A WEB-BASED SYSTEM FOR ASSESSING BRIDGE INVENTORY COMPLIANCE
BASED ON THE NATIONAL BRIDGE INSPECTION STANDARDS

Yang Zhu
B.S., California State University, Sacramento, 2009

PROJECT

Submitted in partial satisfaction of
the requirements for the degree of

MASTER OF SCIENCE

in

SOFTWARE ENGINEERING

at

CALIFORNIA STATE UNIVERSITY, SACRAMENTO

FALL
2011
A WEB-BASED SYSTEM FOR ASSESSING BRIDGE INVENTORY COMPLIANCE
BASED ON THE NATIONAL BRIDGE INSPECTION STANDARDS

A Project

by

Yang Zhu

Approved by:

__________________________________, Committee Chair
Ahmed Salem, Ph.D.

__________________________________, Second Reader
V. Scott Gordon, Ph.D.

Date
Student: Yang Zhu

I certify that this student has met the requirements for format contained in the University format manual, and that this project is suitable for shelving in the Library and credit is to be awarded for the Project.

__________________________, Graduate Coordinator
NikrouzFaroughi, Ph.D.

______________
Date

Department of Computer Science
Abstract

of

A WEB-BASED SYSTEM FOR ASSESSING BRIDGE INVENTORY COMPLIANCE BASED ON THE NATIONAL BRIDGE INSPECTION STANDARDS

by

Yang Zhu

In recent years, recognizing the importance of collecting and analyzing highway infrastructure data, the Federal Highway Administration (FHWA) launched an effort to develop a web-based system for collecting data on compliance review for the National Bridge Inventory (NBI).

The intent of this project is to help the FHWA provide oversight to State Department of Transportation (DOT) to more efficiently perform their bridge inventory assessment and to help them reduce any possible human error related to random sampling and/or assessment record management.

The web-based system has four main features:

- Parse NBI Data
- Perform Compliance Assessment
- Generate an Assessment Report
- Maintain Compliance Assessment History
First, the system parses encoded data from NBI and stores the decoded data into a MySQL database. Second, the system provides functionalities to enable the users to perform compliance assessment, including form filling and file uploading. Third, the users are able to generate assessment report to serve their various needs. Finally, the system maintains compliance assessment history so the users can get back to any review sessions at any given time. Other features are built and made available to the users to support the main features.

The system is built on a Windows version of the XAMPP stack, “an easy to install Apache distribution containing Apache, MySQL, PHP and Perl.”[1] It is deployed on a removable computer storage device (e.g. a typical USB drive). Since the users of the system potentially will have many travels to bridge sites, the main advantage of having this deployment method is that it enables the users to take the system with them to their site visits.

_______________________, Committee Chair
Ahmed Salem, Ph.D.

_______________________
Date
ACKNOWLEDGMENTS

I would like to take this moment to express my appreciation to people who have provided support and/or to who have affected me from the beginning of this project. Without their continuous influence, the project would not have been successful.

First, I would like to thank my project advisor Dr. Ahmed Salem for giving me advice based on his expertise in the area of software engineering. I also appreciate the creativity resulting from the freedom Dr. Ahmed Salem has been giving.

I would also like to thank my second reader Dr. V. Scott Gorgon for his initiative and advices on the project’s database design. I heavily value his profession in database.

Mr. Loren Turner, a Senior Transportation Engineer from California Department of Transportation, has been very much helpful throughout the development of the project, continuously giving me insight and envision on things related to the project. Without his help, this project might have gone in wrong directions.

I would also thank Sarah Skeen and Bill Forrester, my mentor and director at Federal Highway Administration respectively for their freedom so that I can go free flow with my creativity.

Past experience from former students, especially Xing H. Liu and Peng Zhang, has also been helpful.

Finally, I am sincerely and deeply thankful for my parents’ understanding and support on the long journey of the peruse of my master’s degree. They deserve much more credit than I can ever give.
TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acknowledgments</td>
</tr>
<tr>
<td>List of Tables</td>
</tr>
<tr>
<td>List of Figures</td>
</tr>
</tbody>
</table>

Chapter

1. INTRODUCTION
   1.1 Background and Related Work | 1 |
   1.2 NBI History and Current Issues | 3 |
   1.3 Purpose and Scope | 4 |

2. TECHNOLOGY
   2.1 AJAX | 7 |
   2.2 Apache Http Server | 7 |
   2.3 Google Maps API | 8 |
   2.4 jQuery | 8 |
   2.5 JSON | 8 |
   2.6 MySQL | 8 |
   2.7 PHP | 9 |
   2.8 XAMPP | 9 |
   2.9 How Technologies are Used | 9 |

3. FUNCTIONAL REQUIREMENTS | 11 |
   3.1 Functional Requirements | 12 |
3.2 Scenarios ............................................................................................................... 16

4. SYSTEM DESIGN ..................................................................................................... 17
  4.1 Database Design ................................................................................................ 18
  4.2 Data Dictionary .................................................................................................. 19
  4.3 System Sequence Diagrams .............................................................................. 26
  4.4 User Interface Design ....................................................................................... 31

5. SYSTEM IMPLEMENTATION .............................................................................. 36
  5.1 Site Map ............................................................................................................. 36
  5.2 System User Interface ....................................................................................... 37
  5.3 The Request_Handler Directory ..................................................................... 49
  5.4 Other Directories .............................................................................................. 52

6. CONCLUSION AND FUTURE WORK ................................................................. 54
  6.1 Conclusions ....................................................................................................... 54
  6.2 Challenges and Lessons Learned ................................................................... 55
  6.3 Future Work ...................................................................................................... 56

Appendix A. Random Sampling Procedure .............................................................. 58
  A.1 The Procedure .................................................................................................. 58
  A.2 Example ............................................................................................................ 59

Appendix B. Database Design ................................................................................... 61

Appendix C. Database Implementation .................................................................... 62

Appendix D. Detail Bridge Data Sheet ..................................................................... 63

Appendix E. Application Stack .................................................................................. 64
E.1 A List Of Needed Items

Bibliography
# LIST OF TABLES

<table>
<thead>
<tr>
<th></th>
<th>Table Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Table 4.1 Review Session</td>
<td>19</td>
</tr>
<tr>
<td>2.</td>
<td>Table 4.2 Nbi File</td>
<td>20</td>
</tr>
<tr>
<td>3.</td>
<td>Table 4.3 Review Overview</td>
<td>21</td>
</tr>
<tr>
<td>4.</td>
<td>Table 4.4 Metric</td>
<td>22</td>
</tr>
<tr>
<td>5.</td>
<td>Table 4.5 Metric Snapshot</td>
<td>22</td>
</tr>
<tr>
<td>6.</td>
<td>Table 4.6 Metric Review Question</td>
<td>23</td>
</tr>
<tr>
<td>7.</td>
<td>Table 4.7 Metric Review Question</td>
<td>23</td>
</tr>
<tr>
<td>8.</td>
<td>Table 4.8 Master List</td>
<td>24</td>
</tr>
<tr>
<td>9.</td>
<td>Table 4.9 Locked Samples</td>
<td>25</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>1.</td>
<td>Figure 3.1 Use Case Diagram</td>
<td>11</td>
</tr>
<tr>
<td>2.</td>
<td>Figure 4.1 System Design</td>
<td>17</td>
</tr>
<tr>
<td>3.</td>
<td>Figure 4.2 System Sequence Diagram</td>
<td>27</td>
</tr>
<tr>
<td>4.</td>
<td>Figure 4.3 NBI File Upload Sequence Diagram</td>
<td>28</td>
</tr>
<tr>
<td>5.</td>
<td>Figure 4.4 Session Creation Diagram</td>
<td>29</td>
</tr>
<tr>
<td>6.</td>
<td>Figure 4.5 Sample Assessment Sequence Diagram</td>
<td>30</td>
</tr>
<tr>
<td>7.</td>
<td>Figure 4.6 Generate Report Sequence Diagram</td>
<td>31</td>
</tr>
<tr>
<td>8.</td>
<td>Figure 4.7 Template Design</td>
<td>32</td>
</tr>
<tr>
<td>9.</td>
<td>Figure 4.8 Main Content Designs</td>
<td>33</td>
</tr>
<tr>
<td>10.</td>
<td>Figure 4.9 Assessment Level Change Form Design</td>
<td>34</td>
</tr>
<tr>
<td>11.</td>
<td>Figure 4.10 Random Sample Assessment Data Form</td>
<td>35</td>
</tr>
<tr>
<td>12.</td>
<td>Figure 5.1 System Implementation Site Map</td>
<td>36</td>
</tr>
<tr>
<td>13.</td>
<td>Figure 5.2 NBI File Uploader Interface</td>
<td>37</td>
</tr>
<tr>
<td>14.</td>
<td>Figure 5.3 Session Creation Interface A</td>
<td>38</td>
</tr>
<tr>
<td>15.</td>
<td>Figure 5.4 Session Creation Interface B</td>
<td>39</td>
</tr>
<tr>
<td>16.</td>
<td>Figure 5.5 Session Overview Interface</td>
<td>40</td>
</tr>
<tr>
<td>17.</td>
<td>Figure 5.6 Sample Assessment Interface</td>
<td>41</td>
</tr>
<tr>
<td>18.</td>
<td>Figure 5.7 Assessment Level Change Form Interface</td>
<td>42</td>
</tr>
<tr>
<td>19.</td>
<td>Figure 5.8 Assessment Data Form Interface A</td>
<td>43</td>
</tr>
<tr>
<td>20.</td>
<td>Figure 5.9 Assessment Data Form Interface B</td>
<td>44</td>
</tr>
</tbody>
</table>
21. Figure 5.10 Assessment Data Form Interface C ......................................................... 44
22. Figure 5.11 Determination Form Interface A ............................................................. 45
23. Figure 5.12 Determination Form Interface B ............................................................. 46
24. Figure 5.13 Determination Form Interface C ............................................................. 46
25. Figure 5.14 Determination Form Interface D ............................................................. 47
26. Figure 5.15 Tab Navigation on Report Page .............................................................. 48
27. Figure 5.16 Observations on Assessment Report ....................................................... 48
28. Figure B.1 Database Entity Relationship Diagram - Design ................................... 61
29. Figure C.1 Database Entity Relationship Diagram - Implementation .................... 62
30. Figure D.1 Detailed Bridge Data Sheet ................................................................. 63
Chapter 1

