**Software Requirements Specification**

**for**

**Modernizing PD Legacy Repositories**

**Version 1.1 approved**

**Prepared by** Javier Garcia,

Adrian Palomares,

Rawad Moussa,

Marlito Refuerzo Jr,

Christopher Rodriguez,

Tabassuma Torosa,

Paul Ube,

Audelia Valdovinoz,

Pierce Wei

**Office of Public Defender, LA County**

**6 November, 2020**

**Table of Contents**

Table of Contents................................................................................................................. pg 2

Revision History................................................................................................................... pg 3

1. Introduction................................................................................................................ pg 4

1.1. Purpose........................................................................................................... pg 4

1.2. Intended Audience and Reading Suggestions................................................ pg 4

1.3. Product Scope................................................................................................ pg 4

1.4. Definitions, Acronyms, and Abbreviations .................................................. pg 4

1.5. References...................................................................................................... pg 5

2. Overall Description.................................................................................................... pg 6

2.1. System Analysis…......................................................................................... pg 6

2.2. Product Perspective........................................................................................... pg 6

2.3. Product Functions........................................................................................... pg 6

2.4. User Classes and Characteristics.................................................................... pg 6

2.5. Operating Environment.................................................................................. pg 7

2.6. Design and Implementation Constraints........................................................ pg 7

2.7. User Documentation...................................................................................... pg 7

2.8. Assumptions and Dependencies.................................................................... pg 7

2.9. Apportioning of Requirements...................................................................... pg 7

3. External Interface Requirements............................................................................... pg 7

3.1. User Interfaces............................................................................................... pg 8

3.2. Hardware Interfaces....................................................................................... pg 8

3.3. Software Interfaces........................................................................................ pg 8

3.4. Communications Interfaces........................................................................... pg 8

4. Requirements Specification....................................................................................... pg 9

4.1. Functional Requirements............................................................................... pg 9

4.2. External Interface Requirements................................................................... pg 10

4.3. Logical Database Requirements.................................................................... pg 11

4.4. Design Constraints......................................................................................... pg 11

5. Other Nonfunctional Requirements........................................................................... pg 12

5.1. Performance Requirements............................................................................ pg 12

5.2. Safety Requirements...................................................................................... pg 12

5.3. Security Requirements................................................................................... pg 12

5.4. Software Quality Attributes........................................................................... pg 12

5.5. Business Rules............................................................................................... pg 12

6. Legal and Ethical Considerations.….......................................................................... pg 13

Appendix A: Glossary........................................................................................................ pg 14

Appendix B: Analysis Models........................................................................................... pg 15

Appendix C: To Be Determined List................................................................................. pg 16

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Date | Reason For Changes | Version |
| Documentation Subteam | 11/6/2020 | Initial creation/start of documentation process | 1.0 |
| Pierce Wei | 12/5/2020 | Finalization of documentation | 1.1 |
|  |  |  |  |
|  |  |  |  |

**1. Introduction**

**1.1 Purpose**

The purpose of this document is intended to give an overview on the project’s functionalities and nonfunctional requirements. It will go over aspects of the first version of the software and how it was designed.

**1.2 Intended Audience and Reading Suggestions**

This SRS is intended for the Office of Public Defender LA County officials, project managers, employees, contractors, and vendors. The Office of Public Defender LA County staff will also be maintaining the software.

**1.3 Product Scope**

The software is designed to allow employees, contractors, and vendors to fill out personal information that will be passed and shared with six forms in the back-end section. This will allow the users to select a specific request form to fill out with all their information prepopulated already. However, the user is required to enter all their information prior to having all the forms pre populated so that the system will have their information ready. Once the software is released, it should provide an efficient method for contractors and employees to request for any forms they would like without having to repeatedly fill in their information fields.

**1.4 Definitions, Acronyms, and Abbreviations**

AngularJS - JavaScript-based open-source front-end web framework used in developing single-page applications

ISD - Internal Services Department

PDF - Portable Document Format

SRS - Software Requirements Specification

BOX - Secure Content Management Software

Adobe API - Adobe Application Programming Interface

**1.5 References**

All PDF files will be directly linked within the front-end portion of the project when completed. Each PDF will also have attached the Board of Supervisors Policy No. 6.101 (Revised January 2019) at the end of each form.

* County of Los Angeles Internal Services Department: Active Directory/Hosted Registration Form For L.A. County Employees
  + Revised October 2017
* County of Los Angeles Internal Services Department: Active Directory/Hosted Registration Form For Contractor/Vendor
  + Revised October 2017
* County of Los Angeles Downey Data Center Registration For L.A. County Employees
  + Revised October 2015
* County of Los Angeles Downey Data Center Registration For Contractors/Vendors
  + Revised October 2015
* County of Los Angeles Internal Services Department Internet Registration Form For L.A. County Employees
  + Revised November 2011
* County of Los Angeles Downey Data Center Registration For L.A. Contractor/Vendor
  + Revised November 2011
* Employee/Contractor Enrollment Application

**2. Overall Description**

This section will provide background information about the specific requirements of the application service to be developed in brief. Everything will be explained in broad detail for simplified understanding for the customer/client’s benefit.

