13. Personal Communication, Oregon Department of Transportation, Erick Cain, Bridge Inventory Coordinator, December 2013.

   http://advitam-group.com/Content/brochures/EN/Scanprint%20IMS_EN_A4_LR.pdf


16. Figure 6, Finley Engineering. Accessed January 2014.


   www.sungardps.com/solutions/mobile

   http://www.imecotechnologies.com/


   http://www.rugladeshiz.com/algiz-10x/


8.3 Survey Results

Wireless Data Collection and Retrieval of Bridge Inspection/Management Information: Survey Response Summary as of April 29, 2014

Colin Brooks (cnbrooks@mtu.edu), Tess Ahlborn (tess@mtu.edu), and Nate Jesse (njessee@mtu.edu)

Michigan Technological University (MTU) and its research center the Michigan Tech Research Institute (MTRI) are developing a data collection system specifically designed to acquire bridge inspection data, with a focus on element level information. This research is being done as a part of the Michigan Department of Transportation project “Wireless data collection and retrieval of bridge inspection/management information” (OR14-021). To help our research staff gain better insight to the current practices and applied technologies currently available, a short survey was designed and distributed to bridge inspection managers across the United States by Matt Chynoweth of MDOT on March 26, 2014 to the AASHTO Subcommittee of bridges and structures e-mail list. This document will summarize the responses that have been submitted to date.

As of April 29, 2014 a total of twenty-one survey responses have been received. The following states have responded: Delaware, Florida, Hawaii, Illinois, Iowa, Maryland, Minnesota, Missouri, Nebraska, New Hampshire, New Jersey, New York, North Carolina, North Dakota, Ohio, Oklahoma, South Dakota, Texas, Wyoming, Utah, and Virginia. The responses represent a wide range of States and the answers vary in detail and have provided a valuable source of information. A portion of the survey responses will be shared at the end of this document, and all responses will be shared as a separate document once the survey deadline of May 2, 2014 has passed.

So far, over 70% (15 of 21) of responding States use some form of electronic hardware to collect and manage bridge inspection data. Of those using electronic devices, over half use laptop computers to collect and manage data. Currently, there is a moderate number of commercial software packages available for transportation infrastructure management and inspections, as described in the State of the Practice Report submitted to MDOT from the MTRI/MTU research team. Despite the available software packages, the survey responses indicate that many of the bridge inspection departments utilize “in-house” or custom software for data collection and management. In addition to the “in-house” systems, many other State departments use existing commercial software, but have customized the standard software package to fit their specific needs. Of the twenty-one responses, only four use commercial products in their native format. The high number of custom/self-made systems indicates that a single software platform may be unlikely to meet the needs of multiple states or infrastructure departments. Those States that did not utilize an electronic data collection and management have indicated that they are likely to adopt such a system, but some concerns were raised, such as the cost of equipping a staff of bridge inspectors with mobile electronic devices. Also, over 80% of
those that responded indicated that they are willing to be contacted for additional surveys/questions.

These survey responses will be useful to our research and software design. For instance, understanding that many of these agencies are interested in obtaining or improving their bridge data collection and management systems, such as stated by Mike Brokaw of the Ohio DOT, that “Ohio is transition to a web-based system (and that) DOT inspectors collect inspection and maintenance data on an in-house accessed based interface, (which is) decoupled from state servers” is of importance (Mike Brokaw, survey respondent, 2014). Specific phrases, such as “decouples from state servers” helps remind that security is often a priority for database managers. Other information, such as flexibility in personnel usage of software is also an insight that may help determine how modern hardware and software can accommodate users who may be hesitant to adopt new practices. Such an instance was stated by the Minnesota DOT, “Some inspectors prefer to print out copies of the last inspection form the database to take out in the field” (Jennifer Zink, survey respondent, 2014).

Part of our current research vision involves the use of mobile internet (4G LTE connection specifically). Being able to access inspection databases while in the field seems to be a valuable feature for a future inspection and management system. To understand how frequently inspectors would have cellular coverage, we asked the participants to rate how often 4G connections would be available to their inspectors and the answers were rather divergent. Approximate 24% of the responses indicated that cellular coverage/availability in their districts was less than 50%. The number of responses indicating that cellular coverage was greater than 75% was near the same as it was for those with less than 50% coverage. Unfortunately, this question was misunderstood by approximately 20% of the participants, with one responder stating that cellular coverage was never available; we believe that the responder meant to say that they are not equipped with cellular devices while in the field. The following figures breakdown some of the responses that could be quantified. Lastly, the Virginia DOT listed that they use existing commercial software that allows them to include sketches into their work. We intend to follow up with them about this feature as it is a high priority for our current software system.
Figure 1: These statistics detail the mobile device usage of the responding agencies. The mixed category includes agencies that use both tablets and laptops.