INTRODUCTION

This project is a web-based system for performing compliance review on the National Bridge Inventory (NBI) based on the National Bridge Inspection Standards (NBIS). The main focus of the project is to enable officials from Federal Highway Administration (FHWA) to perform compliance assessment using a computer with a web browser.

1.1 Background and Related Work

When Silver Bridge in Point Pleasant, West Virginia, collapsed and killed 46 people on December 15, 1967, federal government initiated a program to ensure safety through bridge inspection and evaluation. During the same year, to ensure the safety of the public in regard of federal highways, the federal government established the Federal Highway Administration (FHWA) based on the National Bridge Inspection Standards (NBIS) and the NBIS’s 23 metrics established by the Secretary of Transportation [2]. Each of the 23 metrics has its own population among the bridges within the states, which is defined in the Metric Population Criteria Document [3].

As a result, State Departments of Transportation (DOT) are required to perform biannual bridge inspections on all public bridges within the state. In California, every aspect of the bridge inspection has to follow exactly as described in the Bridge Inspector’s Reference Manual [4].
Even though the states are required to perform bridge inspections, there is still a need to verify that the bridge inspection is performed according to standards established through the NBIS. As the Minneapolis Bridge, which had been reported safe, collapsed in 2007 at age 40, the need to ensure bridge inspections are performed accordingly caught the public media’s attention [5].

The California Division of the FHWA is responsible for insuring that the state properly inspects the bridges located in California and that the bridges and bridge inspection records are in order. This bridge inspection insurance process is referred to as a periodical process, Compliance Review. Within each Compliance Review cycle, bridge inspection data and reports are reviewed based on a criteria comprised of 23 metrics defined in the NBIS to determine the bridge inventory’s compliance level. Follow-up actions and reports are conducted at the end of Compliance Review cycles [6].

As of now, the 23 metrics are still under modification. Because of that, there are no similar products developed. The only product that relates to this project is an Excelfile which takes the NBI data as input and helps its users to randomly sample a set of bridge for compliance review. This is not very helpful since the users have to perform compliance review and record keeping on their own.
1.2 NBI History and Current Issues

Currently, bridge information is stored in a file format established through the National Bridge Inventory (NBI), and includes various design and construction parameters, such as bridge type, materials, span lengths, year of construction, etc. The bridge data is encoded by the State DOTs and submitted to the FHWA. The format was originally specified in a way to achieve disc space efficiency because memory was very expensive at the time the inspection program was initiated. Bridge data is stored in an ASCII text file containing a fixed length of alpha-numeric values; one bridge data is literally one string with a fixed size of 432 characters without punctuations. The form of the encoded data makes it difficult to view and understand the bridge data, even for those who have worked with the NBI for quite a while. Reading the data in the NBI is almost like reading a binary string and trying to translate the value into human-readable string.

Furthermore, there are more than 24,000 bridges in California. It would be impossible to determine the compliance level on and every bridge across California for each of the metrics within reasonable budget and schedule. So, the bridge inventory compliance assessment has to be based on random sampling to achieve a certain confidence level and to maintain a small marginal error in the results. However, there are problems when people try to manually select random samples. It is difficult to come up with truly random samples. Moreover, it is harder to obtain the selected samples if the inspector
had to come back on the inspection for a second time. In addition, manually selection can be error-prone and biased, which is against the purpose of random sampling.

1.3 Purpose and Scope

The purpose of the project is to develop a web-application to resolve the problems described in the prior section. The NBIS Metric Document defines 23 metrics. However, this project’s scope is limited to the current version of metrics 6 through 23. Metrics 1 through 5 are excluded from this project because these metrics are not directly related to the data store in the NBI files. Future versions of the 23 metrics will be supported by future updates to the project.

The data in the NBI is currently encoded with codes to achieve disc space efficiency. Bridge inspectors have to have a way to look at the decoded data. Below is one sample bridge data stored in the NBI. It tells everything about a bridge, including but not limited to the state the bridges is in, the exact location, inspected dates, inspection data, bridge types, bridge size information, and even costs related to the bridge.

069 1CA0070166000000711158296MUGU LAGOON LAGUNA ROAD NBVC POINT MUGU 999900000000
34060810119061488000373730919490200000020020094010000000000NN115N0000000000A1510400001
0000135001040000400000000013501729999N0000N0000007778N2247218159N5883110010407092
4N Y48N 0709 0000250000030000282009 0N2 0000002688020YT0002502026 NY 0 0814

It is not difficult to see it is not helpful at all for a typical person. If a person is asked to identify which state this bridge is located using the data above, the person has to refer to
the NBI Coding Guide to locate where the state data is within the bridge data string. Even though after reading the NBI Coding Guide and find out that the first 3 characters of the data represent state code, it is still necessary to look up the three-digit code in order to find out which state the code represents. This is also true for District Code, County Code and Place Code being encode in this line of bridge data.

So, part of the purpose of this web-application is to decode the encoded data back to its original form and to represent the decoded data in a readable form back to the users. As a result, anyone who is interested in the NBI bridge data could review bridge records with ease, without referring back to the coding guide for the NBI, and without looking up coding references for specific records for the encoded data. This would essentially increase the productivity in bridge inventory assessment for the same reason.

In order to counter for human resource, time, and budget problems, FHWA has to perform random sampling on the bridge population due to the large set of data. The second part of the purpose of this web-application is to automate the random sampling process for bridge inventory assessment to reduce the human error involves with manually random sampling.

Moreover, this web-application provides a way for the FHWA agencies to review the selected random samples. In addition, this web-application also keeps track of compliance review data for the Federal Agencies to review. In short, this web-
application is a place where federal staff could perform a complete compliance review for the NBI.

In addition to the original intent of this project, this web-application could potentially form the basis for future application development to provide meaningful information to the public. Fact is that the public knows much less about the bridge inventory since it is encoded. Perhaps, the news media is the only one that knows much about national bridges. When major events occur that involve highway bridges, the public media looks at the NBI and comes up with interesting interpretations that no typical person now could verify. It could be nice if a typical person could understand the data in the NBI, verify the interpretation done by public media, and even come up with the person’s own interpretation. Going through a bridge is just part of a very day life. It is not hard to see some people may want to know more about the bridges they are crossing. This project can be the base for providing bridge information to just the personal interest of those people.
Chapter 2

TECHNOLOGY

This section describes the technologies that are used to implement the system. One of the underlying benefit or purpose of this project is to reduce the overall cost. Therefore, an open source framework was the primary choice for the development and deployment for the project. As it turns out, this project can be totally developed and deployed on an open source framework, a XAMPP stack, a cross-platform stack that packages Apache HTTP Server, MySQL database, PHP and Perl. JavaScript, JSON, GoogleMap API, and jQuery are also open source and are utilized in the project.

