|  |
| --- |
| Enterprise Tracking Application |
| Software Requirements Specification |
| Prepared For: Perspecta  Location: Austin, TX |
| Date: 9/7/2020  Revised: 9/18/2020 |
| Angeleen Abesamis  Remnin Ferrer  James Fritz  Yeremy Joseph  Eric Leon |

# Abstract

This document serves as the software requirements specification for the Enterprise Tracking Application that is being developed by the University of Texas at Dallas project team for Perspecta Inc. We will use this document to define what the application will do.

Table of Contents

[Abstract 2](#_Toc52431889)

[List of Figures 4](#_Toc52431890)

[List of Tables 5](#_Toc52431891)

[1. Introduction 6](#_Toc52431892)

[1.1 SRS Introduction 6](#_Toc52431893)

[1.2 SRS Purpose 6](#_Toc52431894)

[1.3 SRS Scope 6](#_Toc52431895)

[1.4 SRS Structure 7](#_Toc52431896)

[2. Use Case Model for Functional Requirements 8](#_Toc52431897)

[2.1 Graphical User Interface Use Case Models 8](#_Toc52431898)

[2.2 Authentication Layer Use Case Models 16](#_Toc52431899)

[2.3 Web Services Use Case Models 17](#_Toc52431900)

[2.4 Persistence Layer Use Case Models 27](#_Toc52431901)

[3. Use Case Model Rationale 28](#_Toc52431902)

[4. Non-Functional Requirements 29](#_Toc52431903)

[4.1 Usability 29](#_Toc52431904)

[4.2 Performance 29](#_Toc52431905)

[4.3 Portability 29](#_Toc52431906)

[4.4 Security 29](#_Toc52431907)

[5. Evidence of Configuration Management 30](#_Toc52431908)

[6. References 31](#_Toc52431909)

# List of Figures

|  |  |  |
| --- | --- | --- |
| Figure | Description | Page |
| 2-1 | Use case 1. | 8 |
| 2-2 | Login page wireframe. | 9 |
| 2-3 | Use case 2. | 10 |
| 2-4 | Use case 3. | 11 |
| 2-5 | Use case 4. | 12 |
| 2-6 | Use case 5. | 13 |
| 2-7 | Home page wireframe. | 14 |
| 2-8 | Authentication Layer | 16 |
| 2-9 | Use case 7. | 17 |
| 2-10 | Use case 8. | 18 |
| 2-11 | Use case 9. | 19 |
| 2-12 | Use case 10. | 20 |
| 2-13 | Use case 11. | 21 |
| 2-14 | Use case 12. | 22 |
| 2-15 | Use case 13. | 23 |
| 2-16 | Use case 14. | 24 |
| 2-17 | Use case 15. | 25 |
| 2-18 | Use case 16. | 27 |

# List of Tables

|  |  |  |
| --- | --- | --- |
| Table | Description | Page |
| 2-1 | Use case 1. | 8 |
| 2-2 | Use case 2. | 10 |
| 2-3 | Use case 3. | 11 |
| 2-4 | Use case 4. | 12 |
| 2-5 | Use case 5. | 13 |
| 2-6 | UI Functional Requirements. | 15 |
| 2-7 | Use case 6. | 16 |
| 2-8 | Use case 7. | 17 |
| 2-9 | Use case 8. | 18 |
| 2-10 | Use case 9. | 19 |
| 2-11 | Use case 10. | 20 |
| 2-12 | Use case 11. | 21 |
| 2-13 | Use case 12. | 22 |
| 2-14 | Use case 13. | 23 |
| 2-15 | Use case 14. | 24 |
| 2-16 | Use case 15. | 25 |
| 2-17 | Web Services Layer Functional Requirements. | 26 |
| 2-18 | Persistence Layer Functional Requirements. | 27 |
| 3-1 | Use case 6 rationale. | 28 |

# 1. Introduction

### 1.1 SRS Introduction

The introduction of the Software Requirements Specification (SRS) provides an overview of the entire SRS with purpose, scope, definitions, acronyms, abbreviations, references and overview of the SRS. The aim of this document is to gather and analyze and give an in-depth insight of the complete Enterprise Tracking Application software by defining the problem statement in detail. Nevertheless, it also concentrates on the capabilities required by stakeholders and their needs while defining high-level product features. The detailed requirements by Perspecta are provided in this document.