Figure 2: This statistic represents the types of inspection and management software currently being used. Notice that many agencies use custom software. The mixed/other category represents modified or customized commercial solutions.
<table>
<thead>
<tr>
<th>Agency</th>
<th>Mobile Devices Use</th>
<th>Software Use</th>
<th>Cellular Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delaware DOT</td>
<td>Laptops</td>
<td>We use Pontis 4.5 and in-house software for photos and supporting documents.</td>
<td>75 – 100%</td>
</tr>
<tr>
<td>Florida DOT</td>
<td>None</td>
<td>N/A</td>
<td>&lt; 10%</td>
</tr>
<tr>
<td>Hawaii DOT</td>
<td>None</td>
<td>None</td>
<td>75 – 100%</td>
</tr>
<tr>
<td>Illinois DOT</td>
<td>None</td>
<td>N/A</td>
<td>10 – 50%</td>
</tr>
<tr>
<td>Iowa DOT</td>
<td>20% of NBI inspectors use laptops to review previous records and to enter quantities and photos. Maintenance shops have started using iPads for non-bridge size culvert inspections.</td>
<td>NBI inspectors use InspectTech and culvert inspectors use Fulcrum app</td>
<td>10 – 50%, rural areas are limited, but urban areas are stronger</td>
</tr>
<tr>
<td>Maryland State Highway Administration</td>
<td>Laptops (HP ProBook 6560b Air Cards</td>
<td>InspectTech</td>
<td>75 – 100%</td>
</tr>
<tr>
<td>Minnesota DOT</td>
<td>Laptops, iPads, Smartphones</td>
<td>Customized InspectTech</td>
<td>75 – 100%</td>
</tr>
<tr>
<td>Missouri DOT</td>
<td>None</td>
<td>None</td>
<td>Estimate in the 50 – 75% range</td>
</tr>
<tr>
<td>Nebraska DOT</td>
<td>Laptop Computers</td>
<td>Pontis 4.4.3. BrM 5.2.1</td>
<td>Unknown at this time, but coverage should be good (+75%) with Verizon. We intend to use this technology starting with this current inspection cycle, Using BrM 5.2.1</td>
</tr>
<tr>
<td>New Hampshire DOT</td>
<td>Laptops</td>
<td>Pontis</td>
<td>Areas of NH are very remote; each inspection team has &quot;dead zones&quot;. Our northern inspection team has coverage in the 10 - 50% range. Our three southern teams have coverage in the 75 - 100% range.</td>
</tr>
</tbody>
</table>

Table 1: Quoted and paraphrased answers from responding State Agencies.
<table>
<thead>
<tr>
<th>Agency</th>
<th>Mobile Device Use</th>
<th>Software Use</th>
<th>Cellular Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Jersey DOT</td>
<td>Laptops (Optional) Only for portion of Inspections</td>
<td>Commercial (Customized for NJ) Bentley Inspect Tech</td>
<td>10 – 50% (Used to take photos of significant defects to send to office for review and action)</td>
</tr>
<tr>
<td>New York State DOT</td>
<td>Laptops</td>
<td>Existing in house software</td>
<td>About 75% cases</td>
</tr>
<tr>
<td>North Carolina DOT</td>
<td>Windows Touch Tablets</td>
<td>In House Software (WIGINS)</td>
<td>Never</td>
</tr>
<tr>
<td>North Dakota DOT</td>
<td>Laptops may be used but most inputting is done at the office</td>
<td>In house created application</td>
<td>10 – 50%</td>
</tr>
<tr>
<td>Ohio DOT</td>
<td>Currently: Ohio is transitioning to a web based system. Future: Ohio will deploy a web based system for all users that can also be used offline in May 2014.</td>
<td>Currently: Inspectors, State and Local, primarily use one of two offline versions that can interface with the mainframe DOS BMS.</td>
<td>We anticipate 50-75% of the state having usable coverage however we expect only 10-50% of the bridges within that coverage</td>
</tr>
<tr>
<td>Oklahoma DOT</td>
<td>We do not collect data in the field with electronic devices</td>
<td>None</td>
<td>Unsure</td>
</tr>
<tr>
<td>South Dakota DOT</td>
<td>None</td>
<td>None</td>
<td>Unsure</td>
</tr>
<tr>
<td>Texas DOT</td>
<td>Laptops</td>
<td>Custom</td>
<td>50 – 75%</td>
</tr>
<tr>
<td>Wyoming DOT</td>
<td>WYDOT is currently using laptops for field data collection.</td>
<td>We are currently using software that was developed in-house. However, WYDOT has recently signed a contract with Bentley / InspectTech and will be using commercial software in the near future.</td>
<td>50 – 75%</td>
</tr>
<tr>
<td>Utah DOT</td>
<td>Currently laptops – hope to transition to tablets this summer</td>
<td>Pontis, customized for Utah DOT</td>
<td>50 – 75%</td>
</tr>
<tr>
<td>Virginia DOT</td>
<td>Tablets (Motion Tablets, Laptops with rotating screens - old)</td>
<td>Commercial software (write and turn into text, voice recognition, sketching)</td>
<td>0%, not allowed by IT</td>
</tr>
</tbody>
</table>

