

Software Requirements Specification

Cloud Database Migration & LACPD New Hire Enrollment Web Application

Version 1.3 approved

**Prepared by Simon Mai, Albert Chen, Shahram Mehri Kalantari, Wilfredo Paz,
Michael Loria, Ryan Lee, Ismael Valenzuela, Joshua Perez, Norberto Gomez
Rosales, Fabio Quintana**

Law Offices of Los Angeles County Public Defender & Cal State LA

13 May 2022

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 5
1.4. Definitions, Acronyms, and Abbreviations	pg 5
1.5. References.....	pg 6
2. Overall Description.....	pg 6
2.1. System Analysis.....	pg 6
2.2. Product Perspective.....	pg 7
2.3. Product Functions.....	pg 7
2.4. User Classes and Characteristics.....	pg 8
2.5. Operating Environment.....	pg 8
2.6. Design and Implementation Constraints.....	pg 9
2.7. User Documentation.....	pg 9
2.8. Assumptions and Dependencies.....	pg 10
2.9. Apportioning of Requirements.....	pg 10
3. External Interface Requirements.....	pg 11
3.1. User Interfaces.....	pg 11
3.2. Hardware Interfaces.....	pg 11
3.3. Software Interfaces.....	pg 11
3.4. Communications Interfaces.....	pg 12
4. Requirements Specification.....	pg 13
4.1. Functional Requirements.....	pg 13
4.2. External Interface Requirements.....	pg 14
4.3. Logical Database Requirements.....	pg 15
4.4. Design Constraints.....	pg 15
5. Other Nonfunctional Requirements.....	pg 16
5.1. Performance Requirements.....	pg 16
5.2. Safety Requirements.....	pg 16
5.3. Security Requirements.....	pg 16
5.4. Software Quality Attributes.....	pg 16
5.5. Business Rules.....	pg 17
6. Legal and Ethical Considerations.....	pg 18
Appendix A: Glossary.....	pg 19
Appendix B: Analysis Models.....	pg 20
Appendix C: To Be Determined List.....	pg 21

Revision History

Name	Date	Reason For Changes	Version
Simon Mai Ryan Lee	11/15/2021 11/30/2021	Initial Creation	1.0
Simon Mai	4/18/2022	Updating Requirements and Plans	1.1
Norberto Gomez Rosales	4/20/2022	Revision and Grammar editing	1.2
Wilfredo Paz	4/23/2022	Updating	1.3

<Add rows as necessary when the document is revised. This document should be consistently updated and maintained throughout your project. If ANY requirements are changed, added, removed, etc., immediately revise your document.>

1. Introduction

The Introduction section of the Software Requirements Specifications (SRS) document should give an overview of the entire document. As you write this document, remember that the purpose of this document is to tell software engineers WHAT the software is supposed to do. The details on HOW the software will be implemented should be specified in the Software Design Document (SDD).

1.1 Purpose

This document aims to identify the requirements for the two PowerApps and the Enrollment App. Identify the needed software and how it is to be modified to meet the Public Defender LA County officials' requirements. This document will cover all aspects of the software.

Identify the purpose of this document and the product whose software requirements are specified in this document. Be sure to mention the version / revision / release number of the software and also mention if the document covers all aspects of the software or only a small part of the software. Do not iterate over the table of contents here. Give a general overview of what this document contains.

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 in the future.

This section should mention what types of readers the document is intended for such as developers, project managers, marketing staff, users, testers, and documentation writers.

Describe what this SRS contains and how it is organized. If each type of reader should interpret this document differently, give a suggested reading sequence for the document that is relevant to that particular reader type. Example: A tester might only be interested in the detailed requirements list, or the marketing staff may only be interested in the very high level descriptions of the project.

1.3 Product Scope

The Power Apps are split into two applications, one is PD-GO, and the other is Help Desk. Both apps must be migrated, and after the migration, the implementation of the azure databases.