#### 1.1.1 Definitions, Acronyms, and Abbreviations

* **Auth** – Authentication.
* **FR** – Functional Requirement
* **PL** – Persistence Layer
* **UC** – Use Case
* **UI** – User Interface
* **WS** – Web Services Layer

#### 1.1.2 References

IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

### 1.2 SRS Purpose

The purpose of the Software Requirements Specification is to provide a detailed overview of the Enterprise Tracking Application software product, its parameters and goals as per the requirements set by Perspecta Inc. This document describes the project's user interface, hardware and software requirements. It defines how Perspecta, and the team see the product and its functionality. In summary, the SRS will help the design and development throughout the software delivery lifecycle (SDLC) processes.

### 1.3 SRS Scope

Under the direction of UT Dallas faculty, the UT Dallas Senior Project Team is tasked to generate and deliver the Enterprise Tracking Application product for Perspecta. The product shall manage integration records throughout the company which will allow project integration tracking capabilities for the Perspecta Enterprise. The product also provides emphasis on permission-based access, ensuring that only users with appropriate credentials can view all records. The four main components of the software are a graphical user interface that displays integration records and allows the user input, an authentication layer that validates user credentials for permissions, a web service layer that allows communication between users and the server through REST API methods, and an MSSQL persistence layer that stores integration record data.

### 1.4 SRS Structure

The SRS is structured as follows:

* Section 1, Introduction. The introduction of the Software Requirements Specification provides an overview of the entire document. It includes the introduction, purpose, scope, and structure of the document.
* Section 2, Use Case Model for Functional Requirements. This section is broken into the major components of the project. It contains the use cases for the project that will be used to identify the project functionality and the resulting functional requirements. The use cases outline the basic flow of the project functions.
* Section 3, Use Case Model Rationale. This section briefly describes the reasoning behind the structure of our use cases.
* Section 4, Non-Functional Requirements. This section contains the non-functional requirements that we will reference when choosing the architecture for the project.

# 2. Use Case Model for Functional Requirements

### 2.1 Graphical User Interface Use Case Models

Requirements table lookup acronym: UI

#### 2.1.1 Login UI

##### 2.1.1.1 Login UI Feature Introduction

The user login interface to verify access via permissions.

|  |  |
| --- | --- |
| Figure 2-1. Use case 1. | |
| Use Case ID: | UC1 |
| Use Case Name: | User Login/Logout |
| Actors: | User |
| Entry Condition(s): | User navigates to the web address of the application. |
| Basic Flow: | 1. The user enters their login id into the Login id text field. 2. The user enters their password into the Password field. 3. The user clicks the Ok button to submit their credentials for authentication and permissions. |
| Exit Condition(s): | The user provides credentials which have the appropriate permissions to access the home page for the web application. |
| Exceptions: | The user does not provide valid login credentials. The page will indicate that the login credentials are invalid. |
| Special Requirements: | The user credentials must exist in the database. |

Table 2-1. Use case 1.

