

The background of the page features an abstract graphic design. It consists of three concentric circles in shades of blue, arranged diagonally from the top right towards the bottom right. Two thin, light blue lines intersect the circles, creating a sense of depth and movement. The overall aesthetic is clean and modern.

# Software Design Description

Senior Project

**Matthew King**  
**6/25/2010**

Hillsborough County Historical Mapping	Version: 1.2
Software Project Management Plan	Date: 02 November 2010

## Revision History

Date	Version	Description	Author
25 June 2010	1.0	Software Design Document for Hillsborough County Historical Mapping	Matthew King
16 July 2010	1.1	Software Design Document for Hillsborough County Historical Mapping	Matthew King

## Table of Contents

<b>1. Introduction</b>	<b>4</b>
1.1. Purpose	4
1.2. Project Scope	4
1.3. Definitions, acronyms, and abbreviations	4
1.4. References	4
1.5. Overview	4
<b>2. Design Considerations</b>	<b>4</b>
2.1. Assumptions	4
2.2. General Constraints	5
2.3. System environment	5
<b>3. Architectural Design</b>	<b>5</b>
3.1. Web Architecture	5
3.2. Block Diagram	6
3.3. Data Design	6
3.3.1. Data Description	6
3.3.2. Maps Database	6
3.3.3. Data Dictionary of Maps table	7
3.3.4. Blog Database	9
3.3.5. Data Dictionary of Blog table	11
<b>4. User Interface Design</b>	<b>12</b>
4.1. Section Overview	12
4.2. Web Pages	12
4.2.1. Home Page	12
4.2.2. Maps Page	13
4.2.3. Blog (Web Log) Page	14
4.2.4. Book Page	16
<b>5. Glossary</b>	<b>18</b>
5.1. Acronyms and Abbreviations	18
5.2. Definitions	18

## **Introduction**

The project is to create an interactive Google<sup>®</sup> map to view historical information in Hillsborough County.

### **1.1. Purpose**

The primary purpose of this document is to present a detailed description of the design elements of the Hillsborough County Historic Maps. This will guide me in the design of the application

### **1.2. Project Scope**

This document will provide details on the design of the various pages of the web application, in particular the Maps page and the Blog page which interact with the MySQL relational database using various scripts.

User will have the ability to view Hillsborough County and interact with various aspect of its history. The user will be able to view historical overlay maps from various years, photos and historical notes. Users will also be able to view Blogs within the application as well as participating and signing up with the Wordpress<sup>®</sup> Blog by linking to it external.

### **1.3. Definitions, acronyms, and abbreviations**

Definitions, Acronyms and Abbreviations are defined in the Glossary, see Section 7.

### **1.4. References**

IEEE Standard 1016-2009, IEEE Standard for Information Technology – Systems Design – Software Design Description, IEEE Computer Society, 2009.

Google<sup>®</sup> Maps API for Flash - <http://code.google.com/apis/maps/documentation/flash/>, Google<sup>®</sup> 2009.

### **1.5. Overview**

This document will go over the design of various aspects of the project including architectural, software components, interface, and database design.

## **2. Design Considerations**

### **2.1. Assumptions**

The user of the web application will have the Adobe<sup>®</sup> Flash plug-in installed along with a current multi-media web browser to view the application.

## 2.2. General Constraints

The application will be completed using the Adobe Flex® architecture and a MySQL relational database. The developer of the application requires Adobe Flex® 3.0, a local web server for testing to include Apache Web Server 2.0, PHP 5.0 and MySQL 5.1 and WordPress database installed. The designer will also need a vector and/or raster image editor such as Adobe® Illustrator/InkScape or Adobe® Photoshop/GIMP. For publishing the application the provider must have Apache Web Server 2.0, PHP 5.0, and MySQL 5.1 or higher and have the ability to include WordPress® Blog.

## 2.3. System environment

The application is designed to work on all operating systems that can support a current media enabled web browser and the Adobe® Flash plug-in.

