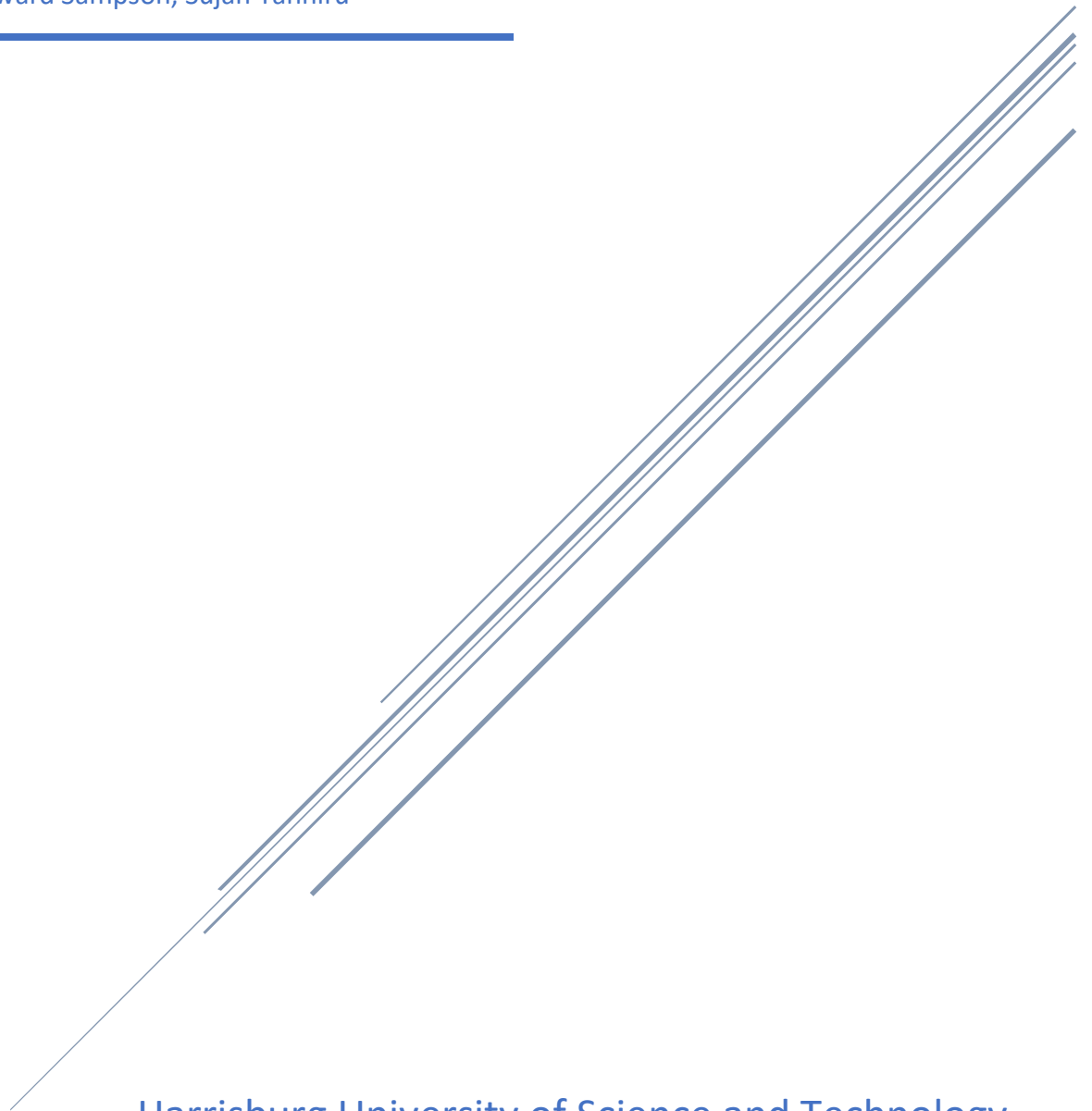


SOFTWARE DESIGN DOCUMENTATION

PA Historical Finder

By Shannon Williams, Daniel Malinsky, Raekwon
Harley, Edward Sampson, Sujan Tanniru