Table 2: Quoted and paraphrased answers from the responding State Agencies. Note in the fourth column that the Virginia DOT responder stated the cellular coverage was not allowed by their IT department. This example shows some of the ambiguity with this question.
8.4 Requirements Document

Wireless Bridge Inspection System
Software Requirements Specification

Michigan Tech Research Institute (MTRI)
3600 Green Court, Suite 100
Ann Arbor, MI, 48105

Prepared for MDOT (Michigan Department of Transportation)
Project MDOT OR14-021 Contract Number 2013-0067, Authorization 2
Project Manager: Richard Kathryn, MDOT

VERSION 1.0 JUNE 2014
Table of Contents

1 Purpose .................................................................................................................. 4
2 Product Scope ....................................................................................................... 4
3 Current State of the Practice ............................................................................... 4
4 Concept of Operations for Tablet-Based System .............................................. 5
5 General Description of Software ........................................................................ 6
  5.1 Product Functions .............................................................................................. 9
  5.2 What the WBIS will not do ............................................................................. 10
  5.3 User Characteristics ....................................................................................... 11
  5.4 General Constraints ....................................................................................... 11
  5.5 Assumptions and Dependencies ..................................................................... 11
    5.5.1 Application Platform Assumptions ......................................................... 11
    5.5.2 Bridge geometry assumptions ................................................................. 12
6 Specific Requirements ......................................................................................... 13
  6.1 System Interfaces ............................................................................................ 13
    6.1.1 User Interfaces .......................................................................................... 13
    6.1.2 Hardware Interfaces ............................................................................... 14
    6.1.3 Communication Interfaces .................................................................... 16
  6.2 Functional Requirements Summary Table .................................................... 17
  6.3 Use Cases ........................................................................................................ 18
    6.3.1 Prototype Use Case ............................................................................... 18
  6.4 Quality Requirements ...................................................................................... 18
    6.4.1 Performance ............................................................................................. 18
    6.4.2 Reliability ................................................................................................ 18
    6.4.3 Availability ............................................................................................... 19
    6.4.4 Security .................................................................................................... 19
    6.4.5 Maintainability ......................................................................................... 19
    6.4.6 Portability ................................................................................................ 19
  6.5 Design Constraints .......................................................................................... 19
6.6 Logical Database Requirements ................................................................. 20
7 APPENDIX A........................................................................................................ 21
7.1 Definitions....................................................................................................... 21
7.2 Coordinate System for Bridge Rendering......................................................... 22
1 Purpose

This Software Requirements Specification (SRS) Document shall contain all of the information needed by a software developer to adequately design the Wireless Bridge Inspection System described in this document. In addition, it delimits the scope of the current project phase and communicates the detailed work plan to MDOT.

2 Product Scope

The 3D MDOT Wireless Bridge Inspection System (3dWBIS), or BridgeView, is a mobile-device application framework enabling the collection of MDOT bridge inspection data for the purpose of generating inspection reports, replacing the current paper-form method. The application serves these core purposes:

1) To allow bridge inspectors to indicate bridge deterioration on the AASHTO (see Definitions) element level;
2) To allow bridge inspectors to precisely locate bridge deterioration within an AASHTO element;
3) To eliminate paper forms and streamline the collection and collation of digital inspection data including field photographs;
4) To provide the bridge inspector with access to previous inspection data; specifically, previous inspection reports.

The application will enable bridge inspectors to enter CoRE/AASHTO element-level inspection data more quickly and consistently than currently allowed. The application will also enable bridge inspectors to access reference data including previous inspection reports and field photographs while in the field.

The goal is to introduce a novel bridge inspection data collection method that streamlines and enhances the element-level inspection process through Inspector interaction with a 3D engineering representation of the bridge under inspection.

3 Current State of the Practice

Currently MDOT bridge inspections are carried out by trained specialist Inspectors who examine all aspects of a bridge, mark up past versions of inspection reports with updated condition state data for a variety of bridge elements and functions, and finalize the current report with data entry at the home office. This is considered the current nominal “use case” for the application. The additional 3D model interactivity will be discussed in section XX.
Field inspection assignments are made via MiBridge (see Definitions) through the Inspection Assignment Dashboard. Candidate bridges for inspection are sorted by their due-inspection dates, nearest due dates first. A team lead is assigned a number of bridges to be inspected that season. Draft bridge inspections can be saved to the BMS but only the originating team lead can edit them.

Routine bridge inspections are conducted in the field by two-member teams. The inspector(s) walk the bridge inspecting and rating the condition of predefined bridge components (elements) while making comparisons to the previous condition of the same as indicated by past inspection reports. Two types of ratings are given, depending on the component: condition ratings, which describe the existing condition compared to the original, as-built condition of the bridge, and appraisal ratings, which describe components in comparison to a new structure built to current standards. During a routine inspection, which is conducted at least once every two years, ratings are given according to criteria specified on three different inspection forms, listed below.