## 3. Architectural Design

### 3.1. Web Architecture

The Hillsborough County Historic Mapping application will run on a client-server system architecture where the client will connect to a web server using a current internet browser and TCP/IP connection. The web server will be hosted by GoDaddy® with a Yahoo® web server as backup. The web server is maintained by the Apache® web server software on a Linux operating system. The web server will communicate with a MySQL database and interact with Google® Maps API using client and server side scripting.

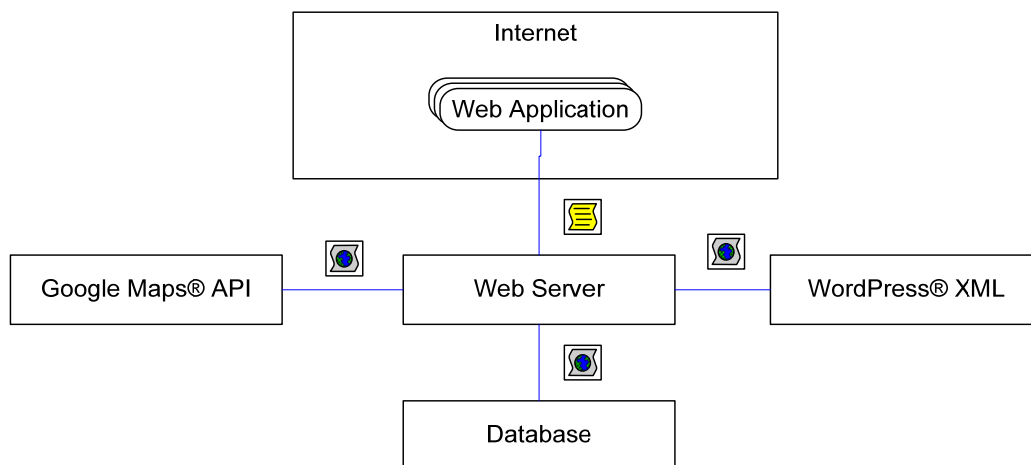


Figure 1

### 3.2. Block Diagram

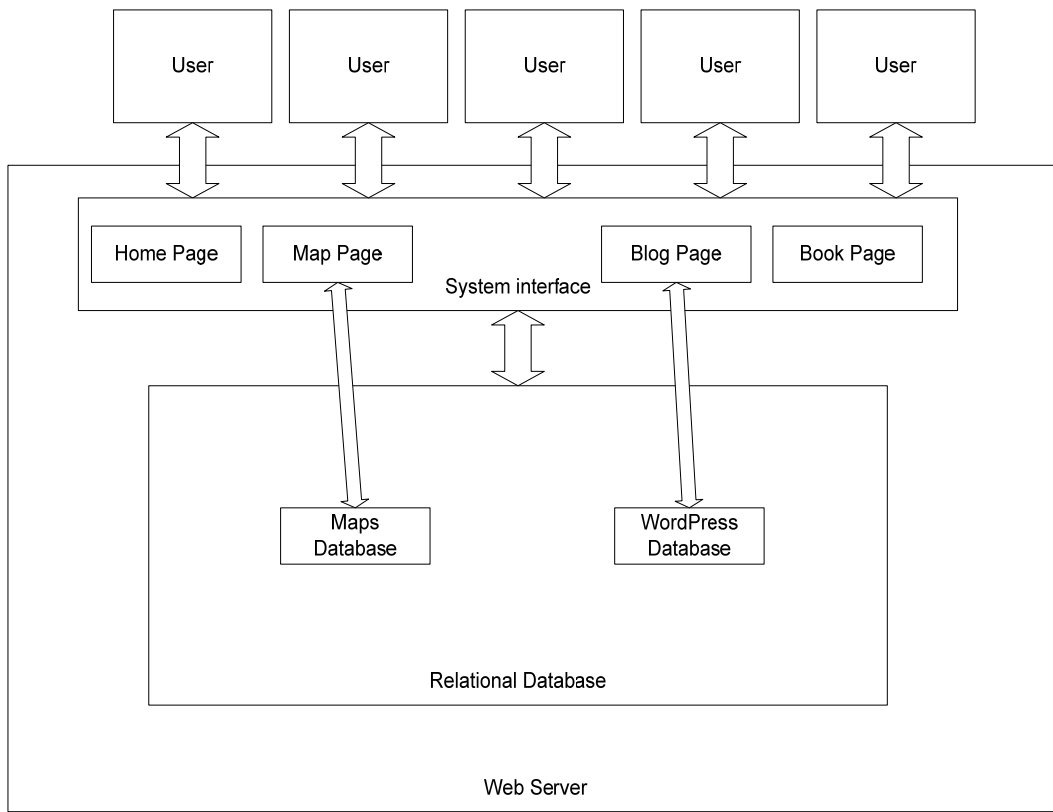


Figure 2

### 3.3. Data Design

#### 3.3.1. Data Description

MySQL database connects with Adobe® Flex ODBC connection using PHP Scripts installed on the local web server.