The Enrollment App should allow Admin or HR to fill employee/contractor information that will fill necessary fields and create a flow of checks and signatures in a specific order into one form in the back-end section. When completed, the software should allow for HR/Admin to efficiently onboard new employees and contractors with the ability to get status updates and without public access to the software.

In this section:

- Identify the software product(s) to be produced by name.
- Explain what the software will do. If necessary mention what the software will not do.
- Describe how the software will be used once released, include benefits of the software, objectives, goals, etc.

This section should give a high-level summary of the software. Do not list all of the requirements here.

1.4 Definitions, Acronyms, and Abbreviations

Azure - A platform as a service (PaaS) database engine that manages most of the management functions such as upgrading, backups, patching, and monitoring without user involvement built for the cloud.

SRS - Software Requirements Specification

Adobe API - Adobe Application Programming Interface

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

Adobe Sign - Adobe Sign is a cloud-based e-signature service that allows the user to send, sign, track, and manage signature processes using a browser or mobile device. It is part of the Adobe Document Cloud suite of services

BOX - Box is a highly secure cloud-based content management, collaborative platform, and file sharing for businesses.

Provide definitions for any relevant terms, acronyms, and abbreviations that are necessary to understand the SRS document. This information may be listed here, in an appendix at the end of the document, or in a completely separate document. If the information is not directly listed in this section provide a note that specifies where the information can be found.

1.5 References

All PDF files to be Assigned by HR/Admin and sent to a new employee or contractor and signers. The files will be directly on Adobe Sign when completed.

- Blank_Active_Directory_ISD_Registration_Form.pdf

- Revised April 2021
- Blank_MainFrame_ISD_Registration_Form.pdf

- Revised April 2021

This section should provide the following:

- A complete list of all documents referenced in the SRS.
- A complete list of any Web addresses referenced in the SRS.
- For each reference mention the title, author, version number, date, and the source or location of the reference.
- Generally you should provide enough information for each reference so the reader of the SRS can easily obtain copies of these references.

References may include items such as: user interface style guides, contracts, standards, other SRS documents, use case documents, supplemental documents that were researched while writing this SRS or while designing the software.

2. Overall Description

The document will provide an overview of information about the specific requirements of the enrollment application. The goal is to have the integrity of the process simplified and understood for the customer/client's benefit.

This section explains the general facts that affect the software and its requirements. You should not state any specific requirements in this section. Here you will provide a background for the actual list of requirements that will be defined in section 4. The point in this section, is to give a high level description of the requirements in plain English for the benefit of the customer / client.

2.1 System Analysis

Power Apps

The goal of this project is to migrate the old database that is being used to the new database using the azure database. Migrating will allow for better scalability and is easier to use.

Enrollment App

The goal of this project is to create an onboarding application where HR/Admin can fill in information for the new employee or contractor. Then send the form out to the recipient to fill the rest of the information followed by signatures from necessary management while allowing HR personnel to

check for form filling correctness and status. This removes the ability for the public to access the software, unlike the last enrollment application design.

This section should describe the overall problem the software is design to solve, and the general approach identified or selected to solve it. In particular, this section should:

- a Define and articulate the goals of the project.
- b Present an analysis which identifies all major technical hurdles associated with carrying the project through to completion.
- c Identify the solutions to overcome the technical hurdles.