**2.1 System Analysis**

The goal of the project is to create an efficient approach for employees and contractors to request a specific form with their information prepopulated in it already. In order to implement this system, we establish a front-end page where users can enter the specific form they request, which will send the data to a back-end database which retrieves their information. For this to happen, we have to create an efficient front-end that can be able to communicate and manipulate data in the back-end.

**2.2 Product Perspective**

This application applies the usage of BOX and Adobe API for users to fill out and sign forms that will be saved in a database. Since BOX is used for the request and storing of information, Adobe API will be how users are able to fill out forms. A structural example of this design is when users are able to fill out forms and it has their information prepopulated.

**2.3 Product Functions**

The functions of the product are:

1. Submit user request form from Angular app
2. Request details are retrieved from back-end and details are added to a database
3. Request details are sent to BOX to generate folders
4. Adobe API generates PDF forms for users to sign
5. User signs forms then submits it via BOX and stored in BOX

**2.4 User Classes and Characteristics**

The people that would be designated to use this product would generally be new employees or contractors/vendors. There are two different forms based on whether the user is an employee or contractor. If the user is an employee per se, their given form(s) would be designed only for an employee to fill out whereas if the user were a contractor, their given form(s) would have all the information needed for a contractor.

**2.5 Operating Environment**

The main operating environment should be a web application that the software will be held on. Any sort of browser and operating system with minimum processing power should work. The other applications working with the app include BOX and Adobe API.

**2.6 Design and Implementation Constraints**

* A server is needed in order to host the application; early development was hosted on developer’s computer
* Trying to understand how user’s information can be shared across all six PDF forms
* Needs to have a secure method of storing user’s information without others being able to access it

**2.7 User Documentation**

User manual will be created, completed, and provided upon completion of the project.

**2.8 Assumptions and Dependencies**

The application will be on an Angular application that should take the request from the user and send the data back to an employee folder in BOX. BOX will then submit the data to a database and it will send data back to BOX for the user to fill out. Once the user fills out the form(s), the data will be sent back via BOX and into the database for record keeping.

**2.9 Apportioning of Requirements**

Ability to recognize multiple languages, mobile application, website.

**3. External Interface Requirements**

**3.1 User Interfaces**

The interface that will practically be used is a website which has Custom Forms attached to it. Users are allowed to choose the form that they would like to fill out. Upon pressing a form, the user will be redirected to that specific Custom Form and will be able to input data into different data fields, which range from text, check boxes, and multiple choice. Upon pressing “submit” for the form, the user will be able to review the information and also be able to send to others before completing the form.

**3.2 Hardware Interfaces**

N/A

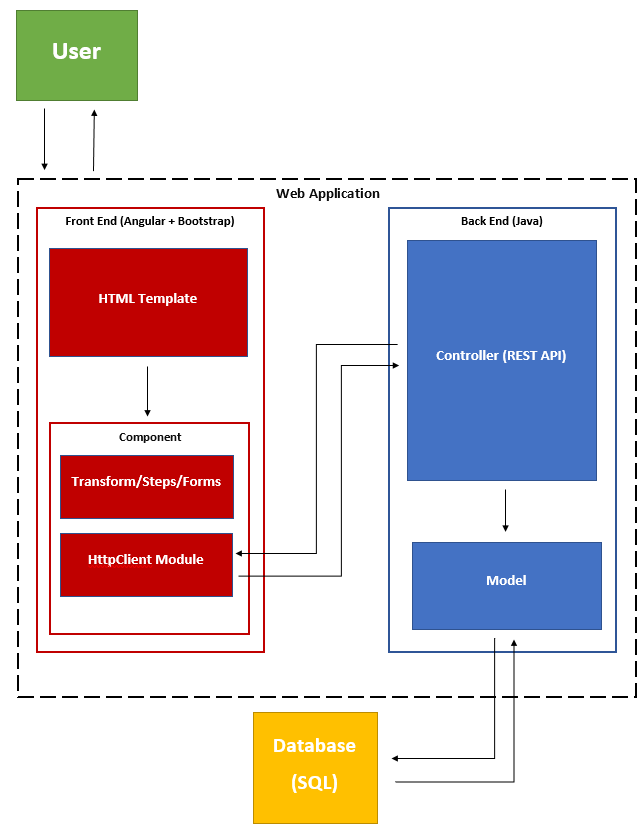
**3.3 Software Interfaces**

N/A

**3.4 Communications Interfaces**

Electronic Forms that will be saved by the user and officially submitted will make use of BOX, which in turn requires access to the account in order to obtain and go over the file once again. Viewing all of the forms that have been submitted requires admin permission.