#### 3.3.2. Maps Database

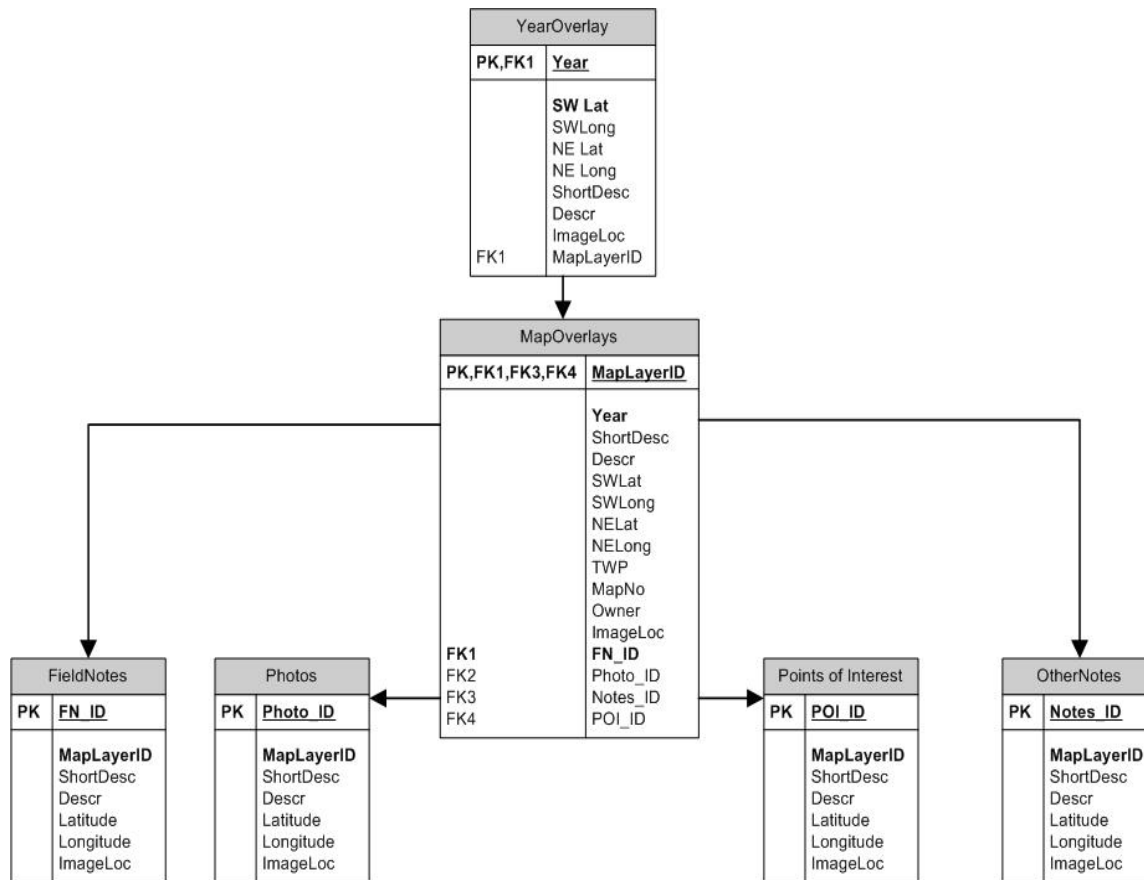


Figure 3

### 3.3.3. Data Dictionary of Maps table

- 3.3.3.1. Year
  - 3.3.3.1.1. Year
  - 3.3.3.1.2. Year of Overall Map
  - 3.3.3.1.3. Integer
  - 3.3.3.1.4. 4 digits
  - 3.3.3.1.5. 1500 – Current year
- 3.3.3.2. Latitude/Longitude
  - 3.3.3.2.1. SWLat, SWLong, SELat, SELong, NWLat, NWLong, NELat, NELong
  - 3.3.3.2.2. Corresponding Latitude and Longitude for point
  - 3.3.3.2.3. Double/Float
  - 3.3.3.2.4. 10 digits (4,6)
  - 3.3.3.2.5. Latitude (-90.000000 – 90.000000)  
Longitude (-180.000000 – 180.000000)
- 3.3.3.3. Short Description
  - 3.3.3.3.1. ShortDesc
  - 3.3.3.3.2. Short name for item
  - 3.3.3.3.3. Varchar/Text
  - 3.3.3.3.4. 50 characters
  - 3.3.3.3.5. 0-50 characters
- 3.3.3.4. Long Description
  - 3.3.3.4.1. Descr
  - 3.3.3.4.2. Long description of item
  - 3.3.3.4.3. Varchar/Text
  - 3.3.3.4.4. 255 characters
  - 3.3.3.4.5. 0-255 characters
- 3.3.3.5. Image Location
  - 3.3.3.5.1. ImageLoc
  - 3.3.3.5.2. Location of image on web server
  - 3.3.3.5.3. Varchar/Text
  - 3.3.3.5.4. 255 characters
  - 3.3.3.5.5. 0-255 characters
- 3.3.3.6. Map Layer ID
  - 3.3.3.6.1. MapLayerID