2.2 Product Perspective

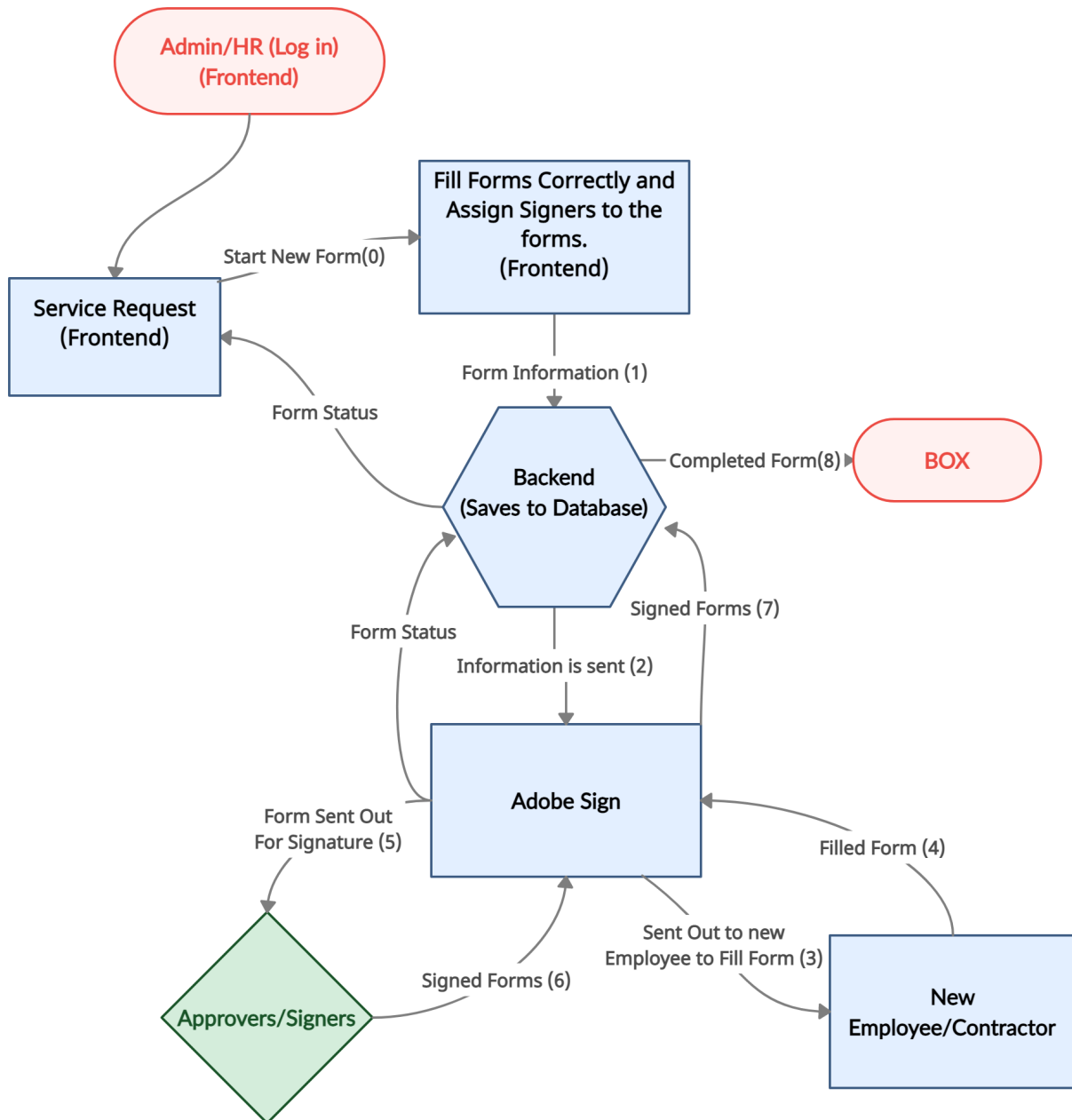
Power Apps -

These applications applies the use of Azure SQL Database. Azure SQL Database allows for more flexibility with high speed connectivity.

Enrollment App -

This application will use Adobe API so HR can fill out information HR has about the new employee or contractor before sending it out to the new employee or contractor to complete the form and check if the employee or contractor filled the documents incorrectly.

Adobe API will act as our received and go software when interacting with our application. We as a group aren't familiar with other applications systems that function like this, but that doesn't mean we are the first. Overall, the task was to integrate Adobe API to work together with our application, and the block diagram will give a better visual representation of how it works.



This section should show how the software may relate to other products. If the software is completely independent of any other products / systems, you should also mention that here.

If the SRS describes software that is a component of a larger system, this section should describe how the requirements of this software relate to the larger system. Be sure to mention and identify interfaces between the larger system and the software that this SRS defines.

