:

Term often used interchangeably with ‘stress’ and ‘load’ testing. To check whether system meets performance requirements. Used different performance and load tools to do this.

2.6 Regression Testing:

Testing the application as a whole for the modification in any module or functionality. Difficult to cover all the system in regression testing so typically automation tools are used for these testing types.

2.7 System Testing:

Entire system is tested as per the requirements. Black-box type testing that is based on overall requirements specifications, covers all combined parts of a system.

2.8 Unit Testing:

Testing of individual software components or modules. Typically done by the programmer and not by testers, as it requires detailed knowledge of the internal program design and code. May require developing test driver modules or test harnesses.

2.9 User Acceptance Testing:

Testing the application with end users. Functionality, usability, and the ability of the system to meet the needs of the users who will be working with the system, will all be tested by having end users complete tasks with temporary data on an end user machine.

See Software Requirements Specifications section 1.5 for source of test type definitions.

3 TEST COMPONENTS

3.1 Description of Test Risks / Issues
3.2 Description of Items to be Tested
3.3 Description of Test Approach(s)
3.4 Description of Test Regulatory / Mandate Criteria
3.5 Description of Test Pass / Fail Criteria
3.6 Description of Test Entry / Exit Criteria
3.7 Description of Test Deliverables
3.8 Description of Test Suspension/ Resumption Criteria
3.9 Description of Test Environmental / Staffing / Training Needs

3.1 Test Risks and Issues:

This section will describe the risks associated with product testing or provide a reference to the documentation where the descriptions are located. This section also outlines appropriate migration strategies and contingency plans.

3.2 Items to be Tested:

This section describes the items, features, and functions to be tested that are within the scope of the test plan. This includes a description of how they will be tested, along with when they will be tested, how they will be tested, and to what quality standards they will be tested. Lastly, there will be a description of the items that are to not be tested.

3.3 Test Approach:

The overall testing approach is described here, along with an outline of any planned tests.

3.4 Test Regulatory and Mandate Criteria:

This section describes any regulations or mandates that the system must be tested against.

3.5 Test Pass and Fail Criteria:

This section describes the criteria used to determine if a test item passed or failed its test.

3.6 Test Entry and Exit Criteria:

This section describes the entry and exit criteria used to start testing and determine when to stop testing.

3.7 Test Deliverables:

This section describes the deliverables that will result from the testing process. Such deliverables include, documents, reports, charts, and more.

3.8 Description of Test Environmental / Staffing / Training Needs:

This section describes the suspension criteria that may be used to suspend all testing or portions of testing. This section also describes the resumption criteria that may be used to resume testing.

3.9 Test Environmental/ Staffing/ Training Needs:

This section describes specific requirements needed for testing to be performed. Such requirements include, hardware, software, staffing, training, and more.

4 COMPATIBILITY TESTING

Test risks and Issues

Since we are working in collaboration with 3 other teams to make our final product, there is always the chance that parts of our modules will not fit cleanly together. To mitigate this risk, we will be keeping in close contact with the other teams through the designing stage of the project, to make sure that we are able to create a complete integration plan.

Items to be tested

Printer

Test to see if the application supports all database requesting needs

Test to see if printer is recognized by tablet

Application UI

Test to see if the application make a proper database request

Database retry button testing

Test to see if the database retry button supports attempts at reconnecting to the database

Other application components

Test if the data entry module is compatible with the other application components

Data Entry

Test if the data entry module is compatible with the client's operating environment

Test Approaches

Printing

Printer connectivity is tested by checking the following:

1. The application notices that the printer is connected when the printer is connected to the tablet
2. The application notices that the printer is disconnected when the printer is disconnected from the tablet

Mapping

Application support is tested by running the application on the machine, and checking the following :

1. GUI elements are responsive and display correctly
2. Application is able to successfully connect, and retrieve data from the database
    a. Manually check all mapping data retrieved from the database, and directly

compare this data to the actual data in the database.

Data Entry

The Data Entry module will be tested for compatibility by performing the following.