- The Bridge Safety Inspection Report (BSIR), which includes the National Bridge Inventory (NBI) condition ratings
- The Structure Inventory and Appraisal (SI&A)
- The CoRE/AASHTO Elements Inspection (the focus of this 3dWBIS)

On the BSIR form, inspector(s) can see the condition ratings given to each component during past inspections. They also see the inspector's comments on each component from previous inspections. The form is organized by the location of the components into the following groups, in order: Deck, Substructure, Superstructure, Approach, and Miscellaneous. The SI&A form is a complete list of both inventory information about a bridge, which is generally unchanging (e.g. the year a bridge was built), and the appraisal ratings for each element.

The CoRE/AASHTO Elements form lists, for each CoRE/AASHTO element, the quantity of that element (an area or volume) that fall into each of the four condition states. For each of the four condition states, the last recorded quantity (the "old" quantity) can be seen.

An MDOT bridge inspector tends to first fill out the NBI ratings and the CoRE/AASHTO elements condition states; the comments are entered into the digitized inspection form within MiBridge at a later date. In general, if the comments are missing from an inspection form, it is assumed to be unfinished. Notes are annotated in the margins of the printed inspection form while in the field. Notes may be related to particular item/element on the form but general notes about the structure are also written.

4 Concept of Operations for Tablet-Based System

Inspectors will use the tablet to download past inspection data for bridges in their Inspection List. At the bridge site, the inspectors will walk the bridge as before, but enter relevant element-level data into appropriate fields in the tablet application’s user interface. The Inspector can navigate
to specific instantiation of an aggregate bridge element (eg, a single column of the “piers” class) by interacting with a 3D representation of the bridge under inspection. This 3D model will be rendered in sufficient detail to enable navigation and localization of all inspected elements, though not considered an engineering-drawing level of representation.

The inspector can enter data in any order and navigate to desired elements through menus or by interacting with the 3D bridge representation. Photos may be captured by the tablet device itself and will be tagged to the bridge and element under inspection.

3dWBIS General Concept of Operations

User Selects a Bridge → System generates Bridge Model Document → System Render 3D Bridge Representation and Inspection Interface → User Interacts with 3D Model and Enters Inspection Data → System Calculates and Saves Element Data

Figure 1: Notional high level workflow for tablet-based bridge inspection

5 General Description of Software

We are developing a mobile application for bridge inspection that will be called the 3D Wireless Bridge Inspection System (also referred to as the “application”, “mobile application”, “system”, or “BridgeView” in this document).

This mobile application will interface with a remote server hosting the surrogate Bridge Management System (sBMS), a relational database management system (DBMS) that will stand in for MDOT’s Bridge Management System (BMS) during software development and testing. The interface between the mobile application and the sBMS is a web application programming interface (API) and consists of HTTP requests. API requests will be received by a server on which the database middleware and database management system (DBMS) is running.

The bridge under inspection will be represented as a 3D rendered object that the Inspector can interact with in order to indicate the location of, and tag inspection data to, condition states of individual bridge elements.

MDOT Wireless Bridge Inspection System - Software Requirements Definition V1

Page 6
The software components that will be developed by the Michigan Tech Research Institute include and are limited to:

- The Wireless Bridge Inspection System tablet application, including the 3D Bridge Rendering Engine
- The surrogate Bridge Management System (sBMS) database
- The Server application that mediates between the sBMS and the Tablet application, including the Bridge Model Schema Generator
- The web-based API
  - The Bridge Model Schema

The software components that are already developed and which will be selected from among a number of appropriate alternatives are:

- The database management system (DBMS)
- The database middleware

The Wireless Bridge Inspection System application will be used in the field by bridge inspectors for the entry of MDOT CoRE Elements/AASHTO Elements inspection data. It will also allow inspectors to view the same inspection data for previous inspections. The inspection data as a whole from one of a bridge’s previous inspections can be used to populate the form for a new inspection of the same bridge, where appropriate.

A new functionality that we believe unique to the industry is the rendering of a 3D representation of the bridge for the Inspector to interact with from a query to the bridge database combined with bridge design rules.

This model is the basis for interaction with Element-level inspection components, indicating localization of conditions on individual instances of an element (for example, a specific pier column or slab). The Inspector enters quantitative values for the individual element condition states, and the Application performs the aggregation over element to provide the final percentage or ratio per element as currently provided in the Element level forms.

However, the underlying granular state of each instance is saved for future reference and Inspections.
3D-WBIS Concept

Figure 2: Wireless Bridge Inspection System Concept

The data entry process will be very similar to the established process in that the entry of inspection data will follow the typical order of MDOT inspections (i.e. as the inspector “walks the bridge”). However, the user will also be able to enter inspection data “out of order,” if desired. Unlike existing inspections, where the inspector must total the deterioration in each condition state for every AASHTO element, the application will do this automatically for the inspector as new areas of deterioration are found and entered into a condition state.