##### 2.1.1.2 Wireframe

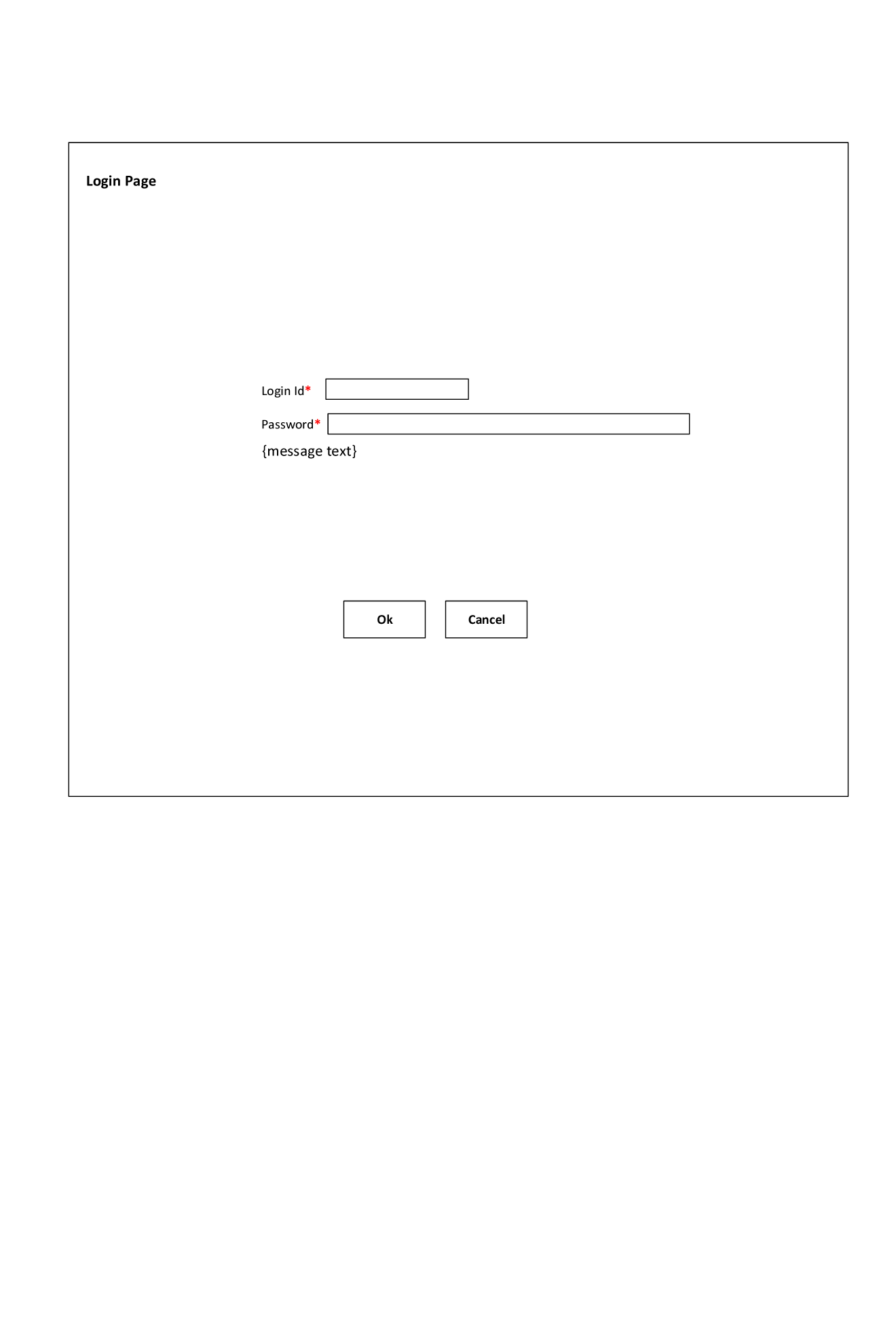


Figure 2-2. Login page wireframe.

##### 2.1.1.3 Associated Functional Requirements

**UI-FR-1**

The login user interface shall provide a text input field for the user to enter an email address as their login id.

**UI-FR-2**

The login user interface shall provide a text input field for the user to enter their password.

**UI-FR-3**

The login user interface shall provide a generic textual feedback message which will be used to notifying the user if their login attempt has failed. The message shall not provide specific information about the precise cause of the login attempt failure.

**UI-FR-4**

The login user interface shall provide a button that will allow the user to transmit their credentials for authentication.

**UI-FR-5**

The login user interface shall provide a button that will allow the user to cancel their login attempt and clear the fields.

#### 2.1.2 Home Page UI

##### 2.1.2.1 Home Page UI Feature Introduction

The home page the user will use to create, retrieve, update, and delete enterprise application integration records.

|  |  |
| --- | --- |
| Figure 2-3. Use case 2. | |
| Use Case ID: | UC2 |
| Use Case Name: | View integration records |
| Actors: | User |
| Entry Condition(s): | The user has logged in and has permission to view the records. |
| Basic Flow: | 1. The user expands either Source Applications or Destination Applications in the left side menu area. 2. The application displays the relevant information from the database within the main content area of the home page. |
| Exit Condition(s): | The user selects a different option from the left side menu or leaves the application. |
| Exceptions: | The user does not have sufficient permissions or there is no relevant information in the database. |
| Special Requirements: | The user must be logged in with valid credentials. |

Table 2-2. Use case 2.

|  |  |
| --- | --- |
| Figure 2-4. Use case 3. | |
| Use Case ID: | UC3 |
| Use Case Name: | Add integration record |
| Actors: | User |
| Entry Condition(s): | The user has logged in and has permission to add a new integration record. |
| Basic Flow: | 1. The user expands either Source Applications or Destination Applications in the left side menu area. 2. The user selects the Add option from the submenu in the left side menu area. 3. The application displays a small page where the user may enter the relevant information for the integration (TBD). 4. The user selects the Submit button to add the integration. 5. The application displays a window with a message confirming the new integration record was added. 6. The user clicks the Ok button to close the confirmation window. |
| Exit Condition(s): | The user has completed the addition process, or they have canceled prior to the record being added. |
| Exceptions: | The user does not have sufficient permissions, or the record exists in the database. |
| Special Requirements: | The user must be logged in with valid credentials. |

Table 2-3. Use case 3.