Harrisburg University of Science and Technology
CISC 397, Spring 2018

Table of Contents

1.	Introduction.....	2
1.1.	Purpose of this Specification Document.....	2
1.2.	Glossary	2
1.3.	Scope of the Product.....	2
1.4.	Overview of the Remainder of the Document.....	2
2.	Software Architecture	2
3.	Deployment.....	3
3.1.	Hardware.....	3
3.2.	Software Prerequisites	3
3.3.	Operating Environment.....	3
3.3.1.	Technology/Tools Usage Licenses.....	3
3.4.	Installing and Configuring the Web Application.....	4
3.5.	Installing MongoDB	4
	Related Licenses.....	5

1. Introduction

1.1. Purpose of this Specification Document

This document specifies the requirements for the software, being developed. The system, referred to as PA Historical Finder, is one that enables users to explore significantly historical information within Pennsylvania.

1.2. Glossary

- **PA Historical Finder** - Name of the system being built
- **System** – used interchangeably with software being built

1.3. Scope of the Product

The purpose of the software is to allow users to search for historical information and to view search results in an interactive interface.

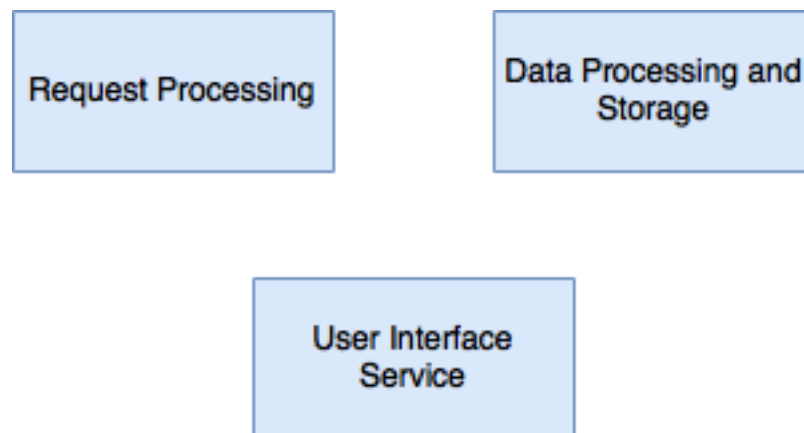


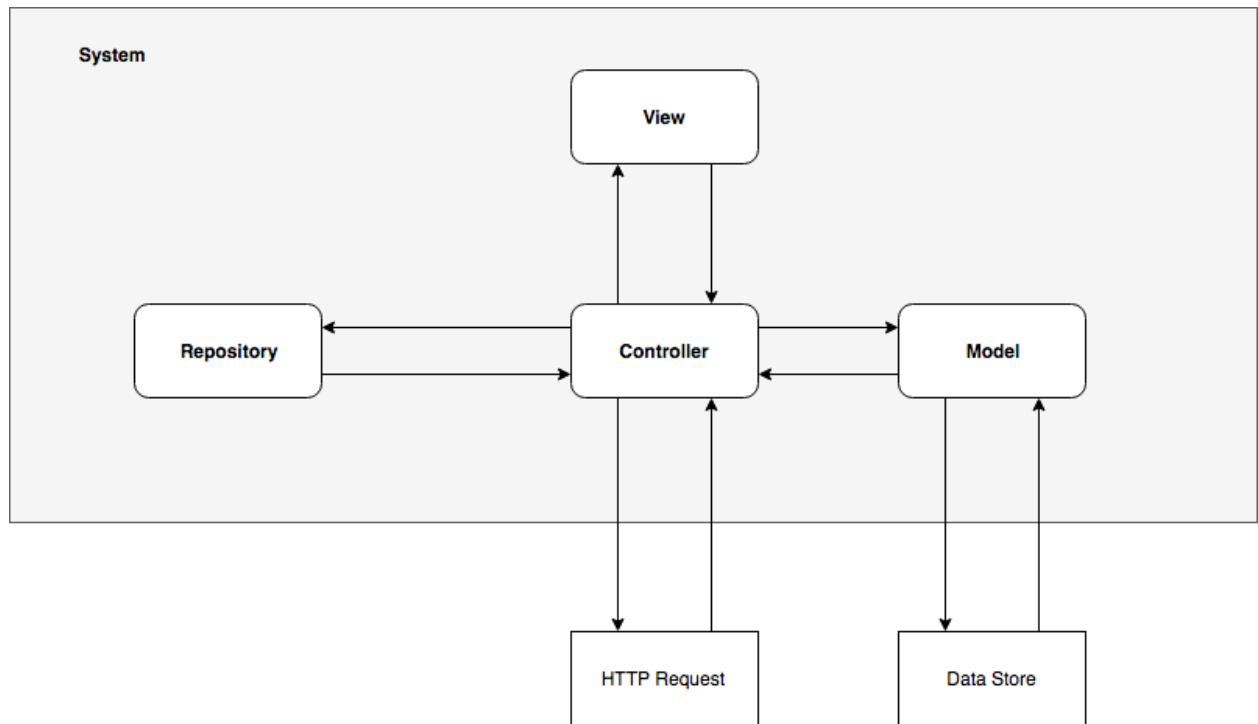
Figure 1.0

1.4. Overview of the Remainder of the Document

The latter part describes the system's architectural design.

2. Software Architecture

The system is designed using Model-View-Controller (MVC) as the foundation. Additional components, like a repository, can be built on top of this MVC architecture.



3. Deployment

3.1. Hardware

At minimum, a server is needed to host the web application and MongoDB. Hosting the website and MongoDB on separate machines is also a choice and the set-up is quite similar in regard to software installation and configuration.

3.2. Software Prerequisites

Before beginning installation of software, the host machine needs to have:

- Python 3.6+ installed
- NodeJS and NPM installed (*latest versions are recommended*)

3.3. Operating Environment

The application will be written in web technologies, supplemented with MongoDB, which will be hosted on a Windows or Linux based machine on separate or same machines respectively. The user simply needs access to a web browser to use the application.