2.1 AJAX

AJAX stands for Asynchronous JavaScript and XML. Web-applications use AJAX to asynchronously talks to a server and update some sections of a page based on server response without refreshing the entire page.

2.2 Apache HTTP Server

Apache HTTP Server is a web server developed and maintained by the Apache Software Foundation. The primary function of a web server is to response to users’ requests from a client, typically a web browser, with requested information. This apache server sits on the deployment removable device for the purpose of this project.
2.3 Google Maps API

Google Maps is a mapping technology freely provided by Google. Using Google Maps API, web developers could geospatially map their data onto the Google Maps, making data visually available to the users of the data. In this project, randomly sampled bridges are displayed on the Google Maps for testing and referencing purposes.

2.4 jQuery

jQuery is a browser-independent JavaScript library, developed with the purpose of simplifying front-end web development. AJAX is fully integrated in the library.

2.5 JSON

JSON stands for JavaScript Object Notation; it is a “lightweight text-based open standard designed for human-readable data interchange [7].” JSON is used in this project to communicate with the web server, the Apache HTTP server.

2.6 MySQL

MySQL is an open-source relational database management system [8], which stores and manages data [9]. It mainly stores the parsed NBI data and compliance assessment data in this project.
2.7 PHP

PHP, a server-side scripting language, originally stands for Personal Home Page, and then changed to Hypertext Preprocessor [10]. It is used in this project as a server-side scripting language to generate dynamic web pages and to provide server-side functionalities.

2.8 XAMPP

XAMPP stands for Cross-platform Apache, MySQL, PHP, and Perl. It is a complete and easy-to-setup package for web development. Although XAMPP’s preconfigured setting is not as secured and is not recommended for web application deployment, the setting is suitable for this project since people outside of FHWA do not have access to the project.

2.9 How Technologies are Used

Since XAMPP is an easy-to-setup package for web development, it is used to set up the basic requirements for the project, which includes the Apache web server, PHP server-side scripting language, and MySQL relational database management system. The Apache web server is used for responding and handling the communication between the users from the front-end and the services provided from the back-end. PHP is responsible for manipulating data and outputting requested information from/to the MySQL databases.
Generally, AJAX is used to increase performance and user experience. Upon completion of a typical HTTP, a page is refreshed to reflect the changes or the user is redirected to a different page. With the use of AJAX, that is no longer the case. AJAX completes its HTTP request and only updates the users’ interested parts of the page. One example of this in the project is the Create Session feature. During session creation, when a user select a state from the dropdown selection box, the page sends out an AJAX request to the server to retrieve existing counties for the state. Then, upon server’s response, only the county dropdown selection box is updated, and rest of the page remains the same.

One way a server returns data to a client is formatting the returning data into JSON format. Most of the HTTP requests in this project returns JSON format data. In the example above, counties for the selected state is returned as JSON.

jQuery, the JavaScript querying library, is included in pages that need it. It makes it easy to implement AJAX and to communicate with JSON data. jQuery also makes it much more easier to access elements within a web page document. A comprehensive documentation can be found on [http://docs.jquery.com/Main_Page](http://docs.jquery.com/Main_Page).

Google Maps is used to display bridges on a map. Through Google Maps API, Google Maps can be embedded into a web page. Then, bridges can be displayed on the Google Maps using their geospatial information in the NBI. Google Maps API documentation can be found on [http://code.google.com/apis/maps/](http://code.google.com/apis/maps/)
Chapter 3

FUNCTIONAL REQUIREMENTS

There are five main functional requirements for this web-based NBI system, and they are captured in the following use case diagram.

![Use Case Diagram](image)

**Figure 3.1 Use Case Diagram**

There are two types of users, NBI Reviewer and Administrators. Bridge Inspectors use four out of the five main functional requirements, Creating New Session, Resume Existing Sessions, Assess Compliance, and Generate Report. Therefore, Bridge Inspectors are the primary users of the system. Administrators use the system to generate necessary reports upon requests, by the public, news media, and/or their own interest. Administrators also use the system to upload NBI files. Because the system functions
based on the presence of NBI data, the administrators have to upload NBI file to the system before NBI review can start any sessions.

Even though functionalities of the system can be grouped based on their user types, this version of the system does not maintain and does not distinguish user types. Every user that has access to the system can utilize any of the functionality.

3.1 Functional Requirements

This section describes each of the function requirements in details, use case by use case.

UC1: Create Review Session

Description: The system maintains a history of compliance assessment using sessions. The NBI Reviewers is provided with a form for inputting session meta data, including boundary of the area, the year, the assessment level, and a description for the session. Creating a new review session includes three smaller functional requirements, Load NBI Data, Randomly Sample, and Determine Preliminary Result.

UC1.1: Load NBI Data

Description: NBI Data is loaded into this web-based NBI system’s database by uploading NBI file in text format. At the time or creating a new review session, relevant data is loaded into memory to determine population sizes for the 23 metrics. Sizes for random sampling are based on the population sizes.
UC1.2 Randomly Sample
Description: Random sampling is the core of this project. A specific method is used to do random sampling. Detailed description is provided in Appendix A.

UC1.3 Determine Preliminary Result
Description: Result for some of the metrics, in theory, can be determined based on the uploaded NBI file. A compliance assessment overview page containing preliminary result is developed. However, the federal government may not want to see any assessment being automatically generated by computer. Therefore, this functional requirement might be excluded in the final version of the project.

UC2: Resume Existing Sessions
Description: This functionality provides users with the ability to resume sessions that are created previously, either in-progress or completed session, to continue performing compliance assessments or to generate report.

UC2.1: Load Existing Data
Description: Whether a session is in-progress or completed, this functionality loads relevant data related to the session upon request. Without this function, users have to start the sessions all over again.
UC3: Assess Compliance

Description: For the users, this functional requirement is the most important piece of this project. The users have to have the ability to input data to the system to help determine compliance. This functional requirement is extended so that the users can upload relevant documents to support their points of view. This functional is also extended so that the users can change assessment for any specific metrics.

UC3.1: Input Data

Description: The users input data to determine compliance for the metrics on a uniformly presented data form. However, different metrics requiring different sets of data is making this requirement a challenging one.

UC3.2: Upload Documents

Description: This functionality provides users with the ability to upload files on to the system. Files can be retrieved as needed. Not all of the 23 metrics requires this functionality, but it is provided to all of the 23 metrics for the users’ convenience.

UC3.3: Change Assessment Level

Description: Assessment level determines the number of bridges to be assessed for a session. Using this function, users can specify a higher or lower assessment level other than the default session level selected during the session creation.
UC4: Generate Report
Description: Generate Report allows users to generate report for any metrics for any sessions.

UC5: Maintain NBI Files
Description: This function allows administrator maintain NBI files. However, this version of the system is only able to upload NBI files, and users cannot replace uploaded NBI files.

UC5.1: Upload NBI Files
Description: Administrators use this function to load NBI files in text format, as the basis data, onto this web-based NBI system’s MySQL database. The system reads in the NBI file, parse the file, and populate the system’s data based on the data within the file.

UC5.2: Replace NBI Files
Description: At times, NBI files are updated. This function allows administrators to replace outdated NBI files with updated NBI file. Due to the complexity and the implication of this functionality, this functionality might not be implemented in the first version of the system.
3.2 Scenarios

This section describes some typical usage scenarios of the system.

SC1: Create Review Session (use case: UC1)

- Open Session Login page
- Switch to the Create New Session tab on the page
- Fill in necessary data
- Submit the form and wait for the server to process
- Redirect to Assessment Overview page

SC2: Assess Compliance (use case: UC3)

- Open Assess Samples page from Assessment Overview page
- Click on Change Assessment Level to change assessment level
- Change assessment level
- Click the view button to view detailed bridge data
- Close detailed bridge page
- Click the power button to bring up the assessment forms for the associated bridges
- Fill out all of the forms for the randomly sampled bridges
- Click Assess Compliance to assess the compliance for the metric
- Redirect to metric report page
Chapter 4

SYSTEM DESIGN

The system is designed so that the whole system works on external hard drive. As long as the users have the access to the hard drive and a computer with browsers, the users can use the system. The reason that the system is not hosted on a server is that the system is only used by employees within the FHWA California Division office. However, if the system gets attention nationally, the system might at least be assessable from within the FHWA nationally. Following is the overall system design.