This section should also mention how the software in this SRS compares to similar systems currently on the market (if any exist). Mention similarities / differences / motivations for creating this software when

compared to an existing product. If the software is a completely new innovation, be sure to mention that as well.

You may provide a block diagram here showing the major components of the larger system, external interfaces, and how your software fits in with the larger system. This is NOT a design or architecture picture. It is simply here to provide a context for the reader to understand where your software fits with the larger system. Your software should be shown as a black box. Remember, it is the purpose of the Software Design Document to present the internal workings of your product.

2.3 Product Functions

The functions of the PD-GO are

1. Make your request.
2. Once the user completes writing their request, the requests will then be added into the database.
3. Managers are notified via email about the request and decide to approve or decline the request.
4. Users will then be shown whether their request was approved or declined.

The functions of the Enrollment App are

1. Login or create a new account with email and password
2. An angular-focused app designed to create custom forms to obtain employee/contractor information
3. Outlook of 'Status Page' within the application that gives you options to continue as an employee, contractor, or input a form # of an existing form.
4. Inputs are recorded and sent as an entry to the database.
5. Inputs are collected into one form to be submitted.
6. Employees will receive an email from Adobe API to fill out and sign forms.
7. Form is sent to a 'requestor' for a review then, sends it to others for additional inspection.

This section provides a summary of the major functions that the software will perform. The functions should be organized in a way that makes them understandable to the customer / client or anyone who is reading this document.

You can provide a bullet list summary of each function and / or provide simple diagrams to show the different functions and their relationships. Again, these diagrams do not show the design of the product, just the relationships between the functions.

Again, describe what the functions of the system need to do, not how you are going to implement them.

2.4 User Classes and Characteristics

Power Apps

The people who would be designated to use this product would be all of the Law Offices of Los Angeles County Public Defender employees with a Public Defender email. One user would be a requester to request either a personal leave or team leave and the other user will be the manager who will either approve or decline the request.

Enrollment App

The people that would be designated to use this product would be HR and Admin. An HR employee or an Admin would fill out basic information about the new employee or contractor then the form would be sent out according to the information provided by HR or Admin.

Identify the various user classes that you anticipate will use this product. User classes may be differentiated based on frequency of use, subset of product functions used, technical expertise, security or privilege levels, educational level, or experience. Describe the pertinent characteristics of each user class. Certain requirements may pertain only to certain user classes. Distinguish the most important user classes for this product from those who are less important.

2.5 Operating Environment

Power Apps

The main operating environment should be on a mobile app or on Power Apps. iOS 14 or later is but recommended iOS 15 or later, Android 9 or later but is recommended Android 12 or later, and Windows 8.1 or later but is recommended Windows 10 (Fall Creators Update with at least 8GB of RAM).

Enrollment App -

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, Spring Boot server, Adobe API, and Adobe Sign Platform.

Provide details on the environment in which the software will live. This should include the hardware platform, operating system, and any other software components or applications that will affect your software and which have to coexist with your software.

2.6 Design and Implementation Constraints

Power Apps - An azure database will be needed to store all the data that will be inputted by the user and to also store all information of the user.

Must have a Public Defender email to be able to access the Power Apps PD-GO and Help Desk.

Enrollment App -

1. A server is needed in order to host the application
2. Unable to Access Adobe Sign API / tokens
3. Need to manually create Adobe Sign access token everytime token expires

4. Adobe Sign is tedious because we have to make multiple accounts('free trial) to access a feature that will help us develop our application.
5. Overall, progress on the project was dragged over time because of a lack of communication with our sponsors.

Give a general description of any items that will influence the ability of the software developers to implement the product. These can include things such as:

Regulatory policies within the organization that may affect development the software.

Hardware limitations

Interfaces with other applications

Parallel operation

Higher-order language requirements

Reliability requirements

Safety and security considerations.

Memory Constraints

Be sure to list any constraints that will affect the software. Generally this is to give an overview of any non-functional requirements that will be detailed in a later section.

2.7 User Documentation

Enrollment App

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

List any user documentation (user manuals, online guides, tutorials, etc) that will be delivered along with the software. Specify in what format each document will given.