**4. Requirements Specification**



**4.1 Functional Requirements**

* The System will be creating a Custom Form using Angularjs to capture employee/contractor form specific information
* Employee Form Key Fields
  + First Name
  + Last Name
  + Telephone
  + Address(Street, City, State, Zip Code)
  + Email ID
  + Employee Type (Defaults to Employee)
  + Employee ID
  + Submission Date (Default to Current Date)
  + Requestor Name
  + Requestor Email
  + Approvers
  + Department Number
  + Department Name
* Contractor Form Key Fields
  + First Name
  + Last Name
  + Contracting Company (Only available if Employee Type is set to Contractor)
  + Telephone
  + Address(Street, City, State, Zip Code)
  + Company Email ID
  + Employee Type (Defaults to Contractor)
  + Employee ID (Only available if Employee Type is set to Employee)
  + Submission Date (Default to Current Date)
  + Requestor Name
  + Requestor Email
  + Approvers
  + Department Number
  + Contract Work Order Number
  + Contract Expiration Date
  + Business Street Address
  + Department Name
* There should be a Request Status page within the AngularApp; the following fields will be searchable from the search page
  + Request ID
  + Employee ID
  + Contracting Company
  + First Name
  + Last Name
* Upon submission of a Form (Request is recorded and an entry is added in the database and status is sent to "Initial Request Sent" and “RequestID” is presented back to the requestor)

**4.2 External Interface Requirements**

As of the moment, the only input necessary would be the User’s input. Specific details regarding the amount that can be put within each field is still currently in the works.

**4.3 Logical Database Requirements**

N/A

**4.4 Design Constraints**

Can only be opened in a web browser on a computer, mobile browsers most likely will not work. There also needs to be a web server for the application to be run on in order for the application to full function. If there is no server to hold the application, it will not start.

**5. Other Nonfunctional Requirements**

**5.1 Performance Requirements**

* Software should reduce the amount of time spent completing forms.
* User interface will be built using AngularJS
* Forms will be created using PDFBox
* Forms will be edited using PDFBox

**5.2 Safety Requirements**

N/A - no loss, damage, or harm should come from improving efficiency of form filling.

**5.3 Security Requirements**

* Host server shall protect the integrity and privacy of data collected from Employee/Vendors.

**5.4 Software Quality Attributes**

* Web App shall adapt to multiple devices.
* Data stored on Employee/Vendors shall easily be accessed for reference/updates.
* Software should flow easily for users to use.
* Software will require little to no maintenance.

**5.5 Business Rules**

* New Employee/Vendors may only view the front-end of the application to submit field responses.
* Employers will be able to access employee submitted fields as well as documents required for respective Employee/Vendor.
* Office of Public Defenders staff have access to the databases

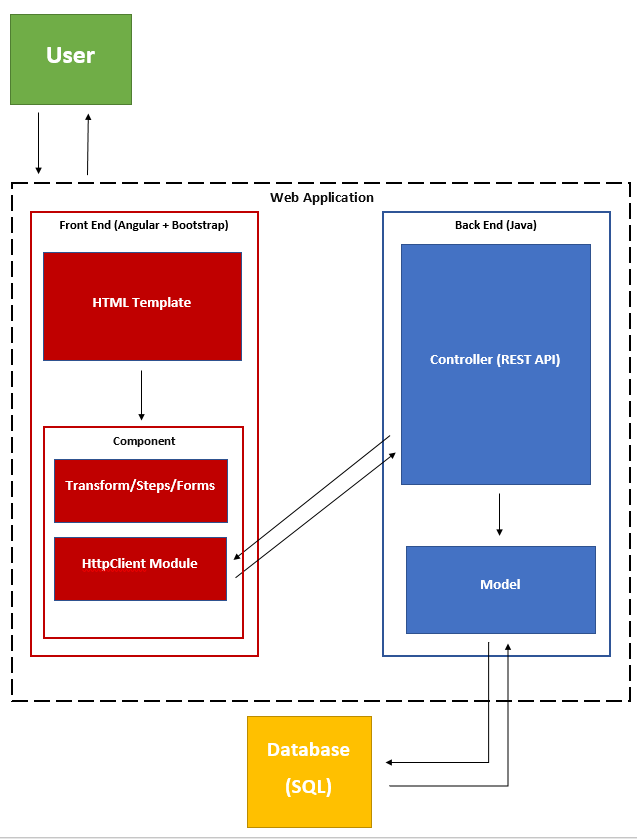
**6. Legal and Ethical Considerations**

No legal issues will come from the application as it will be used solely to ease the creation of employee/vendor documents.

**Appendix A: Glossary**

Refer to section 1.4

**Appendix B: Analysis Models**



**Appendix C: To Be Determined List**