1. Determine all operating systems and/or web browsers which the client is expected to be able to use to run the application
2. Perform data insertion, data editing, and data deletion with the data entry module in every potential combination of these environments.

Test Pass/Fail Criteria

Printer connection
PASS:
The Application notices that the printer is connected to the tablet
FAIL:
The application does not notice that the printer is connected to the tablet

Printer Disconnection
PASS:
The Application notices that there are no printers connected when the printer is disconnected from the tablet
FAIL:
The application fails to notice that the printer has been disconnected from the tablet

Mapping GUI
PASS:
All GUI elements are able to properly display and will respond to appropriate user input.
FAIL:
Some or all GUI elements are unable to properly display. Alternatively some or all GUI fails to respond, responds incorrectly to appropriate user input, or responds to inappropriate user input

Mapping connectivity
PASS:
The application is successfully able to connect and retrieve necessary mapping data
FAIL:
The application is unable to successfully connect to the database. Or the application is able to successfully connect to the database but unable to retrieve some/any of the necessary mapping data

Database request
PASS:
All of the needed data requests are properly constructed and executed upon

application's request
FAIL:
All or some data requests are not properly sent or received by the database.

Database retry button
PASS:
After clicking the database retry button, it successfully attempts to reconnect to the database.
FAIL:
After clicking the database retry button, nothing happens and no request is sent to the database in an attempt to reconnect.

Data Entry Integration with Other Modules
PASS:
Data entry produces correct behavior with the other application components.
FAIL:
Data entry does not work as intended with the other application components.

Data Entry Integration with operating environments
PASS:
Data Entry module produces the same behavior in all operating environments which the client is expected to be able to use.
FAIL:
Data Entry module does not produce the same behavior in all operating environments which the client is expected to be able to use.

Database Compatibility
PASS:
Data can be saved to and retrieved from the database.
FAIL:
Data is not saved to or retrieved from the database.

Client Access Compatibility
PASS:
Data entry operates successfully on the client's computer.
FAIL:
Data entry doesn't operate on the client's computer.

Test Entry/Exit Criteria
Test Entry Criteria: All test hardware platforms must have been successfully installed , configured, and functioning properly. Proper test data is available. All modules are complete and integrated with the application. **Test Exit Criteria:** All Test plans have been run. A certain level of requirements coverage has been achieved.

5 CONFORMANCE TESTING

Test risks and Issues
There would be no risks associated with conformance testing. We will be testing against a current paper and pen system that the client has in place. The software that is being developed for this project is entirely new and will not directly affect the system being replaced.
Items to be tested
Printer Test to see if application formats reports correctly for printing **Application UI** Test to see if the application supports all database requesting needs
Test Approaches
Printing Testing will involve printing all types of reports to ensure that all types of reports are formatted correctly. Testing will also involve printing reports with differing data entered to ensure that all types of data are formatted correctly in a report. These different data will be text of varying length and will include different special characters that might reasonably be included in a report. For each report, compare to a printed official report to ensure that the formatting is identical.

Mapping

Testing will involve loading the application on different machine configurations and running through a series of steps that will test the visibility, and functionality of the application on separate machines. The steps will involve clicking on the different parts of the application, to see if the clicked components perform the necessary functions without error.

Test Pass/Fail Criteria

Printer Reports Format Correctly
PASS:
The reports printed are formatted the same as an official report
FAIL:
The reports printed are formatted different from an official report

Mapping GUI
PASS:
All buttons appear as they should, and provide the correct output when clicked on
FAIL:
Some or any of the buttons don’t appear as they should, and or some/any don’t provide the correct output when clicked on.

Database request
PASS:
The application is able to establish a successful connection to the database. In addition data is able to properly retrieved from the database.
FAIL:
The application is unable to establish a successful connection to the database. Or a successful connection is established but data can’t be properly retrieved.

Test Entry/Exit Criteria

Test Entry Criteria:

All test hardware platforms must have been successfully installed , configured, and functioning properly.

Proper test data is available.

Test Exit Criteria:

All Test plans have been run.

A certain level of requirements coverage has been achieved.