The application will be run on tablet computers, specifically on Android or iOS tablet computing platforms. For the Phase I (first year beta version), the Android operating system is the deployment platform for initial development and testing. Users will connect to a remote server for uploading completed inspection reports and for downloading past inspection reports and bridge models. The application will be available whether the mobile device is connected to the internet and able to reach the remote server or not. In the latter case ("offline mode"), inspection data collected by the user can still be entered in the application. Inspection reports will be saved to the device until they are completed and ready to be uploaded. Completed inspection reports that were previously downloaded for reference will also be available offline. While offline, the...
completed inspection reports will not, of course, reflect any changes made to them on the remote server by another user.

Field photographs and images from remote sensing technologies will be also available through the application. However, metrics from remote sensing surveys and projected remote sensing imagery will not be available. Images can be annotated by the user with comments that will be saved and synced with remote storage.

Users will be able to create and edit new inspection reports saved on their device until they mark the report as formally completed and upload it to the remote server (sBMS). Once the completed inspection report is uploaded, changes can no longer be made through the Wireless Bridge Inspection System. Users can view completed inspection reports including those they have just uploaded as completed. However, these completed inspection reports are available for viewing only (i.e. they are "read-only").

The application will also provide quick-reference materials for bridge inspectors including the AASHTO standard Rating Guides.

The Wireless Bridge Inspection System will annotate bridge inspection reports with automatic metadata as the reports are created, edited, and ultimately submitted to the remote server (sBMS). These metadata will be limited to:

- When a field in an inspection form was changed
- The last value of a field before it was changed
- The date, time, and submitting user identification for submitted inspection reports
- The location and orientation of the mobile device for each field photograph taken

5.1 Product Functions

The WBIS will fulfill the functions listed below.

- Emulate or improve the entry of CoRE/AASHTO element-level inspection data (via User Interface)
- Require users to authenticate themselves before submitting data to a remote server
- Provide data from previous inspection reports, including photos and element ratings
- Provide access to inspection manual information
- Provide limited quality assurance (QA), such as ensuring all fields are filled out, prior to finalizing a report
- Provide the ability to attach photos to the report

The 3dWBIS will make data entry easier by providing:

- An intuitive graphical way to navigate to Bridge elements

MDOT Wireless Bridge Inspection System – Software Requirements Definition V1  Page 9
• A systematic way to visualize past condition state data local to a particular element instance Lookup data such as substitutions for enumerations, abbreviations, or codes (e.g. "5" means "steel")
• Automatically tallying quantities of condition states for element level data
• Drop-down selectors for fields with a defined set of valid inputs

5.2 What the WBIS will not do

Data from scopings (see Definitions), waterway surveys, and other non-routine inspections will not be entered into the WBIS. The WBIS will not provide for the entry of "routine BSIR" inspection data such as the routine National Bridge Inventory (NBI) inspection data or Structure Inventory and Appraisal (SI&A) data. The WBIS will not allow for work recommendations to be filed. The WBIS will not provide any level of decision support such as deterioration rate estimates, bridge life-cycle or asset management products. The application will not enable mobile devices to take any direct contact or non-contact measurements of the bridge (other than photographs). The application will not track the location or behavior of the inspector in the field beyond metadata attached to inspection reports as defined above. The application will not provide any means of communicating between inspectors in the field or between inspectors and office personnel.
5.3 User Characteristics

The Users of the WBIS are expected to be qualified MDOT bridge inspectors (or contract assignees) with a proficient level of experience with current (2013) MDOT routine bridge inspection practices, or in training for such, or other supervisory personnel with experience with current (paper form) practices.

In addition, the User is expected to possess a basic understanding and familiarity with the operating system (OS) of the target hardware platform (e.g., Apple iOS, Android operating system), including but not limited to opening a native application, opening stored or downloaded documents, native navigation actions (home, menus, control panels), and data entry using a touchscreen and/or virtual keyboard.

5.4 General Constraints

The application is constrained by the input methods of the device. Without external hardware, such as a Bluetooth keyboard, all of the candidate tablets use a touchscreen for input. While a stylus may be used with a touchscreen, the application should not assume its presence. The application cannot assume the presence of any input methods other than a touchscreen.

The dimensions and resolutions of the candidate tablets vary, but they are, in all cases, limited. Even on devices with very high resolution displays, such as an iPad with a Retina display, the limited size of the screen will limit the amount of information being displayed at one time. In this regard, standard approaches to graphic design and typography must be used to ensure readability in all conditions by all users.

Each tablet contains different hardware features (cameras, GPS, etc.) with varying specific capabilities. The application cannot assume the presence of every hardware feature, and the behavior and performance of these cannot be guaranteed across platforms.

The application requires a network connection to access the bridge inspection database, but users will frequently be without such a connection. The application cannot assume the presence of a connection to function properly. Similarly, the application cannot assume that the availability of a network connection implies the availability of the database. If the database is unavailable, the application will only be able to store reports locally.