  - 3.3.3.6.2. Index of the Map Layer (Non-Null)
  - 3.3.3.6.3. Integer
  - 3.3.3.6.4. 6 digits (TWP and MapNo)
  - 3.3.3.6.5. 000000-999999
- 3.3.3.7. Township
  - 3.3.3.7.1. TWP
  - 3.3.3.7.2. Township Number
  - 3.3.3.7.3. Integer
  - 3.3.3.7.4. 4 digits



	3.3.3.7.5.	0000-9999
3.3.3.8.	Map Number	
	3.3.3.8.1.	MapNo
	3.3.3.8.2.	Map Number within Township
	3.3.3.8.3.	Integer
	3.3.3.8.4.	2 digits
	3.3.3.8.5.	00-99
3.3.3.9.	Owner	
	3.3.3.9.1.	Owner
	3.3.3.9.2.	User who updated the database
	3.3.3.9.3.	Varchar/Text
	3.3.3.9.4.	25 characters
	3.3.3.9.5.	0-25 characters
3.3.3.10.	Field Note ID	
	3.3.3.10.1.	FN_ID
	3.3.3.10.2.	Auto Number Field Note ID
	3.3.3.10.3.	Integer
	3.3.3.10.4.	4 digits
	3.3.3.10.5.	0000-9999
3.3.3.11.	Photo ID	
	3.3.3.11.1.	Photo_ID
	3.3.3.11.2.	Auto Number Photo ID
	3.3.3.11.3.	Integer
	3.3.3.11.4.	4 digits
	3.3.3.11.5.	0000-9999
3.3.3.12.	POI ID	
	3.3.3.12.1.	POI_ID
	3.3.3.12.2.	Auto Number Point of Interest ID
	3.3.3.12.3.	Integer
	3.3.3.12.4.	4 digits
	3.3.3.12.5.	0000-9999
3.3.3.13.	Notes ID	
	3.3.3.13.1.	Notes_ID
	3.3.3.13.2.	Auto Number Note ID
	3.3.3.13.3.	Integer
	3.3.3.13.4.	4 digits
	3.3.3.13.5.	0000-9999