|  |  |
| --- | --- |
| Figure 2-5. Use case 4. | |
| Use Case ID: | UC4 |
| Use Case Name: | Edit integration record |
| Actors: | User |
| Entry Condition(s): | 1. The user has logged in and has permission to view and edit an integration record. 2. The user has displayed the existing records in the main content area. |
| Basic Flow: | 1. The user selects the area within the record they wish to edit. 2. The application allows the user to edit the record in place. 3. The user makes their edit and presses Enter to submit the change. 4. The application makes the update seamlessly. |
| Exit Condition(s): | The user has completed or cancelled the edit by pressing the Enter key or the Esc key respectively. |
| Exceptions: | The user does not have sufficient permissions. |
| Special Requirements: | The user must be logged in with valid credentials. |

Table 2-4. Use case 4.

|  |  |
| --- | --- |
| Figure 2-6. Use case 5. | |
| Use Case ID: | UC5 |
| Use Case Name: | Delete integration record |
| Actors: | User |
| Entry Condition(s): | 1. The user has logged in and has permission to view and delete an integration record. 2. The user has displayed the existing records in the main content area. |
| Basic Flow: | 1. The user selects the Delete icon within the record they wish to delete. 2. The application prompts the user to confirm they want to delete the record. 3. The user selects Ok to confirm the deletion. 4. The application deletes the record and updates the table in the application. |
| Exit Condition(s): | The user has completed or cancelled the deletion by pressing the Ok button or the Cancel button respectively. |
| Exceptions: | The user does not have sufficient permissions. |
| Special Requirements: | The user must be logged in with valid credentials. |

Table 2-5. Use case 5.