6 FUNCTIONAL TESTING

Test risks and Issues

There would be no risks associated with functional testing. We will be testing against a current paper and pen system that the client has in place. The software that is being developed for this project is entirely new and will not directly affect the system being replaced.

Items to be tested

Print Document

Examine the reports can be printed in the correct format when the printer is recognized and connected correctly

Print Specific Page

Test to ensure that the user specified page is printed

Data Saving

Test if the data is saved in the database successfully

Data Deleting

Test if the data is removed from the database successfully

Data Editing

Test if the application is able update data in the database.

Test Approaches

Data Entry

1. Fill out each type of inspection report and form, and save the data.
2. Open one of each type of inspection report and form and delete them.
3. Open one of each type of inspection report and form, update the data in each and save the changes to them.

Printing

1. Test to see if entire document is printed
2. Test to see if only the specified page is printed

Test Pass/Fail Criteria

Print Button works

PASS:

Reports can be printed in the correct format when the printer is recognized and connected correctly

FAIL:

Reports fail to be printed in the correct format when the printer is recognized and connected correctly

Print Specific Page

PASS:

The user specified page is printed

FAIL:

Nothing is printed or an unspecified page is printed

Data Saving

PASS:

Data is saved in the database successfully

FAIL:

Data is not saved in the database

Data Deleting

PASS:

Data is removed from the database successfully

FAIL:

Data is not removed from the database

Data Editing

PASS:

Application updates the data from database successfully

FAIL:

The data is not updated in the database

Test Entry/Exit Criteria

Test Entry Criteria:

All test hardware platforms must have been successfully installed , configured, and functioning properly.

Proper test data is available.

Test Exit Criteria:

All Test plans have been run.

A certain level of requirements coverage has been achieved.

7 STRESS & LOAD TESTING

Test risks and Issues

There would be no risks associated with stress and load testing. We will be testing against a current paper and pen system that the client has in place. The software that is being developed for this project is entirely new and will not directly affect the system being replaced.

Items to be tested

Printer

Test to see if printer can handle multiple print requests

Application UI

Test to see if the application supports all database requesting needs

Data Entry

Test to see if the application handles all input correctly, including missing inputs and bad data.

Test Approaches

Printing

In order to ensure that the printer can handle multiple requests from the tablet, we send multiple requests to the printer and see if the printer will print the contents of each request successfully.

Application UI

In order to test if the application supports all database requesting needs, we would test to see if all data can be successfully requested and retrieved.

Data Entry

To test the system for handling bad data or missing input, test data will be created which involves incorrect data and missing fields for every field in all inspections and reports. The data will be submitted for both entering data and editing data.

Test Pass/Fail Criteria

Printer can handle multiple requests

PASS:

Printer is able to print each request in successive order

FAIL:

Printer does not print anything, or only prints a certain number of requests

Application UI

PASS:

Application is able to successfully retrieve information from the database

FAIL:

Application can not successfully retrieve information from the database

Data Entry - bad data

PASS:

Application is able to correctly handle attempts at data entry or editing which involve saving invalid data or missing key fields to the database.

FAIL:

Application allows user to input missing fields or bad data or produces errors when such attempts are made.

Test Entry/Exit Criteria

Test Entry Criteria:

All test hardware platforms must have been successfully installed , configured, and functioning properly.

Proper test data is available.

Test Exit Criteria:

All Test plans have been run.

A certain level of requirements coverage has been achieved.

8 PERFORMANCE TESTING

Test risks and Issues

There are no documented risks that will be associated with performance testing. We will be testing against a current paper and pen system that the client has in place. The software that is being developed for this project is entirely new and will not directly affect the system being replaced. Once newer versions of the software are developed, risks and/or issues will be addressed.

Items to be tested

Printer

Ensure that a report can be printed efficiently

Application

Ensure that all data can be retrieved and accessed in reasonable amount of time (< 2 seconds per request)

Test Approaches

Printing

Print a variety of reports and note their printing times. Ensure that the process of printing a report never takes longer than a minute. Ensure that after printing a report, the application is available for further use and is not stuck on the printing process.