### 3.3.4. Blog Database

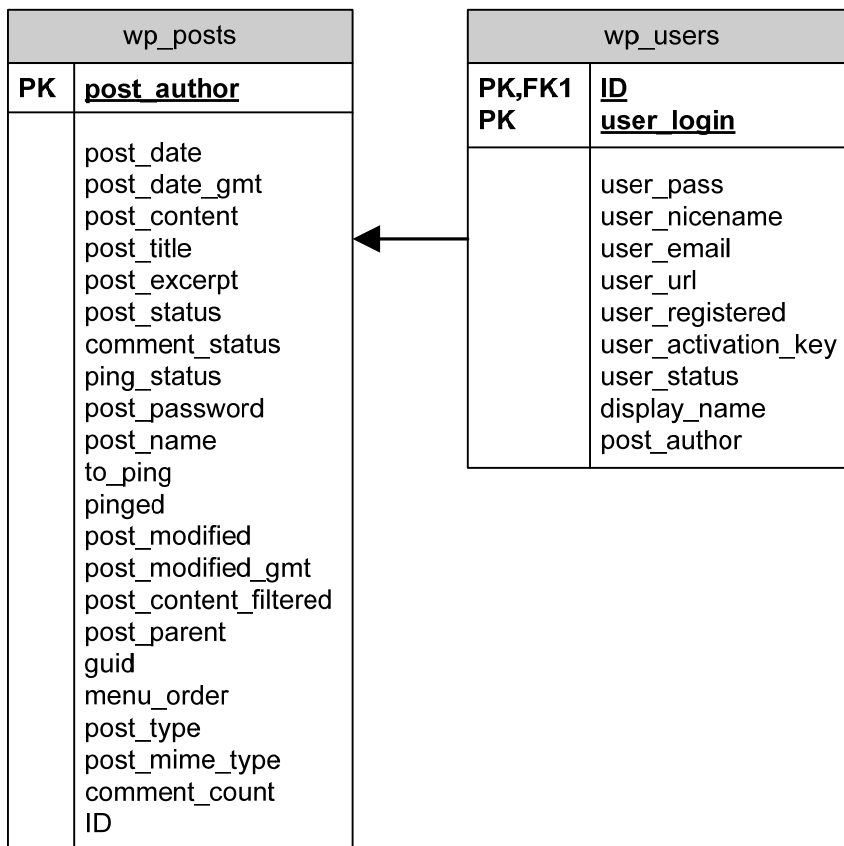


Figure 4

### 3.3.5. Data Dictionary of Blog table

3.3.5.1.	ID
3.3.5.1.1.	ID
3.3.5.1.2.	Auto Number ID of Post
3.3.5.1.3.	Integer
3.3.5.1.4.	20 digits
3.3.5.1.5.	0-9999999999999999999
3.3.5.2.	Author
3.3.5.2.1.	post_author
3.3.5.2.2.	Author of Post ID (from UserId in ps_users table)
3.3.5.2.3.	Integer
3.3.5.2.4.	20 digits
3.3.5.2.5.	0-9999999999999999999
3.3.5.3.	Post Date
3.3.5.3.1.	post_date
3.3.5.3.2.	Date of posting
3.3.5.3.3.	DateTime
3.3.5.3.4.	YYYY-MM-DD HH:MM:SS
3.3.5.3.5.	0000-00-00 00:00:00
3.3.5.4.	Post content
3.3.5.4.1.	post_content
3.3.5.4.2.	Content of the Post
3.3.5.4.3.	Varchar/Text
3.3.5.4.4.	255 characters
3.3.5.4.5.	0-255 characters
3.3.5.5.	Post Title
3.3.5.5.1.	post_title
3.3.5.5.2.	Title of Post
3.3.5.5.3.	Varchar/Text
3.3.5.5.4.	50 characters
3.3.5.5.5.	0-50 characters
3.3.5.6.	Post Status
3.3.5.6.1.	post_status
3.3.5.6.2.	Status of Post
3.3.5.6.3.	Varchar/Text
3.3.5.6.4.	20 characters
3.3.5.6.5.	0-20 characters
3.3.5.7.	Post Name
3.3.5.7.1.	post_name
3.3.5.7.2.	Name of Post
3.3.5.7.3.	Varchar/Text