##### 2.1.2.2 Wireframe

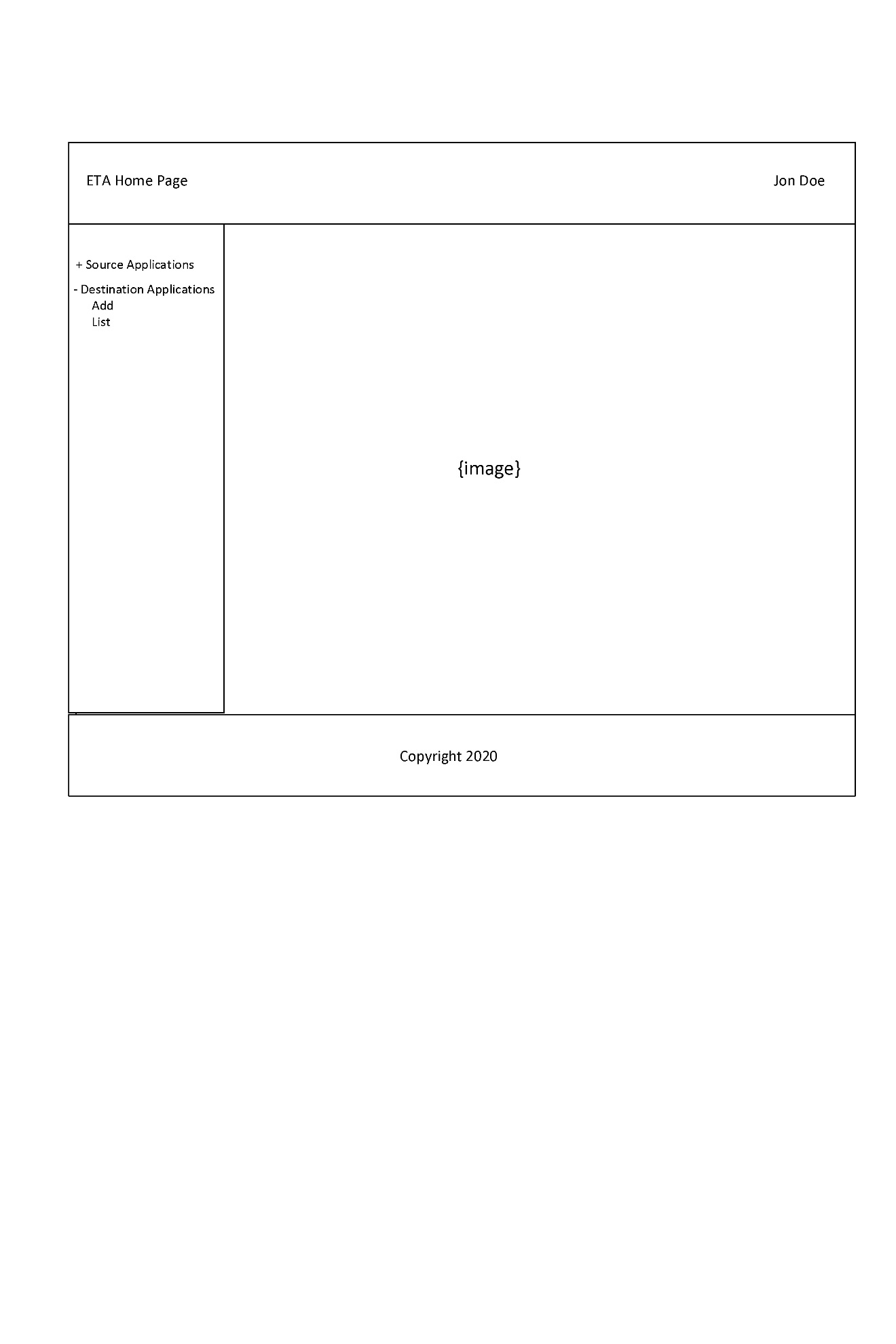


Figure 2-7. Home page wireframe.

##### 2.1.1.3 Associated Functional Requirements

**UI-FR-6**

The home page user interface shall include a header area depicting the application title and the username.

**UI-FR-7**

The home page user interface shall include a footer area depicting the appropriate copyright information.

**UI-FR-8**

The home page user interface shall include a menu area on the left side of the main body.

**UI-FR-9**

The home page user interface left side menu shall include two expandable options, one for source applications and one for destination applications. Each of these expandable options shall have permission-based options allowing the user to add new enterprise application integrations or to list all of the current enterprise application integrations.

**UI-FR-10**

The home page user interface shall include a main content area on the right side of the main body. The main content area shall feature a default image which will be replaced by the formatted table containing the enterprise application integration records.

**UI-FR-11**

The home page user interface main content area records shall allow the user to edit any specific record in place.

**UI-FR-12**

The home page user interface main content area records shall allow the user to delete a record from within the table of results.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Req. ID | Type | Priority | Description | Owner |
| UI-FR-1 | Functional | Essential | Login email | JF, EL |
| UI-FR-2 | Functional | Essential | Login password | JF, EL |
| UI-FR-3 | Functional | Essential | Generic login failure message | JF, EL |
| UI-FR-4 | Functional | Essential | Login submit button | JF, EL |
| UI-FR-5 | Functional | Essential | Login cancel button | JF, EL |
| UI-FR-6 | Functional | Essential | Home page header area | JF, EL |
| UI-FR-7 | Functional | Essential | Home page footer area | JF, EL |
| UI-FR-8 | Functional | Essential | Home page menu area | JF, EL |
| UI-FR-9 | Functional | Essential | Home page menu functionality | JF, EL |
| UI-FR-10 | Functional | Essential | Home page main area | JF, EL |
| UI-FR-11 | Functional | Essential | Home page record edit | JF, EL |
| UI-FR-12 | Functional | Essential | Home page record delete | JF, EL |

Table 2-6. UI Functional Requirements.

### 2.2 Authentication Layer Use Case Models

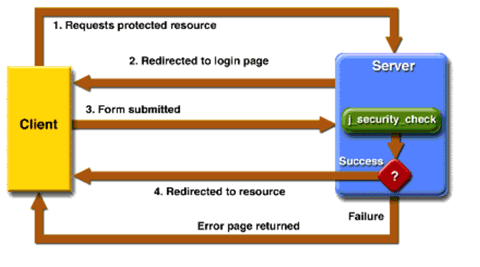


Figure 2-8. Authentication layer.

|  |  |
| --- | --- |
| Use Case ID: | UC6 |
| Use Case Name: | Authentication |
| Actors: | User |
| Entry Condition(s): | Must have permission to access the resource. |
| Basic Flow: | 1. The user requests access to a protected resource. 2. The Web server returns a dialog box that requests the username and password. 3. The client submits the username and password to the server. 4. The server validates the credentials and, if successful, returns the requested resource. |
| Exit Condition(s): | The user has completed or cancelled the log in by pressing the sign out button respectively. |
| Exceptions: | n/a |
| Special Requirements: | User should have a valid credential. |

Table 2-7. Use case 6.