![System Design Diagram]

Figure 4.1 System Design
Since this system is designed to be a web 2.0 application, in minimum it contains a web server, a file server, and a database server [11]. In addition, because of the system’s original intent, the entire system is sustained within a removable external hard drive, where the web server, file server, and database server are hosted. Once the external hard drive is connected to a computer, the whole system is totally accessible from within the computer. When uploading NBI files, users of the system use the system’s interface to select NBI text files that sits on the users’ computer. The web server parses the NBI text file and populates the system’s database through the communication between the web server and the database server. During compliance assessment, attached documents are stored on the file server through the communication between the web server and the file server.

In this version of the project, users have to manually obtain needed bridge data file through the internet or some other ways. The option to automatically retrieving bridge data file within the system, for data validation purpose, is left open. However, in any event, this function is not required and is not included in the scope of this master’s project.

4.1 Database Design

The database for this project is a relational database using MySQL. The database relationship design is available at Appendix B, and the actual database implementation relationship diagram is available at Appendix C.
4.2 Data Dictionary

This section describes the data being stored in the system’s MySQL database.

4.2.1: Table review_session

This table maintains session meta data for identifying compliance assessment sessions.

<table>
<thead>
<tr>
<th>Column</th>
<th>Data Type</th>
<th>Null</th>
<th>Default</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>int(11)</td>
<td>No</td>
<td></td>
<td>Session unique identifier</td>
</tr>
<tr>
<td>year</td>
<td>varchar(4)</td>
<td>No</td>
<td></td>
<td>Assessment session year</td>
</tr>
<tr>
<td>State</td>
<td>varchar(2)</td>
<td>No</td>
<td></td>
<td>Assessment state</td>
</tr>
<tr>
<td>county</td>
<td>varchar(100)</td>
<td>No</td>
<td></td>
<td>County within the assessment state</td>
</tr>
<tr>
<td>created_by</td>
<td>varchar(11)</td>
<td>Yes</td>
<td></td>
<td>Person who created the assessment session</td>
</tr>
<tr>
<td>assessment_level</td>
<td>varchar(50)</td>
<td>No</td>
<td></td>
<td>Assessment level</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• 1 – Intermediate tier 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• 2 – Intermediate tier 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• 3 – In-depth tier 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• 4 – In-depth tier 2</td>
</tr>
<tr>
<td>label_name</td>
<td>varchar(50)</td>
<td>No</td>
<td></td>
<td>Label of the Session</td>
</tr>
<tr>
<td>description</td>
<td>varchar(200)</td>
<td>No</td>
<td></td>
<td>Detailed Description of the Session</td>
</tr>
<tr>
<td>nbi_file_id</td>
<td>int(11)</td>
<td>No</td>
<td></td>
<td>Foreign key to nbi file id in nbi file table</td>
</tr>
</tbody>
</table>
4.2.2: Table nbi_file

This table maintains nbi_filemeta data for the nbi bridge data being loaded into the system’s database.

Table 4.2 NBI File

<table>
<thead>
<tr>
<th>Column</th>
<th>Data Type</th>
<th>Null</th>
<th>Default</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>int(11)</td>
<td>No</td>
<td></td>
<td>Nbi file unique identifier</td>
</tr>
<tr>
<td>year</td>
<td>varchar(4)</td>
<td>No</td>
<td></td>
<td>Year of the NBI file</td>
</tr>
<tr>
<td>state</td>
<td>varchar(2)</td>
<td>No</td>
<td></td>
<td>State of Which the NBI belongs to</td>
</tr>
<tr>
<td>size</td>
<td>varchar(20)</td>
<td>No</td>
<td></td>
<td>No File Size</td>
</tr>
<tr>
<td>upload_date</td>
<td>datetime</td>
<td>No</td>
<td></td>
<td>Date of the NBI File being uploaded</td>
</tr>
<tr>
<td>ref_link</td>
<td>varchar(200)</td>
<td>No</td>
<td></td>
<td>A URL reference to the NBI file sitting on the server</td>
</tr>
</tbody>
</table>

4.2.3: Table raw_data

This table stores bridge and structure data, and the table has more than a hundred of columns. The columns of the table and the corresponding data for the columns follow exactly as described in the NBI Coding Guide. Please refer to the coding guide for the carefully explained data dictionary.

At the time of session creation, corresponding raw data is identified for the session.

Sessions cannot be created without the existence of the raw data. This is also the data basis for the population and random sampling of compliance assessment.
4.2.4: Table review_overview

This table stores an assessment’s overview for the 23 metrics. Each session record has a review overview record for each of the 23 metrics. Combining overview of the 23 metrics gives the overview of an assessment session.

Table 4.3 Review Overview

<table>
<thead>
<tr>
<th>Column</th>
<th>Data Type</th>
<th>Null</th>
<th>Default</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>int(11)</td>
<td>No</td>
<td></td>
<td>Assessment Overview Unique Identifier - Metric Based</td>
</tr>
<tr>
<td>metric_id</td>
<td>tinyint(2)</td>
<td>No</td>
<td></td>
<td>Metric Identifier, foreign key to metric table</td>
</tr>
<tr>
<td>population_size</td>
<td>int(11)</td>
<td>No</td>
<td>0</td>
<td>Population Size for The Corresponding Metric</td>
</tr>
<tr>
<td>size_1</td>
<td>int(11)</td>
<td>No</td>
<td>0</td>
<td>Random Sample Size for Intermediate Tier 1</td>
</tr>
<tr>
<td>size_2</td>
<td>int(11)</td>
<td>No</td>
<td>0</td>
<td>Random Sample Size for Intermediate Tier 2</td>
</tr>
<tr>
<td>size_3</td>
<td>int(11)</td>
<td>No</td>
<td>0</td>
<td>Random Sample Size for In-depth Tier 1</td>
</tr>
<tr>
<td>size_4</td>
<td>int(11)</td>
<td>No</td>
<td>0</td>
<td>Random Sample Size for In-depth Tier 2</td>
</tr>
<tr>
<td>review_type</td>
<td>int(1)</td>
<td>No</td>
<td>0</td>
<td>Assessment Level Indicator</td>
</tr>
<tr>
<td>session_id</td>
<td>int(11)</td>
<td>No</td>
<td></td>
<td>Foreign Key to review_session table</td>
</tr>
<tr>
<td>determinatio_n_status</td>
<td>int(1)</td>
<td>No</td>
<td>0</td>
<td>Field for storing determination status</td>
</tr>
<tr>
<td>schedule_st atus</td>
<td>int(1)</td>
<td>No</td>
<td>1</td>
<td>Schedule status for review</td>
</tr>
<tr>
<td>action</td>
<td>longtext</td>
<td>No</td>
<td></td>
<td>Action to be taken for the current review</td>
</tr>
</tbody>
</table>
4.2.5: Table metric

This table maintains the 23 metrics definition and information.

Table 4.4 Metric

<table>
<thead>
<tr>
<th>Column</th>
<th>Data Type</th>
<th>Null</th>
<th>Default</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>tinyint(2)</td>
<td>No</td>
<td></td>
<td>Metric Unique Identifier</td>
</tr>
<tr>
<td>ref_code</td>
<td>varchar(20)</td>
<td>No</td>
<td></td>
<td>Reference code</td>
</tr>
<tr>
<td>label</td>
<td>varchar(200)</td>
<td>No</td>
<td></td>
<td>Label for the Metric</td>
</tr>
<tr>
<td>q_ref_code</td>
<td>varchar(100)</td>
<td>No</td>
<td></td>
<td>Question reference code</td>
</tr>
<tr>
<td>full_question</td>
<td>varchar(1000)</td>
<td>No</td>
<td></td>
<td>Full Questions Full Metric Criteria Questions as described in the metric document</td>
</tr>
<tr>
<td>sub_def</td>
<td>varchar(1000)</td>
<td>No</td>
<td></td>
<td>Substantial compliance definition</td>
</tr>
<tr>
<td>non_def</td>
<td>varchar(1000)</td>
<td></td>
<td></td>
<td>Non-compliance definition</td>
</tr>
</tbody>
</table>

4.2.6: Table metric_snapshot

This table maintains snapshots of a metric. There is no upper limits on the number of snapshots a metric can have. However, generally, snapshots are just quarterly statements for a metric, and so there are usually 4 snapshots for a metric at the very end of a review session.

Table 4.5 Metric Snapshot

<table>
<thead>
<tr>
<th>Column</th>
<th>Data Type</th>
<th>Null</th>
<th>Default</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>tinyint(2)</td>
<td>No</td>
<td></td>
<td>Unique Identifier for a snapshot</td>
</tr>
<tr>
<td>review_id</td>
<td>int(11)</td>
<td>No</td>
<td></td>
<td>Foreign Key to table review_session</td>
</tr>
</tbody>
</table>
4.2.7: Table metric_review_questions

This table maintains some metric-based review questions. Metrics could have zero or more metric-based questions in this table.