- 3.3.5.7.4. 200 characters
- 3.3.5.7.5. 0-200 characters
- 3.3.5.8. Post Modified
  - 3.3.5.8.1. post\_modified
  - 3.3.5.8.2. Date post modified
  - 3.3.5.8.3. DateTime
  - 3.3.5.8.4. YYYY-MM-DD HH:MM:SS
  - 3.3.5.8.5. 0000-00-00 00:00:00

## 4. User Interface Design

### 4.1. Section Overview

This section provides an example of what the interface will look like for each web page to the user.

### 4.2. Web Pages

#### 4.2.1. Home Page



Figure 5

#### 4.2.1.1. Description

The Home page provides you with links to all the other pages within the web application and displays an overview of what is expected of the application.

#### 4.2.2. Maps Page

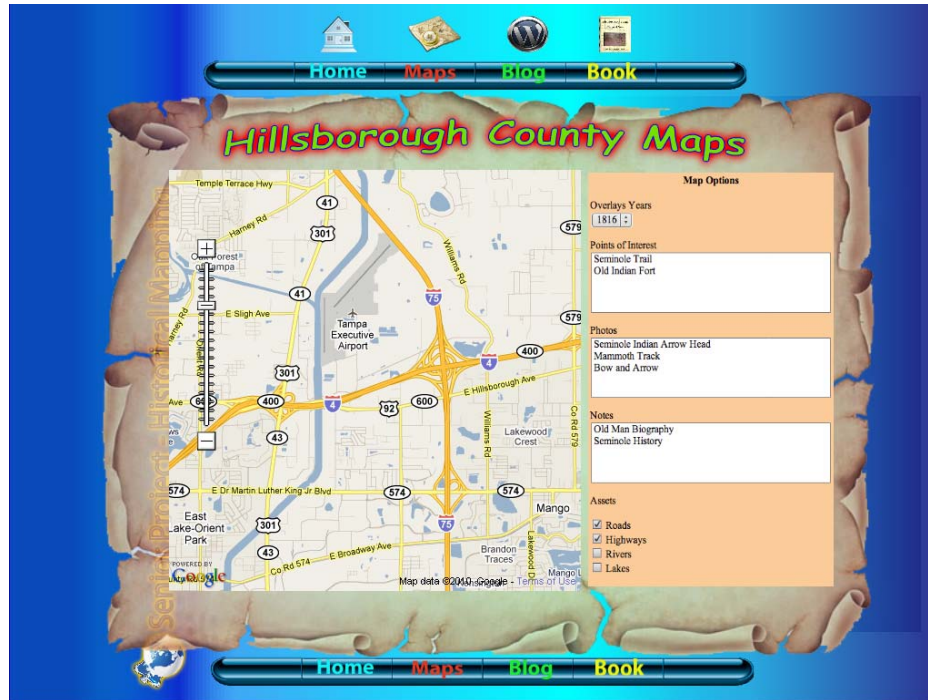


Figure 6

##### 4.2.2.1. Description

The Maps page is the meat of the application. It provides an interactive Google® map to the left and options to be displayed on the map on the right side of the page. Features to display on the map include a year overlay, field notes, points of interest, photos and other information related to that year.

The Google® map is displayed by using the Google® map API provided by Google® for Flash application development.

##### 4.2.2.2. Data members

Year – Selecting the Year will display an overall map for that particular year sectioned off by grids based on Township, Range and MapNo (2919-01 is equivalent to Township 29, Range 19, Map No 01) and will be displayed in the Map Overlays section of the form.