### 2.3 Web Services Use Case Models

Requirements table lookup acronym: WS

#### 2.3.1 Web Services Introduction

The web services shall receive either POST, GET, PUT or DELETE requests based on user’s actions and input, and shall return appropriate results.

|  |  |
| --- | --- |
| Figure 2-9. Use case 7. | |
| Use Case ID: | UC7 |
| Use Case Name: | Search Records |
| Actors: | User, Web Service |
| Entry Condition(s): | The user is logged in and has permission to access to the service. |
| Basic Flow: | 1. The user sends a GET request to the web service by selecting the search option and inputting record information. 2. The web service returns appropriate integration records based on search criteria. 3. The application displays the results of the GET request in the main content area. |
| Exit Condition(s): | The user performs other actions or exits the application. |
| Exceptions: | The user has insufficient permissions. The database lacks information needed. The web service encounters an error with the GET request. |
| Special Requirements: | The user is logged in and must have valid credentials. |

Table 2-8. Use case 7.

|  |  |
| --- | --- |
| Figure 2-10. Use case 8. | |
| Use Case ID: | UC8 |
| Use Case Name: | View Records |
| Actors: | User, Web Service |
| Entry Condition(s): | The user is logged in and has permission to access to the service. |
| Basic Flow: | 1. The user sends a GET request to the web service by expanding either the Source Applications or Destination Applications. 2. The web service returns appropriate integration records based on search criteria. 3. The application displays the results of the GET request in the main content area. |
| Exit Condition(s): | The user performs other actions or exits the application. |
| Exceptions: | The user has insufficient permissions. The database lacks information needed. The web service encounters an error with the GET request. |
| Special Requirements: | The user is logged in and must have valid credentials. |

Table 2-9. Use case 8.

|  |  |
| --- | --- |
| Figure 2-11. Use case 9. | |
| Use Case ID: | UC9 |
| Use Case Name: | View Records (Limited) |
| Actors: | User, Web Service |
| Entry Condition(s): | The user is logged in and has limited permission to access the service. |
| Basic Flow: | 1. The user sends a GET request to the web service by expanding either the Source Applications or Destination Applications, whichever option the user has permission to view. 2. The web service returns appropriate integration records based on search criteria. 3. The application displays the results of the GET request in the main content area. |
| Exit Condition(s): | The user performs other actions or exits the application. |
| Exceptions: | The user has insufficient permissions. The database lacks information needed. The web service encounters an error with the GET request. |
| Special Requirements: | The user is logged in, must have valid credentials, and has limited permission to view applications. |

Table 2-10. Use case 9.

|  |  |
| --- | --- |
| Figure 2-12. Use case 10. | |
| Use Case ID: | UC10 |
| Use Case Name: | Add Records |
| Actors: | User, Web Service |
| Entry Condition(s): | The user is logged in and has permission to add records. |
| Basic Flow: | 1. The user clicks the Add button on either the Source Applications or Destination Applications, whichever option the user has permissions to add. 2. The application will display a form to be filled out by the user with the necessary application fields. 3. A POST request to the web service will be sent once the form is submitted. |
| Exit Condition(s): | The user performs other actions or exits the application. |
| Exceptions: | The user has insufficient permissions. The database lacks information needed. The web service encounters an error with the POST request. |
| Special Requirements: | The user is logged in, must have valid credentials, and has sufficient permissions to add applications. |

Table 2-11. Use case 10.

|  |  |
| --- | --- |
| Figure 2-13. Use case 11. | |
| Use Case ID: | UC11 |
| Use Case Name: | Add Records (Limited) |
| Actors: | User, Web Service |
| Entry Condition(s): | The user is logged in and has permission to add records. |
| Basic Flow: | 1. The user clicks the Add button on either the Source Applications or Destination Applications, whichever option the user has permissions to add. 2. The application will display a form to be filled out by the user with the necessary application fields. 3. A POST request to the web service will be sent once the form is submitted. |
| Exit Condition(s): | The user performs other actions or exits the application. |
| Exceptions: | The user has insufficient permissions. The database lacks information needed. The web service encounters an error with the POST request. |
| Special Requirements: | The user is logged in, must have valid credentials, and has sufficient permissions to add applications. |

Table 2-12. Use case 11.

|  |  |
| --- | --- |
| Inserting image...  Figure 2-14. Use case 12. | |
| Use Case ID: | UC12 |
| Use Case Name: | Edit Records |
| Actors: | User, Web Service |
| Entry Condition(s): | The user is logged in and has permission to edit records. |
| Basic Flow: | 1. The user clicks the Edit button to the web service by expanding either the Source Applications or Destination Applications, whichever option the user has permission to edit. 2. The web service returns appropriate integration records based on search criteria. 3. With sufficient permissions, an Edit button will display next to the record. 4. The application will display a form to be edited by the user with the necessary application fields. 5. A PUT request to the web service will be sent once the form is submitted. |
| Exit Condition(s): | The user performs other actions or exits the application. |
| Exceptions: | The user has insufficient permissions. The database lacks information needed. The web service encounters an error with the PUT request. |
| Special Requirements: | The user is logged in, must have valid credentials, and has limited permission to view applications. |

Table 2-13. Use case 12.