Table 4.6 Metric Review Question

<table>
<thead>
<tr>
<th>Column</th>
<th>Data Type</th>
<th>Null</th>
<th>Default</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Int(11)</td>
<td>No</td>
<td></td>
<td>Unique Identifier for a question</td>
</tr>
<tr>
<td>question</td>
<td>text</td>
<td>No</td>
<td></td>
<td>Complete standard English question is stored here..</td>
</tr>
<tr>
<td>metric</td>
<td>tinyint(2)</td>
<td>No</td>
<td></td>
<td>Of which metric the question is used for. Foreign key to metric table</td>
</tr>
<tr>
<td>question_ordering</td>
<td>int(11)</td>
<td>No</td>
<td></td>
<td>Order of the question being display on compliance assessment form and report.</td>
</tr>
</tbody>
</table>

4.2.8: Table metric_review_answers

This table maintains answers for metric-based review questions.

Table 4.7 Metric Review Question

<table>
<thead>
<tr>
<th>Column</th>
<th>Data Type</th>
<th>Null</th>
<th>Default</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Int(11)</td>
<td>No</td>
<td></td>
<td>Unique Identifier for a question</td>
</tr>
</tbody>
</table>
24

<table>
<thead>
<tr>
<th>Column</th>
<th>Data Type</th>
<th>Null</th>
<th>Default</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>int(11)</td>
<td>No</td>
<td></td>
<td>Unique Identifier</td>
</tr>
<tr>
<td>bridge_id</td>
<td>int(11)</td>
<td>No</td>
<td></td>
<td>Foreign Key to the bridge record</td>
</tr>
<tr>
<td>session_id</td>
<td>int(11)</td>
<td>No</td>
<td></td>
<td>Foreign Key to the session of which the master list is for</td>
</tr>
<tr>
<td>random_number</td>
<td>int(11)</td>
<td>No</td>
<td></td>
<td>The random number assigned when creating session</td>
</tr>
</tbody>
</table>

4.2.9: Table master_list

At the time a session creation, a master_list is built based on random sampling on the nbi data. Each session has one master_list. The master_list is also used to change assessment level for metrics.

Table 4.8 Master List

4.2.10: Table locked_samples

At the time a session creation, randomly selected samples are stored in this table.

Bridges IDs are used to reference data in table raw_data.
Table 4.9 Locked Samples

<table>
<thead>
<tr>
<th>Column</th>
<th>Data Type</th>
<th>Null</th>
<th>Default</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>int(11)</td>
<td>No</td>
<td></td>
<td>Locked Sample Unique Identifier</td>
</tr>
<tr>
<td>metric_id</td>
<td>int(2)</td>
<td>No</td>
<td></td>
<td>The Metric ID that the Locked Sample is Associated with</td>
</tr>
<tr>
<td>bridge_id</td>
<td>int(11)</td>
<td>No</td>
<td></td>
<td>The Associated Bridge ID, foreign key to table raw_data</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• raw_data -&gt; bridge_id</td>
</tr>
<tr>
<td>comment</td>
<td>varchar(300)</td>
<td>No</td>
<td>No</td>
<td>Additional comments or notes for the locked sample</td>
</tr>
<tr>
<td>review_id</td>
<td>int(11)</td>
<td>No</td>
<td></td>
<td>Foreign Key to Overview ID</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>• review_overview -&gt; id</td>
</tr>
<tr>
<td>is_reviewed</td>
<td>tinyint(1)</td>
<td>No</td>
<td>0</td>
<td>Sample Review Status Indicator; 0 for no, 1 for yes.</td>
</tr>
</tbody>
</table>

4.2.11: Locked Sample Data Tables

Each of the metric requires a different set of questions and data in order to determine the nbi’s compliance. Thus, there are 23 locked_sample_data tables, which names are locked_sample_data_ appended with the metric numbers. Each table has an id column as its primary key and a locked_sample_id field as a foreign key to reference locked sample in the locked_sample table. Specific fields unique to the metrics for each of the tables are listed and described below. Again, since metrics 1 through 5 are not in the scope of this project, tables for those metrics are also not included in the scope of this project. They are listed below anyway.

1. Locked_sample_data_1
2. Locked_sample_data_2
3. Locked_sample_data_3
4. Locked_sample_data_4
5. Locked_sample_data_5
6. Locked_sample_data_6
7. Locked_sample_data_7
8. Locked_sample_data_8
9. Locked_sample_data_9
10. Locked_sample_data_10
11. Locked_sample_data_11
12. Locked_sample_data_12
13. Locked_sample_data_13
14. Locked_sample_data_14
15. Locked_sample_data_15
16. Locked_sample_data_16
17. Locked_sample_data_17
18. Locked_sample_data_18
19. Locked_sample_data_19
20. Locked_sample_data_20
21. Locked_sample_data_21
22. Locked_sample_data_22
23. Locked_sample_data_23

4.3 System Sequence Diagrams

This section describes some typical but important flows of the system. Some other relatively smaller flows of the system are not included. The following sequence diagram shows the end-to-end flow of the system in a higher level, followed by lower level sequence diagrams in sub sections.
Detailed sequence on how administrator upload NBI file to the system is described in section 4.3.1. The process of creating a new session is described in section 4.3.2. The sequence diagram for assessing randomly sampled bridges is in section 4.3.3. And finally, the procedural steps on how reports are generated is in section 4.3.4. The database objects in all of these diagrams refer to the system’s internal MySQL database. These sequence diagrams is not meant to be comprehensive of all of the operations in the system. However, the diagrams do provide a good view of the high-level operations.
4.3.1 NBI File Upload

This sequence diagram shows how a NBI file is uploaded to the system. Administrator visits the NBI_file.php page to request a NBI file upload. NBI_file.php submits the upload form to uploadNBIData.php. Then, uploadNBIData.php inserts the NBI file’s meta data into the database, parse the NBI file submitted by the administrator, and update the database while there is more data in the NBI file. In the end, NBI file.php notifies the administrator whether the file upload has been successful.
4.3.2 Session Creation

This diagram shows how NBI Reviewers create compliance review sessions. A NBI Reviewer requests session_log_in.php and chooses to create a new session. The NBI Reviewer fills out the required form for session_log_in.php to validate. Then, create_session.php creates a master list and select samples randomly in parallel. Randomly sampling includes a series of steps; details of the algorithm in described in one of the Appendixes. After a session has been created, the master list is also inserted into
the database. The create_session.php will then redirects the user to review_overview.php, indicating that the session has been created.

4.3.3 Assess Sample

![Sample Assessment Sequence Diagram](image.png)

This sequence shows a typical flow for assessing samples. The NBI Reviewer requests assess_sample.php and sees a list of random samples displayed on both GoogleMap and on a table. The NBI Reviewer reviews, inputs data, and attaches files for each of the random samples as necessary. After all of the interested samples are reviewed, bridge inspector requests to determine compliance for the samples. The bridge inspector is taken to determine_compliance.php as a result of the request. The bridge inspector
reviews a summary of the assessment data and fills out the determination form. The system’s database is updated accordingly.

4.3.4 Generate Report

![Generate Report Sequence Diagram](image)

Figure 4.6 Generate Report Sequence Diagram


4.4 User Interface Design

This section describes some general user interface design.
4.4.1 Template Design

Since this is a project for FHWA, the web page’s template is going to be very the same as the FHWA’s official web site, except for some minor changes. For an example, the following is a screenshot of a page using the template.

![Screenshot of template design](image)

**Figure 4.7 Template Design**

The first element in the template is the background image, which covers the entire browser’s space. On top of the middle of the background image, there is the fixed width space for all of the web pages of this project; the fixed width space is to be referred to as the page.

The page itself has a white background-color filling the entire page. Within the page, there is a header on top, a body, and a footer area at the bottom of the page. Within the
header, there is the FHWA’s logo and FHWA’s full name in text, followed by a drop-down navigation menu. Within the footer area, which is at the bottom of the page, in addition to the logo and the full name of FHWA, there are varies links to other resources related to federal government. Finally, in between the header and footer area, there is the body area, which contains a headline right under the dropdown menu, followed by a sub-navigation vertical menu floating on the left and a main content area floating on the right.