Map Overlay – Selecting a particular grid on the map or in the form will provide you with the field notes, points of interest, photos and other notes related to that overlay.

Field Notes – Available field notes will display in the form for the overlay selected and as green thumbtacks on the Google® map.

Points of Interest – Available points of interest, if there are any, will be displayed in the form and on the map as a blue thumbtack.

Photos – Available photos will be displayed on the form for selection and on the map as yellow thumbtacks.

Other Info – If there are other info that do not apply above it will be displayed on the map as an orange thumbtack and displayed on the form.

#### **4.2.2.3. Methods**

Year Select – When a year is selected a PHP script will connect to the database and display the image on the map with the available overlays displayed in the form.

Overlay Select – When an overlay grid is selected on either the map or the form a PHP script connects to the database and retrieves all related field notes, points of interests, photos and/or other notes.

Opacity Slider – Another feature will allow the user to control the opacity of the overlay image on the map. When the slider is moved a PHP scripts updates the opacity by manipulating the XML related to that image.

### **4.2.3. Blog (Web Log) Page**

#### **4.2.3.1. Recent Blogs**



Figure 7

#### 4.2.3.2. Description

The Blog Page displays the most current posts from the external Historical Maps WordPress® blog. A list of available blogs will be displayed with the name of the poster, date and short description. A display window will show the selected post with its contents.

A link is also provided to connect to the external WordPress® Blog where a user can request to participate or view other web logs.

#### 4.2.3.3. Data Members

Post Author – Displays the username of poster of blog.

Post Date – Displays the date the post was created.

Post Title – Displays the title of the post

Post Content – Displays the content of the post in a separate display window.

#### 4.2.3.4. Methods

Post Select – Selecting a specific post in the selection window calls a PHP script which pulls the post content from the database and displays in the post window.

#### 4.2.3.5. External Website

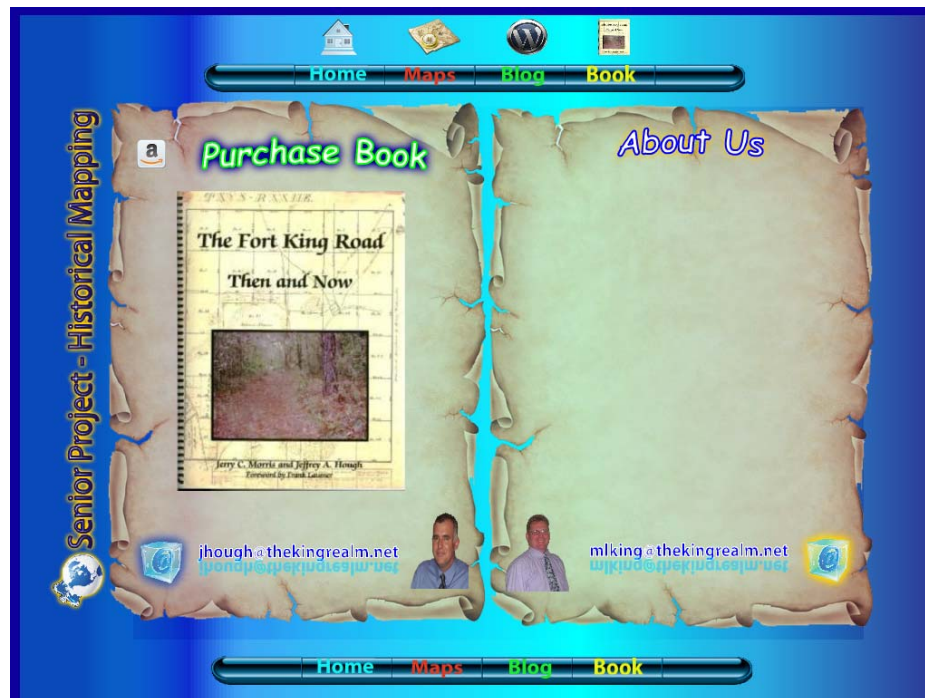


##### 4.2.3.5.1. Description

This is the linked external website created with WordPress® and installed on the web server. The administrator can access and administer the Blog from this website, approve users, and update posts as needed.