2.8 Assumptions and Dependencies

Enrollment App -

The application will be on an Angular application which takes service request data from the user and send the data to the Spring Boot back-end application. Data sent to the back-end will be stored in a database and may be sent back to the front-end for reviewing. Once the user gets to the step of signing the document, the document will not be considered complete until all other specified users electronically sign the document.

List any other factors that may affect the requirements stated in this document. These are not design constraints (as mentioned above) but are factors that if changed will affect the requirements.

For example, you might assume that a specific operating system is available on the hardware allocated for this product. If this operating system were not available, then the SRS would have to be changed accordingly.

List any third-party or commercial components that you plan to use. List any software component dependencies that you might want to reuse from another project.

This section is a catch-all for anything that will affect your software design that did not fit into any of the previous sections. If anything will affect your requirements, mention it here.

2.9 Apportioning of Requirements

Enrollment App

Minor bugs that arose throughout the process would delay further versions of the system but possible fixes to the system is possible given the time.

List any requirements that might be delayed until future versions of the system.

3. External Interface Requirements

Again, this section is a high level description for the non-technical people who may be reading this document. Everything in this section is to detail how your software interacts with any external interfaces, whether these are other software or even hardware interfaces.

3.1 User Interfaces

Enrollment App -

The interface that will be used is a website where the user can choose the form to be sent. Upon selecting a form, the admin will have the ability to input data in data fields given the information known about the employer, like text fields, radio buttons, and drop-down lists. Upon pressing “save,” the user will be able to review the information and submit it to the form filler based on the information provided

Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed. Details of the user interface design should be documented in a separate user interface specification.

Be sure to mention any requirements that must conform to the Americans with Disabilities Act.

3.2 Hardware Interfaces

Enrollment App - N/A

Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware, and communication protocols to be used.

This is not a description of hardware requirements such as "This program will run on a PC with 16 gigabytes of RAM." This section details the actual hardware devices that your application will interact with. Items such as robotics components, lighting, audio / video equipment, etc. If your software does not have an hardware interface requirements, then state this instead.

3.3 Software Interfaces

Enrollment App -

Forms that will be submitted and saved will use the Adobe Sign API. This requires the access to the Adobe account to obtain the files. To view all the documents that have been completed, it is required to have Adobe account permission.

List any other software products and interfaces that your requirements must utilize. For each required product, list the name, version number, and source.

Document any APIs that your software will have to access in order to interact with other software products. Also document how your software will communicate / pass information to the external software.

Example, your customer uses MySQL version 1 and you are required to use that in your design, then you must specify this here.

This section is NOT for specifying software that you think would be good to use. This is for customer-specific requirements that you HAVE to interact with.

3.4 Communications Interfaces

Enrollment App -

We will be using the front end angular website and adobe sign to communicate with the employee. The front end will display any instructions while the adobe sign will display and allow the employee to sign the document. The front end and adobe sign will also be used to allow the department to accept or deny the signed document. These documents will then be put into Box which will allow the public defender's office to have access to them.

4. Requirements Specification

This section contains all of the necessary software requirements with enough detail to allow designers to accurately design the software to satisfy those requirements, and to allow testers of the software to verify that all requirements have been satisfied. The requirements should include a description of every input to the system, every output, and all functions performed by the system in response to an input or output.

The biggest thing to remember is that this section is for the software developers (technical people) while the previous sections were for the customers / non-technical people.

Also remember that this is not HOW things will be implemented, but WHAT will be implemented.

Requirements should be written according to the following:

- Specific requirements should be correct, unambiguous, complete, consistent, ranked for importance and / or stability, verifiable, modifiable, and traceable.
- Specific requirements should be cross-referenced to earlier documents that are relevant.
- All requirements should be uniquely identifiable using a consistent numbering system, i.e. 1.1, 1.2, 1.1.2, and so on.
- Requirements should be organized in a logical manner to provide the most readability.

Use the following format for each requirement:

The system shall... (this means this requirement is mandatory).