|  |  |
| --- | --- |
| Figure 2-15. Use case 13. | |
| Use Case ID: | UC13 |
| Use Case Name: | Edit Records (Limited) |
| Actors: | User, Web Service |
| Entry Condition(s): | The user is logged in and has permission to edit records. |
| Basic Flow: | 1. The user clicks the Edit button to the web service by expanding either the Source Applications or Destination Applications, whichever option the user has permission to edit. 2. The web service returns appropriate integration records based on search criteria. 3. With sufficient permissions, an Edit button will display next to the record. 4. The application will display a form to be edited by the user with the necessary application fields. 5. A PUT request to the web service will be sent once the form is submitted. |
| Exit Condition(s): | The user performs other actions or exits the application. |
| Exceptions: | The user has insufficient permissions. The database lacks information needed. The web service encounters an error with the PUT request. |
| Special Requirements: | The user is logged in, must have valid credentials, and has limited permission to view applications. |

Table 2-14. Use case 13.

|  |  |
| --- | --- |
| Inserting image...  Figure 2-16. Use case 14. | |
| Use Case ID: | UC14 |
| Use Case Name: | Delete Records |
| Actors: | User, Web Service |
| Entry Condition(s): | The user is logged in and has permission to delete records. |
| Basic Flow: | 1. The user clicks the View button on either the Source Applications or Destination Applications, whichever option the user has permissions to view. 2. The web service returns appropriate integration records based on search criteria. 3. With sufficient permissions, a Delete button will display next to the record. 4. When the user clicks on the Delete button a confirmation prompt will appear. 5. After confirming to proceed with the deletion, the application will send a DELETE request to the Web Service and the record will be removed. |
| Exit Condition(s): | The user performs other actions or exits the application. |
| Exceptions: | The user has insufficient permissions. The database lacks information needed. The web service encounters an error with the DELETE request. |
| Special Requirements: | The user is logged in, must have valid credentials, and has sufficient permissions to add applications. |

Table 2-15. Use case 14.

|  |  |
| --- | --- |
| Figure 2-17. Use case 15. | |
| Use Case ID: | UC15 |
| Use Case Name: | Delete Records (Limited) |
| Actors: | User, Web Service |
| Entry Condition(s): | The user is logged in and has permission to delete records. |
| Basic Flow: | 1. The user clicks the View button on either the Source Applications or Destination Applications, whichever option the user has permissions to view. 2. The web service returns appropriate integration records based on search criteria. 3. With sufficient permissions, a Delete button will display next to the record. 4. When the user clicks on the Delete button a confirmation prompt will appear. 5. After confirming to proceed with the deletion, the application will send a DELETE request to the Web Service and the record will be removed. |
| Exit Condition(s): | The user performs other actions or exits the application. |
| Exceptions: | The user has insufficient permissions. The database lacks information needed. The web service encounters an error with the DELETE request. |
| Special Requirements: | The user is logged in, must have valid credentials, and has sufficient permissions to add applications. |

Table 2-16. Use case 15.

#### 2.3.2 Associated Functional Requirements

**WS-FR-1**

**The application shall send GET requests to the web service when a user searches for records.**

**WS-FR-2**

**The application shall send GET requests to the web service when a user views records.**

**WS-FR-3**

**The application shall send POST requests to the web service when a user creates a new record.**

**WS-FR-4**

**The application shall send PUT request to the web service when a user edits records.**

**WS-FR-5**

**The application shall send DELETE requests to the web services when a user deletes records.**

**WS-FR-6**

**The application shall display appropriate results based on user permissions and web service requests.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Req. ID | Type | Priority | Description | Owner |
| WS-FR-1 | Functional | Essential | Search records send GET | RF, AA |
| WS-FR-2 | Functional | Essential | View records send GET | RF, AA |
| WS-FR-3 | Functional | Essential | Create records send POST | RF, AA |
| WS-FR-4 | Functional | Essential | Edit records send PUT | RF, AA |
| WS-FR-5 | Functional | Essential | Delete records send DELETE | RF, AA |
| WS-FR-6 | Functional | Essential | Appropriate and limited results | RF, AA |

Table 2-17. Web Services Layer Functional Requirements.

### 2.4 Persistence Layer Use Case Models

Requirements table lookup acronym: PL

#### 2.4.1 Persistence Layer Introduction

The persistence layer shall provide information about the database and common CRUD functionalities.