5.5 Assumptions and Dependencies

5.5.1 Application Platform Assumptions
The Wireless Bridge Inspection System (WBIS) must be portable and accessible in the field under varying weather and traffic conditions. The intent is to provide an application which, when paired with suitable hardware, will provide little to no encumbrance to the user. A potential
standard for comparison is any leading mobile application that has achieved great popularity and adoption in the mobile market. The platforms for today's mobile applications generally offer similar hardware support (e.g., 5-20 megapixel camera) but are significantly different in terms of the operating system. There are a small number of operating systems in wide use on so-called "smartphones" within the United States; in descending order of 2014 Q4 market share (according to Kantar Worldpanel\(^1\)), they are: Android (50.6%), iOS (43.9%), Windows Phone (4.3%), and Blackberry (0.4%). Market share estimates for tablets are harder to come by, but PC Magazine found that, in a 2012 survey (http://www.pcmag.com/article2/0,2817,2405972,00.asp), 52% of tablet owners possessed an iPad (iOS) while 51% of tablet owners possessed an Android model (these estimates do not add up to 100% as they include users who own more than one device type).

The primary considerations in selecting platform(s) for the WBIS software specifications are: 1) Maximizing platform availability for the application as measured by market share; 2) Ensuring that specific end-user devices are supported; and 3) Providing a consistent user experience across platforms. With regards to these considerations, the development team proposed to support the Android and iOS platforms, which combined represent 94.5% of the smartphone devices and each roughly half of the tablet devices currently on the market (2013 Q4 and 2012 estimates, respectively). Developing for these platforms should provide for a consistent user experience and will provide the end-user and customer, the Michigan Department of Transportation, with the fewest constraints on hardware selection and purchase due to the wide variety of choices afforded by these two platforms.

We assume that the target mobile device platforms will have the hardware necessary to support functional requirements, specifically:

- The mobile device is running either the iOS or Android operating system in good working order
- The mobile device will have a camera that can be activated by applications on the device
- The mobile device has a wireless broadband connection (either 3G or 4G) and a cellular data service plan
- The mobile device will only be used within the United States of America
- The mobile device has a touch screen and a virtual keyboard
- A GPS receiver is available on the device and installed applications have access to it

5.5.2 Bridge geometry assumptions

\(^1\) http://www.kantarworldpanel.com/dw1.php?se=news_downloads&id=399

MDOT Wireless Bridge Inspection System – Software Requirements Definition V1  Page 12
Regarding the rendered bridge geometry, for the first version of the Application, “skew” is ignored; that is all bridges are rendered “orthonormal” with 90 degree angles between joints and spans, piers perpendicular to the carried roadway, and no diaphragm offset.

The model schema is being designed to accommodate calculations deriving from skew, but we are ignoring skew for simplicity in this iteration.

Also, due to the fact that there are no indicators in BMS for any asymmetries, bridges are rendered as longitudinally and laterally symmetric based on “longest span length.” Many variances from symmetry can be achieved through the Bridge Customization step that is available when a user first visits a bridge.

6 Specific Requirements
6.1 System Interfaces
6.1.1 User Interfaces
The WBIS mobile application will facilitate the entry of CoRE/AASHTO inspection data through a virtual model of the bridge under inspection. This 2D or 3D model will allow the user to identify and select elements of the bridge based on touching or tapping the bridge model as it appears on a touchscreen. After selecting a CoRE/AASHTO element in this manner, an “enhanced presentation” (e.g. zoomed-in and/or panned view) of the element will allow for the delineation of spatially-explicit deterioration information by touching or tapping on the affected area of that element. The overall model can be rotated through constrained views using either direct touch manipulation or controls.

Elements can also be located through contextual menus including a search utility where matching results are filtered to a keyword search for the element’s name in real time. Selecting an element in this manner will present the user with the “enhanced presentation” of the model. The alternate “enhanced presentation” may include finer detail than the overall model, allowing for the user to precisely locate a certain area of the element, and will likely present a constrained or predefined viewing geometry.

In the “enhanced presentation,” users can tap or touch an area of the element to enter information about the dimensions and condition state of that area. This tap or touch interaction will launch a modal or space-filling window that describes the data to be entered, including the approximate shape of the area, the dimensions of the shape (which could be given in units of area or as the percentage of the overall element), and the condition state that area should be associated with. In all of the data entry interfaces, fields that have been changed will be styled in a certain way so as to indicate to the user that the field has changed (marked as “dirty”).
The virtual model of the bridge will be initialized from SI&A element-level information about the bridge deck, substructure, superstructure, approach, and railings; such information presented by the Bridge Model Schema. Other data is not available from the BMS database; namely fascia widths, pier/column shapes & diameters, beam shapes, pin and hanger locations (if any), and bearing locations (if pin & hangers) are used. Reasonable defaults based on the domain knowledge we have captured from MDOT and Tess Ahlborn at MTU will be used to render the “Initial Bridge Model.”

Additional user input will be required to customize the virtual model, however. User input will be gathered through a series of prompts which initialize to defaults based on best assumptions. After this is done once for any bridge, the user will be prompted only to accept or reject these settings upon viewing the same bridge again; rejecting these settings will require the user to enter the missing information required to render the virtual model of the bridge.