The system should... (this means a desired feature, but may be delayed until later).

This system may... (A optional, nice-to-have feature that might not be implemented).

Remember to number each requirement for traceability. Use a system such as 1.1, 1.1.1, 1.1.2.1, and so on. Each requirement need to be testable. Avoid statements that are general and vague such as "The system shall be easy to use." or "The system shall be developed using good software engineering practices."

Do not include examples. Remember that this is a specification and the designer should be able to read this and build the system without having to bother the customer again. Every minute detail must be documented here.

EVERYTHING in section 4 must be written following the above guidelines.

4.1 Functional Requirements

Enrollment App -

- Employee/Contractor Form Key Fields
 - Employee Type (Employee or Contractor)
 - First Name
 - Middle Initial
 - Last Name
 - Email Address
 - County Department/Organization Name

- County Department Number
- Cell Phone Number
- Work Phone Number
- Employee Number
- Contractor Company Name
- Contract Work Order Number
- Contract Expiration Date

Functional requirements define the fundamental actions that must take place in the software in accepting and processing the inputs and in processing and generating the outputs. These are generally listed as “shall” statements starting with "The system shall..."

These include:

- Validity checks on the inputs
- Exact sequence of operations
- Responses to abnormal situation, including
 - Overflow
 - Communication facilities
 - Error handling and recovery
- Effect of parameters
- Relationship of outputs to inputs, including
 - Input/Output sequences
 - Formulas for input to output conversion

It may be appropriate to partition the functional requirements into sub-functions or sub-processes. This does not imply that the software design will also be partitioned that way.

This section should be as detailed as possible, again, listing WHAT your software is going to do, not HOW you are going to accomplish it.

4.2 External Interface Requirements

Enrollment App -

- Name: Automation for document sendoff
- Description: Will send the document for others to approve the signed document. The document will also be saved to Box
- Source: Adobe sign signed document
- Data format: PDFs saved to Box.

The input will come from the user, and examples of the inputs that the user will put in the fields are mentioned in 4.1 Functional Requirments. The approvers will only need to sign the documents and edit their fields if needed.

This contains a detailed description of all inputs into and outputs from the software system. It complements the interface descriptions in section 3 but does not repeat information there. Remember section 3 presents information oriented to the customer/user while section 4 is oriented to the developer.

It contains both content and format as follows:

- Name of item
- Description of purpose
- Source of input or destination of output
- Valid range, accuracy and/or tolerance
- Units of measure
- Timing
- Relationships to other inputs/outputs
- Screen formats/organization
- Window formats/organization
- Data formats
- Command formats
- End messages

4.3 Logical Database Requirements

N/A

4.4 Design Constraints

There are no design constraints because everything will be run on adobe sign which is cloud-based. It can run on any device that contains a web browser. It will only need a server to host the application; if there is no server, it will not run.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

Enrollment App -

- Software should reduce the time to complete multiple forms
 - Software will remove the need to manually forward documents
- This section specifies any numerical / statistical requirements imposed on the software such as:
- The number of terminals to be supported
 - The number of simultaneous users to be supported
 - Amount and type of information to be handled

Dynamic numerical requirements may include, for example, the numbers of transactions and tasks and the amount of data to be processed within certain time periods for both normal and peak workload conditions.

All of these requirements should be stated in measurable terms. For example, "95% of the transactions shall be processed in less than 1 second" rather than, "An operator shall not have to wait for the transaction to complete."

(Note: Numerical limits applied to one specific functional requirement are normally specified as part of that requirement and should be listed in section 4. This part is more for performance / statistical requirements imposed on the entire system as a whole.)

5.2 Safety Requirements

Enrollment App - There are no safety requirements needed due to the software being used to submit forms efficiently.

Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product's design or use. Define any safety certifications that must be satisfied.

5.3 Security Requirements

Enrollment App - The host server shall protect the data collected from employees/ contractors.

Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements. Refer to any external policies or regulations containing security issues that affect the product. Define any security or privacy certifications that must be satisfied.