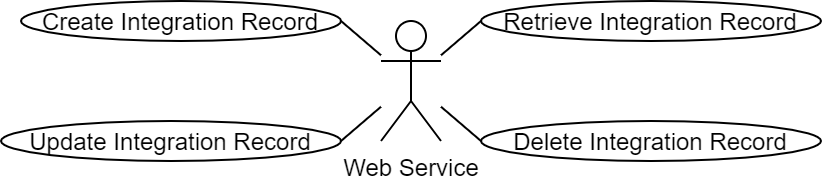


Figure 2-18. Use case 16.

#### 2.4.2 Associated Functional Requirements

**PL-FR-1**

**The database shall be able to create entries.**

**PL-FR-2**

**The database shall be able to retrieve entries.**

**PL-FR-3**

**The database shall be able to update entries.**

**PL-FR-4**

**The database shall be able to delete entries.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Req. ID | Type | Priority | Description | Owner |
| PL-FR-1 | Functional | Essential | Database Create. | RF, AA |
| PL-FR-2 | Functional | Essential | Database Retrieve. | RF, AA |
| PL-FR-3 | Functional | Essential | Database Update. | RF, AA |
| PL-FR-4 | Functional | Essential | Database Delete. | RF, AA |

Table 2-18. Persistence Layer Functional Requirements.

# 3. Use Case Model Rationale

|  |  |
| --- | --- |
| UC6 | The application shall provide secure authentication through the application interface. |
| 1. Will the credentials be encrypted on the back end? 2. Will we implement the authentication layer directly or will we use a third-party source? 3. Will it be possible to add new users via the application interface? | |
| Assumptions   * A server containing the necessary credentials will be available upon deployment. | |
| Option 1 | Basic authentication using username and password. |
| Option 2 | JSON token based authentication via OAuth2.0 protocol. |
| Option 3 | XML token based authentication via SAML 2.0 protocol. |
| Choice | Option 1 |
| Rationale | Option 1 is the best option due to time constraints and because the customer intends to add their own authentication measures after the application has been deployed. |

Table 3-1. Use case 6 rationale.

# 4. Non-Functional Requirements

### 4.1 Usability

NFR1: The site shall function as a single-page application per the client’s request. The integration data will be displayed using a sortable, table style grid.

### 4.2 Performance

NFR2: The site shall respond to user input within a reasonable amount of time.

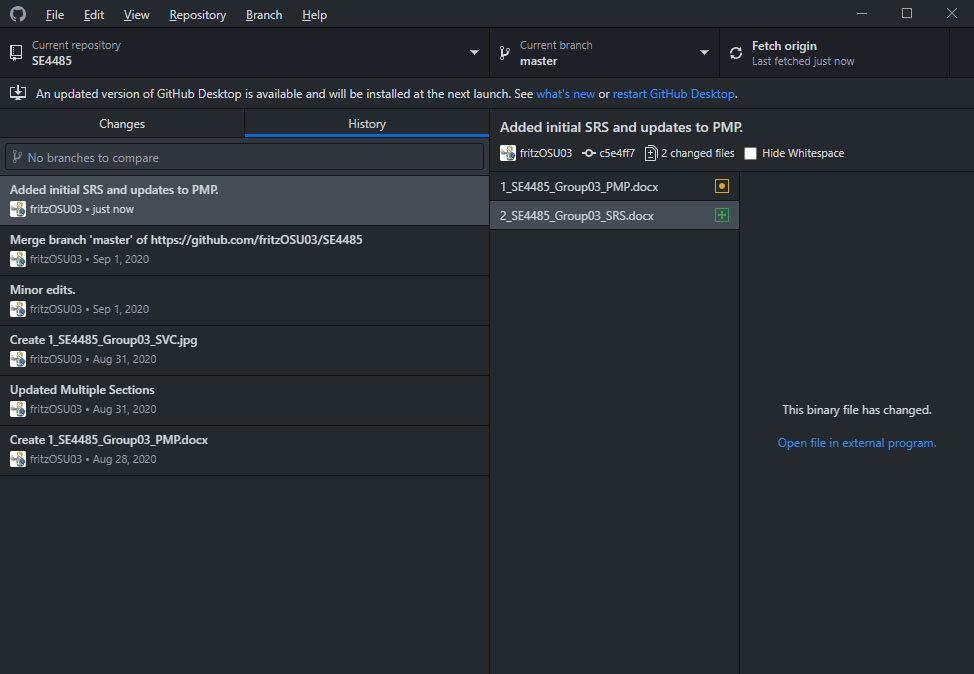
### 4.3 Portability

NFR3: The application shall function in modern web browsers.

### 4.4 Security

NFR4: The system shall encrypt and store user credentials in an authentication database per the client’s request.

# 5. Evidence of Configuration Management



# 6. References

IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

END OF DOCUMENT