3.3.1. Technology/Tools Usage Licenses

- **MongoDB:** GNU APL v3.0 License [1]
- **NodeJS:** "...permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy,

modify, merge, publish, distribute, sublicense, and/or sell copies of the Software...”
[2]

- **Select2:** MIT License [3]
- **Google Map JS API:** Google Map APIs Terms of Service [4]

3.4. Installing and Configuring the Web Application

After downloading the project files on the host machine, you simply need to:

1. **Installing Dependencies:** In the command line or terminal, cd into the project root directory and run the command `npm install --save`. Doing so will install all the dependencies for the web application.
2. **Environment URL:** In the project’s root folder, navigate to `views>shared>footer.ejs` file and look for the `EnvironmentUrl` variable near the end and change it to your server’s URL for HTTP traffic.
3. **Google reCaptcha:** After you have setup reCaptcha API keys in the admin dashboard, which at the time is <https://www.google.com/recaptcha/admin>, simply change the `secretKey` to one in the admin dashboard. The `secretKey` is found in post request handler for “/submitlogin” in the `server.js` file.

3.5. Installing MongoDB

After choosing the host machine for the MongoDB instance, you simply need to:

1. **Install MongoDB:** Using the normal installation methods as available on <https://docs.mongodb.com/manual/installation/>.
2. **Load the Landmark Dataset:** Ensure to run the command `pip install pymongo`, then change the `client` variable’s connection parameter to point to your instance on MongoDB.

Related Licenses

- [1] F. S. Foundation, "GNU Affero General Public License," [Online]. Available: <http://www.gnu.org/licenses/agpl-3.0.html>.
- [2] N. Foundation. [Online]. Available: <https://raw.githubusercontent.com/nodejs/node/master/LICENSE>.
- [3] K. Brown. [Online]. Available: <https://github.com/select2/select2/blob/master/LICENSE.md>.
- [4] Google, "Google Map APIs Terms of Service," 4 January 2018. [Online]. Available: <https://developers.google.com/maps/terms>.