6.1.2 Hardware Interfaces

6.1.2.1 Specific Device Compatibility Assumptions

After the process of selecting UnrealEngine4 as our application deployment framework due to its 3d capabilities and User Interface extensibility, we learned that there is a reduced set of compatible Android devices as specified by the UnrealEngine4 developers. This list is growing weekly as the community tests deployment to various tablet platforms.

As of June 2014 the following table lists the developer-certified compatible chipsets. We also list specific devices which meet this requirement, as well as a cellular data capability and minimum memory and speed requirements.
Table 1: Table of Android device compatibility from UnrealEngine (the 3D Rendering Framework for the Tablet Application)

<table>
<thead>
<tr>
<th>Device</th>
<th>LDR</th>
<th>Basic Lighting</th>
<th>Full HDR</th>
<th>Full HDR w/ Scan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tegra</td>
<td>Expected</td>
<td>Expected</td>
<td>Unsupported</td>
<td>Unsupported</td>
</tr>
<tr>
<td>Adreno 320</td>
<td>Supported</td>
<td>Supported</td>
<td>Expected</td>
<td>Unsupported</td>
</tr>
<tr>
<td>Adreno 330</td>
<td>Supported</td>
<td>Supported</td>
<td>Expected</td>
<td>Unsupported</td>
</tr>
<tr>
<td>Mali 400</td>
<td>Expected</td>
<td>Expected</td>
<td>Expected</td>
<td>Expected</td>
</tr>
</tbody>
</table>

The following table lists individual devices we have tested:

<table>
<thead>
<tr>
<th>Device</th>
<th>LDR</th>
<th>Basic Lighting</th>
<th>Full HDR</th>
<th>Full HDR w/ Scan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Galaxy S4 (Adreno 320)</td>
<td>Supported</td>
<td>Supported</td>
<td>Expected</td>
<td>Unsupported</td>
</tr>
<tr>
<td>Nexus 5 (Adreno 330)</td>
<td>Supported</td>
<td>Supported</td>
<td>Supported</td>
<td>Supported [1]</td>
</tr>
</tbody>
</table>

1: The Nexus 5 with the latest publicly available driver performs poorly when using features from the Full HDR tier. We have been working closely with Qualcomm in this area and they have developed faster drivers that remove the bottlenecks we were running into. Our HDR features are fully supported on their latest internal drivers which we hope will be available to the public soon.

2: Similar to the Nexus 5, the Kindle Fire HDX runs into some bottlenecks in the Adreno 330 driver and we expect it to perform well in the future with an updated driver.

Note that there are more Adreno devices than those listed in the second table; those represent the ones UE4 has tested inhouse.

Below is a table which identifies compatible tablet devices, as of June 2014.
Table 2: June 2014 listing of recommended development and "beta" Android devices

<table>
<thead>
<tr>
<th>Device</th>
<th>GPU</th>
<th>Data</th>
<th>Screen size</th>
<th>RAM/ROM/ Storage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Xplore RangerX Rugged Tab</td>
<td>Integrated PowerVR SGX544 w/ ISP</td>
<td>3G</td>
<td>10.1</td>
<td>1GB (7)/ 32GB ROM/ 128GB</td>
</tr>
<tr>
<td>Samsung Galaxy Tab SM-t325 4G Lte (Tab Pro 8.4)</td>
<td>Adreno 330</td>
<td>4G</td>
<td>8.4</td>
<td>2GB / 16GB ROM / uSD / 64GB</td>
</tr>
<tr>
<td>Samsung Galaxy Note Pro 12.2 (MSM8974A/4W2)</td>
<td>Adreno 330 405MHz</td>
<td>3G/4G</td>
<td>12.2</td>
<td>3GB / 32/64GB / uSD / 64GB</td>
</tr>
<tr>
<td>Samsung Galaxy Tab Pro</td>
<td>Adreno 330 405MHz</td>
<td>3G/4G</td>
<td>10.1</td>
<td>3GB (eDRAM) / 16/32 / 64GB</td>
</tr>
<tr>
<td>Fujitsu Arrows</td>
<td>Adreno 330 405MHz</td>
<td>3G/4G</td>
<td>10.1</td>
<td>2GB / 64GB / uSD (7)</td>
</tr>
<tr>
<td>Amazon Kindle Fire HDX 8.9</td>
<td>Adreno 330 405MHz</td>
<td>3G/4G</td>
<td>8.9</td>
<td>2GB / 64GB / none(7)</td>
</tr>
<tr>
<td>Sony Xperia Z2 MSM8974Ab v3</td>
<td>Adreno 330 578MHz</td>
<td></td>
<td>10.1</td>
<td>3GB</td>
</tr>
</tbody>
</table>

The two recommended devices are the Xplore RangerX series of ruggedized tablets, with obvious field advantages but higher cost. The second recommendation is the Samsung Galaxy Tab Pro line, a top consumer product with superior screen brightness and required graphics chipset.