4.4.2 Main Content Design

![Review Existing NBI Files or Upload a New NBI File to the System](image)

**Figure 4.8 Main Content Designs**

Using the same page as an example, the main content area contains a description area on the top. Followed by the description area is the page specific content area. This area is different from page to page. However, the look and feel for tables would be the same. Table headers are with blue background color and white text. Other types of text other than table headers are in black color. Tables have no interior borderlines, but there might be spaces and paddings for cells.
4.4.3 Forms Design

Mainly, there are two forms for this project. The common features for these forms are floating divisions, drag-gable areas, outlined with blue solid lines, and blue background title area with white text on the top. However, specifically, these two forms are all different since they serve different purposes.

![Assessment Level Change Form Design](image)

Figure 4.9 Assessment Level Change Form Design

The screenshot above is the Assessment Level Change Form. It has a warning area with red text enclosed by red solid line, reminding the users that change an assessment level may implies an increase in workload. Then, there is the area for displaying population size, random sample sizes for the four assessment levels for the selected metric of the session. Following the population information, there is the drop-down selection box for selecting levels. At the very bottom of the form, there are the submit button and the close button for the form.
The second and perhaps the most important form is the Compliance Review Supplemental Data form, shown below.

Figure 4.10 Random Sample Assessment Data Form

The above is the Compliance Review Supplemental Data Form. Its details will be described in chapter 5 since it ties more to its operation.
Chapter 5

SYSTEM IMPLEMENTATION

5.1 Site Map

This web based consists of a series of web pages and the necessary files to make how those pages the way they look. The following is a site map for this web-based system, including web pages and the first level folders from the root.

[Diagram of system implementation site map]

Figure 5.1 System Implementation Site Map
5.2 System User Interface

This section describes the pages with graphical interface in the site map shown above, except for include_database.php, which is not a web page, but a PHP class file.

5.2.1 Page nbi_file.php

![Figure 5.2 NBI File Uploader Interface](image)

This page consists of two parts. On the top, there is a form with three fields, year, state, and file upload. The file upload field is used for browsing NBI file from the user’s local machine. The year and state is for selecting year and state for the NBI file being uploaded in the file upload field. At the bottom, there is a table listing all of the NBI files that are already in the system sorted by the file’s year in descending order. This page is meant to be used by administrators only. However, use access control is not enforced in this version of the system. This applies to all of the other NBI Reviewer pages. Those
pages are meant to be used by NBI Reviewer, but any users have access to the system have access to bridge inspector’s pages.

5.2.2 Page session_log_in.php

This page serves two purpose and thus have two parts handled by tabs.

![Session Creation Interface A](image)

Figure 5.3 Session Creation Interface A

The first tab labeled with “Resume Old Session” is used to resume sessions created earlier. There are three fields in this page. The third field is a drop-down selection box listing all of the existing sessions. The third field can be filtered by any combination of the values in the first two fields, the session year field and the state field. Once one of these two field changes its value, the page filters out the list of available sessions by the combination of the two fields. By default, all sessions are available for selection.
Figure 5.4 Session Creation Interface B

The second tab labeled with “Start New Session” is used to create a new session. It is a form consisting of six required fields: User Name, State, Year, County, Assessment Level, and Detail Description, and these fields are used to identify the session once it is created. State, County, Assessment Level fields are disabled by default. A selection made in the Year field will trigger a request to the database to retrieve available State values. The State field becomes active upon the completion of the request. Similar manner, County field becomes active upon the selection on the State field, and the Assessment Level field becomes active upon the selection on the County field. A session will not be created until all of the fields are filled with values.
5.2.3 Page review_overview.php

Figure 5.5 Session Overview Interface

This screenshot does not show the entire page, but it does reflect the purpose of the page. It mainly contains a table listing the compliance status of each of the 23 metrics. Once determination has been made for a metric, the Compliance column shows the metric’s compliance determination as a link, which takes the user to the metric’s report page. Also, the metric’s row in the table is colored coded accordingly; compliance metric is in green, conditionally compliance is in light blue, substantially compliance is in yellow,
and finally non-compliance is in red. In the table, the Conduct/Review link takes the user to the Assess Samples page for the corresponding metric.

5.2.4 Page assess_samples.php

![Sample Assessment Interface](image)

**Figure 5.6 Sample Assessment Interface**

This page is used to assess random samples and consists several components. The first component is the buttons labeled with number on the top. Clicking on buttons switches the assessment to the buttons’ corresponding metric numbers. Following the buttons is some data for the metric, including the population, assessment level, and sample size of
the metric. Further down is the Google Map showing the random sample’s geospatial location on the map. After the Google Map, there is a table, listing the identification data of the random samples sorted by their unique IDs in ascending order.

The zoom buttons on the right of each row in the table brings up the bridge’s detailed/comprehensive data page. One sample data page sheet is shown in Appendix C.

Additionally, this page has two hidden forms, the Assessment Level Change Form and the Assessment Data Form, both of which are shown and described below.

![Assessment Level Change Form](image)

Figure 5.7 Assessment Level Change Form Interface

Clicking on the wrench icon on the top of the map brings up this Assessment Level Change Form. Users can use to form to change an assessment level for the metric they are reviewing. Upon submitting the form, the system updates the locked samples in the
database by adding more samples or by removing locked samples. The bridge icons on
the Google Maps and the listing on the table are updated accordingly.

![Assessment Data Form Interface A](image)

Figure 5.8 Assessment Data Form Interface A

Clicking on the power button next to random sample’s identification in the table brings
up this Supplemental Data Form, which is used to answer questions, write notes, and/or
attach documents for the sample.

On the very top of the form is the identification of the bridge that activates the form.
Followed by are some tabs labeled with numbers. The numbers represent all of the
metrics that the bridge is used for assessment. Within each of the tabs, there is an
accordion with two sections. The first section of the accordion is for users to answer
questions and to write comments for the bridge. The first half of the first section displays relevant information of the bridge that helps the users to answer the assessment questions. Followed by the relevant information is the assessment questions used for assessing the bridge. In addition, there is a text area for users to write comments.

![Assessment Data Form Interface B](image)

**Figure 5.9 Assessment Data Form Interface B**

The second section of the accordion within the tab, shown above, is a form for users to attach documents. Users can also delete documents that are already uploaded. This file upload functionality is implemented using a jQuery File Upload library.

![Assessment Data Form Interface C](image)

**Figure 5.10 Assessment Data Form Interface C**
Furthermore, users can also choose to show or hide the Google Map and/or the table.

The page with hidden map and table is shown above.

Finally, user can press the Assess Compliance button shown above to get to the `determine_compliance.php` page. This Assess Compliance button is at the bottom of the page regardless of the visibility of the map and the table.

### 5.2.5 Page `determine_compliance.php`

The whole purpose of this page is to submit a snapshot for a compliance determination.

There are four parts on this page.

![NBI Compliance Determination Form](image)

**Figure 5.11 Determination Form Interface A**
The first part is some basic identification for metric being reviewed, including the metric’s reference number, the metric definition for compliance status.

<table>
<thead>
<tr>
<th>Assessment Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assessment Level: Intermediate - Tier 1</td>
</tr>
</tbody>
</table>

1. Are fracture critical members properly identified?
   - 1 out of 14 samples received 'Yes'; 1 received 'No'; 12 samples received 'TBD'.

2. Is the inspection frequency listed?
   - 1 out of 14 samples received 'Yes'; 1 received 'No'; 12 samples received 'TBD'.

3. Are the inspection procedures complete?
   - 1 out of 14 samples received 'Yes'; 1 received 'No'; 12 samples received 'TBD'.

4. Are conditions described?
   - 1 out of 14 samples received 'Yes'; 1 received 'No'; 12 samples received 'TBD'.

5. Are procedures followed during the FCM inspections?
   - 1 out of 14 samples received 'Yes'; 1 received 'No'; 12 samples received 'TBD'.

Figure 5.12 Determination Form Interface B

The second part is an assessment summary including the assessment level, population size, and sample size for the metric. This part also includes the questions used to assess each of the samples and a summary of what kind of answers the questions are getting.

<table>
<thead>
<tr>
<th>Earlier Observation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Highway Agency: California Department Of Transportation</td>
</tr>
<tr>
<td>Snapshots:</td>
</tr>
<tr>
<td>2011-09-20 12:42:23 SS: Not enough data to determine compliance...</td>
</tr>
<tr>
<td>2011-10-13 15:06:25 SS: Enter compliance statement here:</td>
</tr>
<tr>
<td>2011-10-26 14:14:04 SS: bu ho ge</td>
</tr>
<tr>
<td>Previous Action: selk xs</td>
</tr>
</tbody>
</table>

Figure 5.13 Determination Form Interface C
The third part is a snapshot of earlier observation. Earlier assessment compliance statements can be review here for reference and for helping filling out the determination form that comes below it.