Application

Retrieve a large number of data and record the times it takes from request sent to request received.

Test Pass/Fail Criteria

Printer prints efficiently

PASS:

The printing process takes less than a minute

FAIL:

The printing process takes longer than a minute or the application hangs while printing

Application

PASS:

The requests are received in reasonable amount of time (< 2 seconds)

FAIL:

The requests take longer than expected or are not retrieved at all (>2 seconds)

Test Entry/Exit Criteria

Test Entry Criteria:

All test hardware platforms must have been successfully installed , configured, and functioning properly.

Proper test data is available.

Test Exit Criteria:

All Test plans have been run.

A certain level of requirements coverage has been achieved.

9 REGRESSION TESTING

Test risks and Issues

There are no documented risks that will be associated with regression testing. We will be testing against a current paper and pen system that the client has in place. The software that is being developed for this project is entirely new and will not directly affect the system being replaced. Once newer versions of the software are developed, risks and/or issues will be addressed.

Items to be tested

Printed Forms

Verify the printed forms are formatted correctly and look like the original inspection forms. Also verify that data found on original forms is contained in new printed forms.

Map

Verify that the map within the Mapping sub-project correctly displays the towns and data.

Data Entry

Verify that the application data entry module works in conjuncture with any modules added after it.

Test Approaches

Manual testing will be conducted to verify that the software being developed correctly replicates the previous system. In order to acquire this goal, each team must test their respective sub-project. The required testing steps for regression testing for each sub-project are listed below:

Printing

1. Verification of correctness for the printing function
    a. The printed form looks very similar to the original inspection form and contains all of the data in the original inspection form

Data Entry

1. Enter inspections and reports and confirm they have been saved in the database.
2. Edit inspections and reports and confirm they have been updated in the database.
3. Delete inspections and reports and confirm they are no longer in the database.

Mapping

1. Verification of correctness for displayed map
    a. Obtain a copy of a map of towns used by the client.
    b. Verify that the new system's map is correct by checking that the new system's map:
        i. Correctly displays towns and all boundaries within towns.
        ii. Correctly displays lot boundaries.
        iii. Correctly displays data about wells/septic tanks found on inspection forms when clicking on a map marker.

Test Pass/Fail Criteria

Prints Successfully

PASS:

Reports can be printed in the correct format when the printer is recognized and connected correctly

FAIL:

Reports fail to be printed.

Displays Towns Correctly

PASS:

The towns of Leverett and Sunderland and all land is displayed within the application.

FAIL:

The towns of Leverett and Sunderland and all land is not displayed within the application.

Displays Lot Boundaries Correctly

PASS:

All lot boundaries within the towns are correct and do not overlap with other lot boundaries.

FAIL:

Lot boundaries within towns are not correct and/or overlap with other lot boundaries.

Displays Data Correctly

PASS:

When clicked, the well/septic tank markers display all data that is found on the original inspection forms.

FAIL:

When clicked, the well/septic tank markers do not display all data that is found on the original inspection forms.

Data Entry

PASS:

Enter, edit, and delete functionality for reports and inspections behaves as intended after the addition of any other modules.

FAIL:

After the addition of new modules to the application, some or all of the data entry functions no longer produces correct behavior.

Test Entry/Exit Criteria

Testing will begin once all sub-projects within the application are done.

Testing will conclude once each sub-project is thoroughly tested as outlined in the regression testing template.

10 SYSTEM TESTING

Test risks and Issues

There are no documented risks that will be associated with system testing. We will be testing against a current paper and pen system that the client has in place. The software that is being developed for this project is entirely new and will not directly affect the system being replaced.

Items to be tested

Printer

Test to see if printer is functioning properly

Application UI

Test to see if the application supports all database requesting needs

Data Entry

Test to see if entering data, editing data, and deleting data for reports and inspections works correctly within the context of the entire system after all modules have been added to the application.

Test Approaches

Printing

Print a variety of reports and verify that the printed reports match the expected format. Ensure the process of printing different reports is continuous. Also ensure the printing processes can be finish in a acceptable time and the application is ready for further use after printing.