5.4 Software Quality Attributes

Enrollment App -

- The webform should be usable on multiple devices, all browser platforms, and all operating systems.
- Employees/Contractor's data will be stored and be easily accessed for updates and references

Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, re-usability, robustness, test-ability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.

5.5 Business Rules

- Employees will have access to the front end to sign two forms using adobe sign
- Contractor and appointed signees will have the ability to accept the employees form
- Public Defender's office employees will have access to signed forms from Box

6. Legal and Ethical Considerations

A legal and ethical consideration for our project is keeping the forms and the employee's information confidential.

Appendix A: Glossary

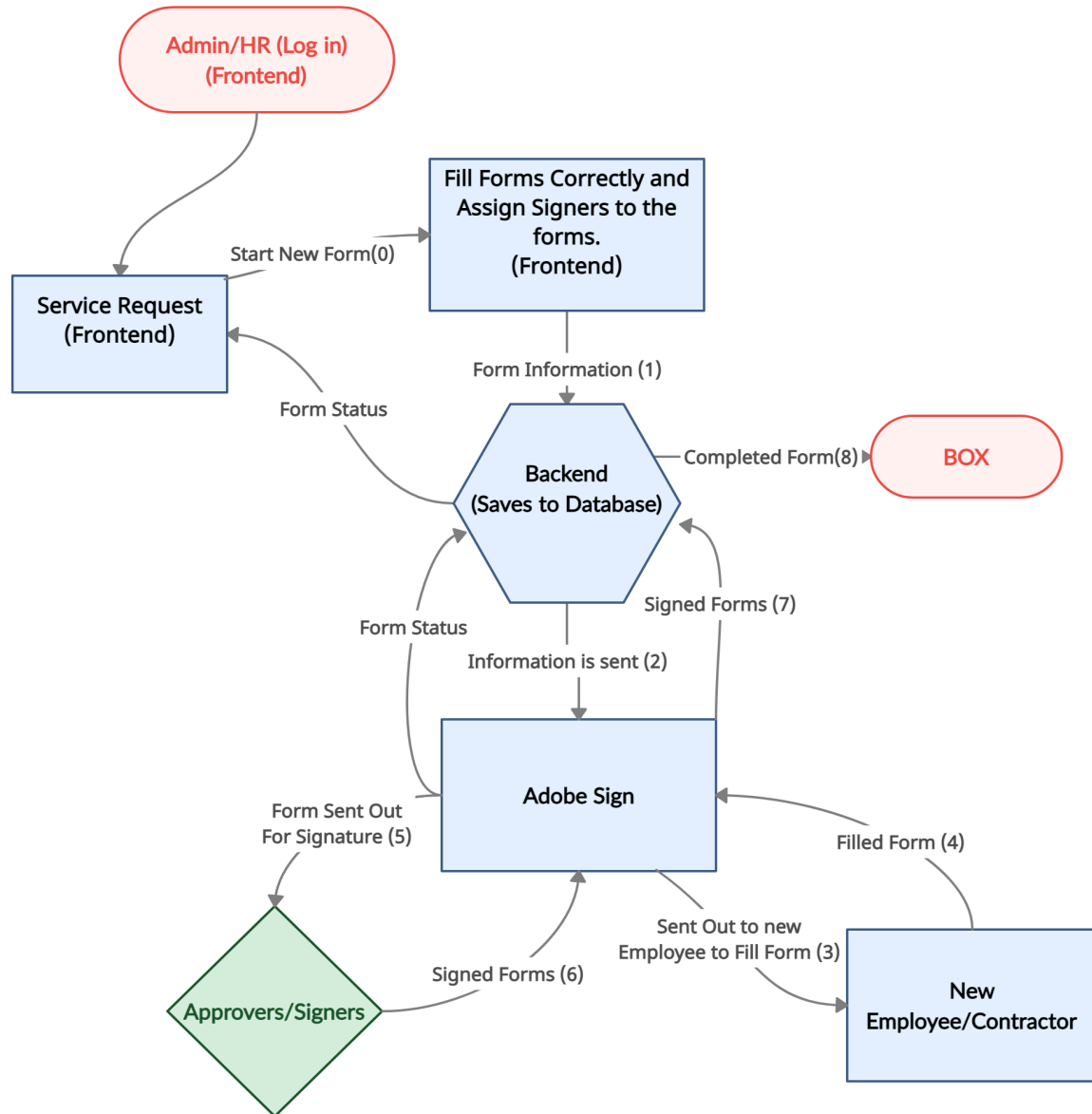
Refer to section 1.4.

Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.

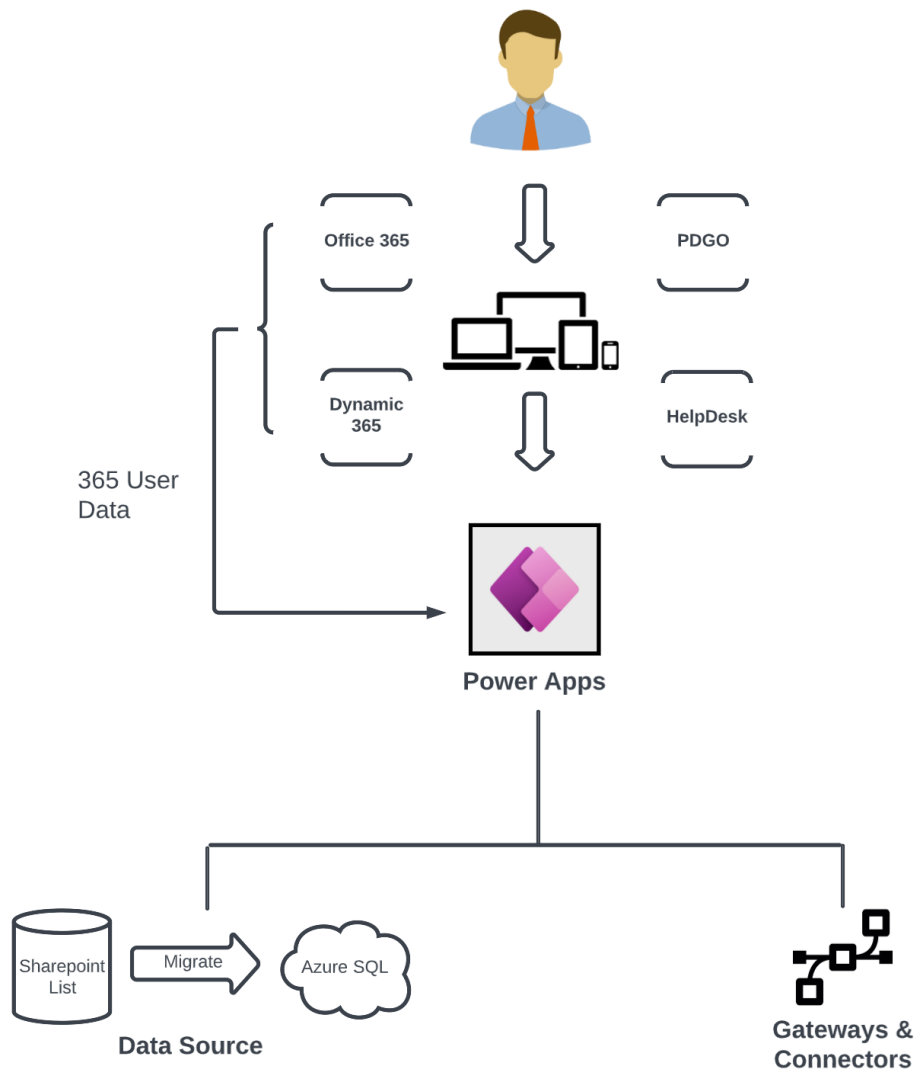
If this section is very short you may include it in section 1.4. If your list is very long you may include it here and put a reference to this Appendix in section 1.4.

Appendix B: Analysis Models

Enrollment Application



Power Apps



Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.

Appendix C: To Be Determined List

Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.