![Compliance Determination Form Interface D](image)

**Figure 5.14 Determination Form Interface D**

Finally, the last part is a small form for users to fill out. The form consists of two selection boxes for determining status for assessment schedule and compliance, respectively, and two text areas for inputting statements for compliance determination and action to be taken upon this review. In addition, there is an input field for the user to input their name.

5.2.6 Page report.php

This page generates a printer-friendly report page for selected metrics. For the most part, this page looks very similar to the determine_compliance.php page, except for two
differences, a tab on top of the report and a compliance determination instead of a form at the bottom.

![Figure 5.15 Tab Navigation on Report Page](image1)

The first difference is the tabs on the top of the page, which is used for switch between the 23 metrics. The reason for having the tabs here is to extend the user’s experience with printing reports. Again, no reports would be available for metric 1 through 5 for this version of the project.

![Figure 5.16 Observations on Assessment Report](image2)
The second difference is that there is no form on this page. Rather, the input in the form on the determine_compliance.php page is submitted to the database. The report page then retrieves the input and the earlier observation for this assessment for the metric from the database to display here.

5.3 The Request_Handler Directory

While a combination of JavaScript files, CSS files, and PHP files creates virtually all of the files described in section 5.2, there is a set of PHP scripting files worth mentioning. Without their existence, pages in section 5.2 would not accomplish what they are supposed to do. However, at the same time, these PHP scripting files do not execute by themselves nor do they function by themselves. Rather, they are usually activated by AJAX calls from pages in section 5.2. Altogether, with pages in section 5.2, they fulfill certain sets of requirement. So, they are as important as those pages with user interface in section 5.2. They are stored in folder Request_Handler, a sub directory from the root. The following sub sections only describe the scripting files in a general sense. Detailed documentation please refer back to each of the individual scripting files.

5.3.1 Script generateReport.php

Called by report.php, this script takes a metric number as its parameter and returns report data in JSON format, a slim data exchange format for the internet. The returning data contains five main categories, report_meta, metric_identity, assessment_summary, earlier_observation, and random samples for the metric.
5.3.2  Script getAvailableStates.php

Called by session_log_in.php, this script takes a year as its parameter and returns available states back to its calling page in JSON format. The JSON is really as simple as a zero-indexed array.

5.3.3  Script getCounties.php

Called by session_log_in.php, this script takes a state as its parameter and returns available counties back to its calling page in JSON format. This JSON data can also be treated as a zero-indexed array.

5.3.4  Script getReviewSessions.php

Called by session_log_in.php, this script takes a no parameter and returns available sessions back to its calling page in JSON format. This JSON data can also be treated as a zero-indexed array.

5.3.5  Script loadApplicableMetrics.php

Called by assess_samples.php, this script takes a locked sample’s id parameter and returns all of the locked sample’s applicable metrics back to its calling page in JSON format. This JSON data can also be treated as a zero-indexed array.

5.3.6  Script loadLockedSampleData.php

Called by assess_samples.php, this script takes a locked sample’s id and a metric number as its parameters and returns the locked sample’s data back to its calling page in JSON format. This JSON data can also be treated as a zero-indexed array.
5.3.7 Script loadLockedSamples.php

Called by assess_samples.php, this script takes a session id and a metric number as its parameter and returns all of the locked samples applicable to the giving session id and metric number back to its calling page in JSON format. This JSON data has two main categories, overview and samples. The overview category has the metric’s population, metric number, session id, assessment level, review id, the sizes for the four assessment levels, and finally the locked samples.

5.3.8 Script submitReviewForm.php

Called by assess_samples.php, this script takes a locked sample’s id and a metric number as its parameters and updates the locked sample’s corresponding record in the metric’s corresponding table. Returns ‘ok’ when success.

5.3.9 Script update_assessment_level.php

Called by assess_samples.php, this script takes a new level number, an old level number, and a metric number as its parameters. This page not only updates the assessment level as request, but also locks down more samples or deletes some samples as necessary.

Returns ‘OK’ upon success.

5.3.10 Script update_assessment_status.php

Called by determine_compliance.php, this script takes six parameters, a metric number, compliance status, schedule status, a string statement, an action statement, and a user
name. This page updates the status in review_overview table and insert a new record into the metric_snapshot table. Users are redirected to report.php upon success.

5.4 Other Directories

This section briefly describes each of the other directories under the root directories.

5.4.1 Directory CSS

This directory stores the general CSS files that control the look and feel of the website.

5.4.2 Directory Document

This directory stores all kinds of documents, including but not limited, manuals and examples.

5.4.3 Directory FTPUpload

This directory stores the attachments uploaded during compliance assessment. NBI original input files are also stored in this directory.

5.4.4 Directory images

This directory stores the images used to build this project.

5.4.5 Directory includes

This directory stores general PHP include files. These files provides functionalities for other pages.

5.4.6 Directory jquery_plugins

This directory stores jQuery open source plugins downloaded from the internet.
5.4.7 Directory js

This directory stores some general JavaScript files.

5.4.8 Directory Smarty

This directory stores the Smarty template engine.

5.4.9 Directory templates

This directory stores the Smarty templates for the project.

5.4.10 Directory templates_c

This directory stores the compiled smarty templates.
Chapter 6

CONCLUSION AND FUTURE WORK

6.1 Conclusions

This web-based system is able to provide the basic functionality as previously described in the functional requirements section. To summarize, the features this system is providing are listed below.

- The system is able to parse NBI file and store the data within the NBI file into the system’s MySQL database system.
- The system is able to randomly select sample when its user is creating a new compliance review session. Reviewers no longer need to manually perform random sampling.
- The system is able to collect data for the randomly selected samples, including answers to yes or no questions, comments on the samples, and/or attachments needed for the samples.
- Once data for randomly selected samples is collected, the system is able to provide a summary of the collected data. Users can review the data and determine compliance for a metric. Users can also provide compliance statement and required action along with the compliance result for a metric.
- The system is able to provide functionality for users to change assessment levels for metrics.
• The system is able to provide a SI&A-like sheet for user to the review. The system is also able to display bridges on Google Maps.

• The system is able to provide a summary compliance status for the metrics, except for metrics 1 through 5 since they are out of the scope of this project.

• The system is also able to provide a basic report for the metrics, except for metrics 1, 2, 3, 4, and 5 since they are out of the scope of this project.

In short, the system is able to collect data for NBI compliance review. A basic framework for NBI compliance review is set for future improvement. Although, more comprehensive report is not provided by this system at this moment, necessary data is collected and stored in the system’s database.

6.2 Challenges and Lessons Learned

There have been challenges implementing the system. Software engineering requires developers constantly communicate about the requirements and get feedbacks on implementation. This lack of communication leads to delays and misunderstanding of requirements. Nonetheless, this turns out to be very valuable experience to have, although I had to learn it the hard way that communication is essential for success.

Another challenge for this project is that scope of the project was hardly defined for a long time. At the very beginning of the project, there were only concepts on how clients would perform the work without the system. Moreover, the concepts are not in the field
of Computer Science. A lot of document reading and searching was done before understanding and defining a set of requirements.

Finally, growing requirements was a challenge. At a point, we had to clearly define a number of requirements for this version of the project. Otherwise, there would have been more and more requirements. Learning scope creep, the uncontrollable changes in requirement [12], is so valuable.

6.3 Future Work

Although the system is able to perform a number of tasks, there are areas that can be improved and there are some other nice features that should be added to the system if time allows. They are listed below.

- First are the format and the content of the report for the metrics. The current system provides a very basis data and information. More comprehensive content for the report has to be defined before improvement can be made for the report format.

- Another area is that the questions using to assess/review bridges can be further refined. The FHWA agencies are conducting future meetings to come up with better and more throughout questions. At this point, the questions are satisfactory for FHWA California Division.
• Currently, review session does not support multi-county review. However, it is mentioned that the system should support multi-county review session in the future.

• The system needs to maintain information more than just bridge assessment data, including, but not limited to, meta data for staff management and file management. However, further discussion on how that information is fed and maintained within the system is needed. As mentioned earlier in section 6.1, metric 1, 2, 3, 4, and 5 reviews are not included in this system. The main reason is that the system is not set up to support staff management and metric 1, 2, 3, 4, and 5 rely heavily with the existence of staff management. File management design and implementation are dependent on staff management. Unless staff and file management is in place, there is no easy way to review and assess those 5 metrics.

• In addition to the basic report for metrics, it is good to have a report for the executives, summarizing results from the 23 metrics. Also, it is nice to be able to provide reports in other formats other than web pages.