6.1.3 Communication Interfaces

The mobile application will communicate with the remote server, nominally the surrogate Bridge Management System (sBMS) or canonical Bridge Management System (BMS), over the web through normal HTTP requests. Each request for retrieving from or submitting data to the server will therefore take the form of a Uniform Resource Identifiers (URI). The practice of implementing and using this type of communication interface is sometimes referred to as Representative State Transfer (REST). When requests originate in the Secure Socket Layer (SSL) over HTTPS, these requests and responses are securely encrypted between the user's device and the remote server.

The REST API will describe completely the available data operations that can be conducted between the mobile application on a tablet in the field and the sBMS/BMS on a remote server. Operations such as retrieving past inspection reports or uploading completed inspection reports will each have a unique URI through which a connection is made between the mobile application on a tablet device and the remote server. Documents or data objects available through the REST API are generally called resources. Some of the resources will be read-only whereas others will be resources that can be created on (uploaded to) the remote server.
### 6.2 Functional Requirements Summary Table

The following table lists the major capabilities of the 3dWBIS System.

Subsequent versions of this requirements document will expand each requirement.

<table>
<thead>
<tr>
<th>Req</th>
<th>Requirement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.2.1</td>
<td>System presents user with 3D representation of bridge under inspection</td>
<td>Based on a bridge model schema encoding a plurality of expected highway bridge types, a 3D version of a specific bridge model is displayed for interaction, with sufficient detail to enable inspection and markup of all AASHTO bridge elements</td>
</tr>
<tr>
<td>3.2.2</td>
<td>User can interact with model using native (touch) controls as well as intuitive navigation controls</td>
<td>Model views can be manipulated (in a set of constrained views) such that all AASHTO elements are accessible</td>
</tr>
<tr>
<td>3.2.3</td>
<td>User can view major bridge element groupings</td>
<td>Bridge elements are grouped into Superstructure, Substructure, Deck.</td>
</tr>
<tr>
<td>3.2.3</td>
<td>User can select individual AASHTO elements for markup</td>
<td>AASHTO element will be isolated or ‘zoomed’ in an ‘enhance view’ in order to expose representations of physical attributes that are rated by inspector</td>
</tr>
<tr>
<td>3.2.4</td>
<td>User can indicate spatially localized data within a single instance of an AASHTO element</td>
<td>Graphical ‘indication’ of inspected locale on AASHTO element via native interaction with 3D model element by touching approximate location on representation</td>
</tr>
<tr>
<td>3.2.5</td>
<td>User can enter text data and associate it with an AASHTO element or spatial subset of an AASHTO element</td>
<td>AASHTO element fields can be accessed from either a listing drop-down type input field or the graphical-object navigation described in 3.2.3</td>
</tr>
<tr>
<td>3.2.6</td>
<td>System aggregates individual ratings across multiple instantiations of elements for the reported percentage condition state rating</td>
<td>In cases where AASHTO elements represent a collective entity with a single reported rating (e.g., piers), the user enters individual quantities on a per-instance basis, and the system adds the quantities for each instance to a collective value. The user must enter the quantitative “amount” value for the condition state (e.g., square feet)</td>
</tr>
<tr>
<td>3.2.7</td>
<td>System generates a AASHTO CoRE Element Report</td>
<td>System aggregates and summarizes element-level condition state data to populate the familiar Element Inspection Form</td>
</tr>
<tr>
<td>3.2.8</td>
<td>User can take photos and attach to an inspection event</td>
<td>Native tablet camera hardware can be used to take photos at the bridge site during inspections. These photos are “linked” to the Bridge / Location / Element and Date of the Inspection Event, and can be retrieved and viewed by Inspectors at any time. The</td>
</tr>
</tbody>
</table>
6.3 Use Cases

6.3.1 Prototype Use Case
Structure number 10922, S13-81003: Curtis Road over M-14

1. Inspector selects bridge by structure number
2. System requests data from server
3. Server generates Bridge Model Document
4. Server sends Model.xml and (additional bms data) to System
5. System validates document
6. System renders 3D Bridge Representation
7. User customizes model
8. User inspects bridge in “walkthrough” mode
9. User enters condition states on a per-element-instance basis
10. System calculates overall (aggregate) condition states
11. System generates Report
12. System posts AASHTO form data to sBMS
13. System saves 3D element data and updated XML model

6.4 Quality Requirements

6.4.1 Performance
Transactions between the application and the remote server will be performed in the background (asynchronously) whenever possible. In such cases, the wait time experienced by the user will be zero. Actual wait times will vary with aspects of the wireless broadband connection (e.g. 3G versus 4G, signal strength).  

6.4.2 Reliability
- Recovery from software or hardware failure must be robust to protect the time investment of inspectors
- Inspection reports will be automatically saved (“autosaved”) to the device
- Additionally, inspection reports will be automatically backed up to the remote server (sBMS or BMS)
- The application software should reasonably reduce CPU usage to promote long battery life in a mobile environment.