#### 4.2.4. Book Page





#### 4.2.4.1. Purchase Book

##### 4.2.4.1.1. Description

The Purchase Book (left) section of this page allows a user to link to Amazon to purchase the text 'The Fort King Road Then and Now' by Jerry Morris and Jeff Hough. A link is also provided to e-mail one of the authors of the book.

##### 4.2.4.1.2. Methods

Selecting the Amazon icon or the text will link you to an external Amazon website for purchasing the text.

Selecting the e-mail link will bring up the users default e-mail program allowing the user to e-mail one of the authors of the book.

#### 4.2.4.2. About Us

##### 4.2.4.2.1. Description

The About Us (right) section of this page allows a user to view information about one of the author's, as well as, the developer of the site/application.

##### 4.2.4.2.2. Methods

Selecting the e-mail link will bring up the users default e-mail program allowing contact with the developer.

## **5. Glossary**

### **5.1. Acronyms and Abbreviations**

AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
Blog	Web log
CSS	Cascading Style Sheet
DOM	Document Object Model
FPU	Floating Point Unit
HTML	Hypertext Markup Language
KML	Keyhole Markup Language
MXML	Macromedia Extensible Markup Language
PHP	PHP Hypertext Preprocessor
QA	Quality Assurance
RDBMS	Relational Database Management System
RIA	Rich Internet Application
SDK	Software Development Kit
XHTML	Extensible Hypertext Markup Language
XML	Extensible Markup Language
XSLT	Extensible Stylesheet Language

### **5.2. Definitions**

#### **Action Script**

A scripting language, similar to JavaScript, used primarily for developing Adobe® Flash applications.

#### **Apache Web Server**

Web server software that allows the publishing of web sites.

### **Application Programmer Interface**

Programming interface that enables one software interface to interact with another.

### **Asynchronous Java Script and XML**

A way of developing web applications using a combination of XHTML, CSS, DOM, XML, XSLT, Asynchronous data retrieval via XMLHttpRequest, and JavaScript.

### **Cascading Style Sheet**

Style sheet language that allows for formatting and layout of web sites.

### **Document Object Model**

A web interface that provides a way for programs and/or scripts to update the content, structure, and style of web documents dynamically.

### **Flash**

A development environment created by Adobe used to create and view animation programs for the internet.

### **Flex**

A development environment created by Adobe used to create Flash based applications from a programmer (vs. designer) perspective using MXML and Action Script.

### **Hypertext Markup Language**

A set of tags and rules used to publish and format websites.

### **Java**

A programming language designed for multi-platform development based on C++.

### **JavaScript**

A web scripting language used to make web pages more dynamic based on the Java language.

### **Keyhole Markup Language**

A language created to express geographic information used with various applications including Google® maps and Google® Earth based on the Extensible Markup Language.

### **Macromedia Extensible Markup Language**

A language created by Macromedia based on the Extensible Markup Language used for developing Flash Applications.

### **MySQL**

A relational database management system owned by Sun Microsystems® and used on the Apache Web Server.

### **Object-Oriented**

A programming paradigm based on objects and data.

### **Open-Source**

Technology in which source code is freely available for use.

### **PHP**

A scripting language used to interact with a web server to provide an interactive, dynamic web experience.

### **Python**

An interpreted, object-oriented programming language.

### **Rich Internet Application**

An application similar to a desktop application but for use with the web and web technologies.

### **Ruby**

An open-source object oriented programming language that can be used on multiple platforms.

### **Ruby on Rails**

An open-source web application framework used to develop for the web based on the Ruby language.

### **Software Development Kit**

Development tools used to create various software packages.

### **Web Log**

A way for individuals to provide information, similar to a journal, to the public via the web.

### **XMLHttpRequest**

An application programming interface that transfers data between a client and server.

**XML**

Language used to represent data in a standard format.

**XSLT**

A web technology used to add styles to XML and other marked up text.