• Assessment for minimum level should be added to the system. The reason this feature is not in the system is that minimum assessment has nothing to do with random sampling. The system does provide a foundation for implementing this feature.
APPENDIX A
Random Sampling Procedure

A.1 The Procedure

1. Randomly generate a number for each of the bridges.

2. Sort the bridges by their random numbers in ascending order.

3. Determine the population size for each of the metrics.

4. For each of metrics in their sorted order, determine the number of the bridges needed for compliance determination.

5. Sort the metrics by sample size in descending order.

6. The first so many of bridges on the sorted bridge list, (the number depending on the number of bridges needed for the first metric on the sorted metric list), is used to initialize the assessment master list.

7. For each of the metrics in their sorted order,

8. Copy bridges that are applicable to the metric from the assessment master list

9. If the number of bridge copied is less than the number of bridges needed, search for applicable bridges from the sorted bridge list from top to bottom to find the remaining.

10. Append the bridges found to the master list.

11. The next metric on the sorted metric list uses the updated master list to complete steps a, b, and c.
A.2 Example

I will use 4 metrics to illustrate the algorithm. Say we need 19 bridge for metric 6, 17 bridges for metric 7, 19 bridges for metric 20, and 14 bridges for metric 9. These four metrics is sorted by their sample size, resulting in 6, 20, 7, 9.

Assuming the bridges are already sorted by random numbers in ascending order, the first 19 bridges from the sorted bridges are used as the assessment master list. Since the entire population is applicable to metric 6 and metric 20, the first 19 bridges would also be the random samples for metric 6 and metric 20.

The next on the sorted list is metric 7. Not all bridge is applicable to metric 7. So we have to identify whether a bridge is applicable to metric 7 first before the bridge is chosen. Say 10 of the bridges in the assessment master list are applicable metric 7, those 10 bridges are used metric 7. We continue to search for the remaining 7 bridges from the sorted bridges in their sorted order. When a bridge is found to be applicable to metric 7, the bridge is used for metric 7 until 7 applicable bridges are found. The bridge is also added to the assessment master list. At the end of processing metric 7, 7 more bridges are added to the assessment master list, and the assessment master list ends up with 26 bridges, 7 additional bridges to the original 19.

Now, we need 14 bridges for metric 19. The procedure would be exactly like how it is done for metric 7. However, the updated assessment master list, which contains 26 bridges, is used instead of original master list. The updated assessment master list is
updated again according for other metrics. If again, 10 bridges in the updated assessment master list are applicable, 4 more bridges will be added to the updated assessment master list, which ends up with a total of 30 bridges.
Figure B.1 Database Entity Relationship Diagram - Design
APPENDIX C

Database Implementation

Figure C.1 Database Entity Relationship Diagram - Implementation
APPENDIX D

Detail Bridge Data Sheet

FIGURE D.1 DETAILED BRIDGE DATA SHEET

<table>
<thead>
<tr>
<th>(1) STATE NAME - CALIFORNIA</th>
<th>CODE 069</th>
</tr>
</thead>
<tbody>
<tr>
<td>(2) STRUCTURE NUMBER</td>
<td>25 0021K</td>
</tr>
<tr>
<td>(3) INVENTORY ROUTE (OWNERS) - ON</td>
<td>10700040</td>
</tr>
<tr>
<td>(4) HIGHWAY AGENCY DISTRICT</td>
<td>04</td>
</tr>
<tr>
<td>(5) PLACE CODE</td>
<td>00000</td>
</tr>
<tr>
<td>(6) COUNTY CODE</td>
<td>053</td>
</tr>
<tr>
<td>(7) FACILITY CARRIED</td>
<td>BALLEY RD, W 2 ON RP</td>
</tr>
<tr>
<td>(8) LOCATION</td>
<td>06- CC-004 R20. 1.</td>
</tr>
<tr>
<td>(9) REASON - LENGTH</td>
<td>25 095.30</td>
</tr>
<tr>
<td>(10) LOCATION</td>
<td>30 DEG 08 MIN 14.85 SEC</td>
</tr>
<tr>
<td>(11) LATITUDE</td>
<td>121 DEG 36 MIN 58.30 SEC</td>
</tr>
<tr>
<td>(12) LONGITUDE</td>
<td>94 1SET 94</td>
</tr>
<tr>
<td>(13) BORDER BRIDGE STATE CODE</td>
<td>94 1SET 94</td>
</tr>
<tr>
<td>(14) DATA</td>
<td>1</td>
</tr>
<tr>
<td>(15) BORDER BRIDGE STRUCTURE NO.</td>
<td></td>
</tr>
</tbody>
</table>

**STRUCTURE TYPE AND MATERIAL**

<table>
<thead>
<tr>
<th>TYPE - BOX BEAM OR GIRDERS - MULTIPLE</th>
<th>CODE 205</th>
</tr>
</thead>
<tbody>
<tr>
<td>STRUCTURE TYPE A/P/E: NOT APPLICABLE</td>
<td>CODE 000</td>
</tr>
<tr>
<td>NUMBER OF SPANS IN MAIN UNIT</td>
<td>001</td>
</tr>
<tr>
<td>NUMBER OF SPANS</td>
<td>0000</td>
</tr>
<tr>
<td>DECK STRUCTURE TYPE - CONCRETE C.I.P</td>
<td>CODE 1</td>
</tr>
<tr>
<td>WEARING SURFACE / PROTECTIVE SYSTEM</td>
<td></td>
</tr>
</tbody>
</table>

**AGE AND SERVICE**

| YEAR BUILT | 1097 |
| RECONSTRUCTED | 0000 |
| TYPE OF SERVICE ON HIGHWAY | 15 |
| LANES ON STRUCTURE 02 | UNDER STRUCTURE 00 |
| AVERAGE DAILY TRAFFIC | 12 00 |
| YEAR OF ADT 2000 | 1099 |
| TRUCK ADT 06% | 0000 |
| BYPASS DETOUR LENGTH | 0000 |

**GEOMETRIC DATA**

| LENGTH OF MAXIMUM DATA | 01 00 |
| STRUCTURE LENGTH | 00 00 |
| CURB OR SIDEWALK LEFT 60.0 | 00 00 |
| BRIDGE ROADWAY WIDTH Curb to Curb | 00 00 |
| DECK WIDTH OUT TO OUT | 00 00 |
| APPROACH ROADWAY WIDTH (W/SHOULDER) | 00 00 |
| BRIDGE MEDIAN - NO MEDIAN | 00 00 |
| SKew 0 DEG | 00 00 |
| INVENTORY ROUTE NOT VERIY CLEAR | 00 00 |
| INVENTORY ROUTE TOTAL HORZ CLEAR | 00 00 |
| M viện UNDER CLEAR REF - HIGHWAY | 00 00 |
| M.instances UNDER CLEAR REF - NEITHER | 00 00 |
| NAVIGATION HORIZONTAL CLEARANCE | 00 00 |

**PROPOSED IMPROVEMENTS**

| TYPE OF WORK NOT APPLICABLE | CODE 000 |
| LENGTH OF STRUCTURE IMPROVEMENT | NOT APPLICABLE |
| BRIDGE IMPROVEMENT COST | NOT APPLICABLE |
| ROADWAY IMPROVEMENT COST | NOT APPLICABLE |
| TOTAL PROJECT COST | NOT APPLICABLE |
| YEAR OF IMPROVEMENT COST ESTIMATE | NOT APPLICABLE |
| FUTURE ADT | 00 00 |
| YEAR OF FUTURE ADT | 00 00 |

**NAVIGATION DATA**

| INSPECTION DATE | 00 00 |
| CRITICAL FEATURE INSPECTION | 00 00 |
| OTHER SPECIAL INSPECTION | 00 00 |
| CRITICAL FEATURE PERFORMANCE | 00 00 |
| OTHER SPECIAL INSPECTION | 00 00 |

**INSPECTIONS**

| CRITICAL FEATURE PERFORMANCE | 00 00 |
| OTHER SPECIAL INSPECTION | 00 00 |
APPENDIX E
Application Stack

E.1 A List of Needed Items

- Machine running with Windows XP/7
- A USB drive with a minimum of 1GB memory
- XAMPP, can be downloaded from apachefriends.org
- PHP, can be downloaded from php.net
- Smarty, can be downloaded from smarty.net
- jQuery, can be downloaded from jquery.com
- jQuery Plugin File Upload, can be downloaded from http://aquantum-demo.appspot.com/file-upload
- jQuery Plugin Tool Tip, can be downloaded from http://plugins.jquery.com/node/3678
BIBLIOGRAPHY