Test Pass/Fail Criteria

Printer works

PASS:

Printer is connected is successfully printing reports that match the expected format

FAIL:

Printer is connected but does not correctly print the selected report

Data Entry

PASS: Entering, editing, and deleting form and inspection data behaves as intended after all modules have been added to the application.
FAIL: Some or all functionality of the Data Entry form fails to produce the correct behavior when all modules are implemented together in the system.

Test Entry/Exit Criteria

Test Entry Criteria:

All test hardware platforms must have been successfully installed , configured, and functioning properly.

Proper test data is available.

All separate modules have been completed and added to the application.

Test Exit Criteria:

All Test plans have been run.

A certain level of requirements coverage has been achieved.

11 UNIT TESTING

Test risks and Issues

There would be no risks associated with unit testing. We will be testing against a current paper and pen system that the client has in place. The software that is being developed for this project is entirely new and will not directly affect the system being replaced.

Items to be tested

Data Saving

Test if the data is saved correctly. Go through each part of each module which will be accessing the database and make sure that they are entering data correctly.

Data Editing

Test if changes to data are submitted and saved correctly to the database. Attempt to edit each field of every form for test reports and test inspections to confirm no values fail to update with the new changes.

Data Deleting

Test if the data is removed successfully.

Test Approaches
Data Saving: Check if data in database matches the input. Data Editing: Check if data in database reflects changes made through data entry form. Data Deleting: Check if the data is still in the database after attempting to remove with data entry form.
Test Pass/Fail Criteria
Data Saving PASS: The data in database matches the input. FAIL: The data in database does not match the input. **Data Editing** PASS: The changes to the data made with the data entry form are reflected in the database. FAIL: The data in the database does not match what was submitted as an edit. **Data Deleting** PASS: The target data is not in the database. FAIL: The target data is still in the database.
Test Entry/Exit Criteria
Test Entry Criteria: All test hardware platforms must have been successfully installed , configured, and functioning properly. Proper test data is available. **Test Exit Criteria:** All Test plans have been run. A certain level of requirements coverage has been achieved.

12 USER ACCEPTANCE TESTING

Test risks and Issues
There is always the risk that the user will not like the finished product. We will be attempting to mitigate this as much as possible by allowing him to have the last word on our designs.
Items to be tested
Print entire document Test to see if a document is being printed correctly **Print specific page** Test to see if the specified page is being printed correctly **Application UI** Test to see if the application supports all database requesting needs **Data Entry Form** Test to see if the data is able to be saved, edited and deleted by the client.
Test Approaches
User acceptance testing would be approached by having the client use a prototype of the software application. The client would attempt to perform the following tasks using placeholder data. ● Enter data for an inspection and report and enter it into the application through the data entry form ● Edit a preexisting inspection and report through the data entry form ● Delete a preexisting inspection and report through the data entry form ● Printing an entire document ● Printing a specific document page ● Clicking on a well on the map ● Clicking on a septic tank on the map
Test Pass/Fail Criteria
Print a document PASS: The document is printed successfully FAIL: The document is not printed or is printed in an unexpected format

Print a specific page
PASS:
The specified page is printed successfully
FAIL:
Either the specified page is not printed or nothing is printed

Data Saving
PASS:
Data is saved to the database successfully without the client having experienced any issues.
FAIL:
Data is not saved in the database
The client is not able to determine how to enter data
The data saving functionality does not meet the client’s actual needs

Data Deleting
PASS:
Data is removed from the database successfully
FAIL:
Data is not removed from the database
The client is not able to determine how to delete data
The data deletion functionality does not meet the client’s actual needs

Data Editing
PASS:
Application updates the data in the database successfully
FAIL:
Application does not update the data in the database
The client is not able to determine how to edit data
The data editing functionality does not meet the client’s actual needs

Test Entry/Exit Criteria

Test Entry Criteria:

All test hardware platforms must have been successfully installed , configured, and functioning properly.

Proper test data is available.

Client is available to test prototype.

Test Exit Criteria:

All Test plans have been run.

A certain level of requirements coverage has been achieved.