BeicipFranlabAsia

SYSTEM REQUIREMENTS SPECIFICATIONS

(SRS)

CORE APPLICATION
(Module 2 - Existing CORE Admin Features)

FOR

BEICIP-FRANLAB ASIA

Document Title:	SYSTEM REQUIREMENT SPECIFICATION (Existing CORE Admin Features Workflow)
Document Version:	1.4
Version Date:	17/3/2021
Document Status:	Approved / Reject
Distribution:	CORE team

	Name	Signature	Role	Date
Prepared by:	Zainoor Asykin Zainol Abidin		System Analyst	17/3/2021
Reviewed by:	Ikhwan Yusoff		Project Coordinator	17/3/2021
Reviewed by:	Furqan Raza		Technical	18/03/2021
Endorsed by:	Tuan Luqman		Technical Solutions Architect	22/3/2021
Endorsed by:	Afifi Aznan		Head of Business Integration & Transformation	22/03/2021

Document Revision History

Revision Date	Version	Summary of Changes	Revised by
4/2/2021	rev 1.0	Draft	Zainoor Asykin Zainol Abidin
11/2/2021	rev 1.1	Updates based on remarks from Beicip	Zainoor Asykin Zainol Abidin
9/3/2021	rev 1.2	Added missing use cases and amended based on the feedbacks	Zainoor Asykin Zainol Abidin
12/3/2021	rev 1.3	Added missing use case (CORE_ECA-02), added portal setting in basic path (CORE_ECA-05), update destination backup, status log and post-condition (CORE_ECA-07), removed unnecessary things in indexing and searching (CORE_ECA-09)	Zainoor Asykin Zainol Abidin
17/3/2021	rev 1.4	Added sorting metadata flows in basic path (CORE_ECA-01, CORE_ECA-02), changed use case name (CORE_ECA-02), update basic path and post condition for 'Local Backups' and 'Schedules' (CORE_ECA-07)	Zainoor Asykin Zainol Abidin

Document Classification

SECRET or RAHSIA	
CONFIDENTIAL or SULIT	
INTERNAL USE or UNTUK DALAMAN	X
OPEN or TERBUKA	

Table of Contents

List of Tables

List of Figures

1. Introduction

The System Requirement Specification (SRS) is provided to aid user through the important components, variables and options necessary to provide a system that meets the user's need in the most cost-effective, high quality and secure method possible. This application, named Centralized Online (Data) Repository Ecosystem (CORE) is to provide Beicip-Franlab Asia with the organized and centralized platform which include user management, document management and project management and more.

Following documents with specific intention are attached together with this SRS and should be always cross referenced to each other when using the document as well as developing the system.

1. User Requirement Specification (URS) – Describe the important components, variables and options necessary to provide a system that meets the user's need

2. Wireframe booklet – Provide mockup, Entity Relationship Diagram (ERD) and database mapping

1.1 Purpose

This document is intended as a guide in the development of Existing CORE Admin Features workflow which is part of a CORE elements content. Due to rapid development requirement, the system development documentation (URS and SRS) is broken down to every single CORE workflow element hence the User and Developer team would be able to focus to only specific content and smaller scope.

It will go through several stages during the project:

1. **Draft**: The first version or draft version is compiled after requirements have been discovered, recorded, classified, and prioritized.

2. **Proposed**: The draft document is then proposed as a potential requirements specification for the project. The proposed document should be reviewed by several parties, who may comment on any requirements and any priorities, either to agree, to disagree, or to identify missing requirements. Readers include end-users, developers, project managers, and any other stakeholders. The document may be amended and re-proposed several times before moving to the next stage.

3. **Validated**: Once the various stakeholders have agreed to the requirements in the document, it is considered validated.

4. **Approved**: The validated document is accepted by representatives of each party of stakeholders as an appropriate statement of requirements for the project. The developers the use the requirements document as a guide to implement and to check the progress of the project as it develops.

1.2 Project Scope

This web-based application, named Centralized Online (Data) Repository Ecosystem (CORE) developed to enhance data management and business processes through a typical web-based and database integrative process that includes comprising of data source input, collector, analyzer and dashboard output.

In general, 13 workflows have been identified for current development stage. They are as following:

1. User and Role Management
2. Existing CORE Admin Features
3. Document Management
4. Template Management
5. Data Loading/Migration
6. Project-based Document Management
7. Management Dashboard
8. Build in OCR
9. Deep Search Capability
10. GIS/Mapping Module
11. Search Engine
12. Analytics
13. Emailing Mechanism

1.3 Intended Audience

We expect that this document will be used by people with different skill sets.

Name	Position/ Role	Team	Action
Furqan Reza	Technical	BFA	Review the SRS/URS
Tuan Luqman	Technical Solutions Architect	BFA	Endorse the SRS/URS
Afifi Aznan	Head of Business Integration & Transformation	BFA	Endorse the SRS/URS

Table 1 - Intended Audience

1.4 Version History

Version	Date	Author	Comments
rev 1.0	4/2/2021	Zainoor Asykin Zainol Abidin	Initial release, draft
rev 1.1	11/2/2021	Zainoor Asykin Zainol Abidin	Updates based on remarks from Beicip
rev 1.2	9/3/2021	Zainoor Asykin Zainol Abidin	Added missing use cases and amended based on the feedbacks
rev 1.3	12/3/2021	Zainoor Asykin Zainol Abidin	Added missing use case (CORE_ECA-02), added portal setting in basic path (CORE_ECA-05), update destination backup, status log and post-condition (CORE_ECA-07), removed unnecessary things in indexing and searching (CORE_ECA-09)
rev 1.4	17/3/2021	Zainoor Asykin Zainol Abidin	Added sorting metadata flows in basic path (CORE_ECA-01, CORE_ECA-02), changed use case name (CORE_ECA-02), update basic path and post condition for 'Local Backups' and 'Schedules' (CORE_ECA-07)

Table 2 - Version History

1.5 Acronyms and Definitions

Acronym	Definition
API	Application Programming Interface
SRS	System requirement specification
CORE	Centralized Online (Data) Repository Ecosystem
URS	User requirement specification
ECA	Existing CORE Admin Features
SysML	Systems Modelling Language

Table 3 – Acronyms and Definitions

2. Overall Description

2.1 Objective

Main objective of Existing CORE Admin Features (ECA) workflow is to provide user functionality access within the system in the CORE application for the administrator to manage:

a. Deep search metadata management
b. Well Management
c. Health check-up
d. Setting up OCR engine & troubleshooting
e. Bulk load workflow
f. Basic and advanced administrator settings,
g. Backup management
h. Search index management.
i. Project Tracker Setting

2.2 System Concept

In Existing Core Admin Features (ECA) workflow, the Super Admin is able to manage the crucial functionalities of CORE application as mentioned above. As the module involves mainly in the system management setting, it's important to highlight that any changes to the settings may affect the functionalities of the system as a whole.

For example, it's crucial to setup the backup management in the agreeable interval i.e. every 3 days for both application and database backup. Other than that, for Deep Search Setting, the application consists of 6 sub-settings which are Category, Well, Stakeholder, Department, Location and Event.

For Health Check-up, it's important for Super Admin's information to take note of any documents that may fall under certain issues. Those issues include Broken Links, Special Characters, Documents without Content, Duplicate Documents and Too Long Paths. This would be beneficial to plan for any further actions.

3. Architecture

3.1 System Architecture
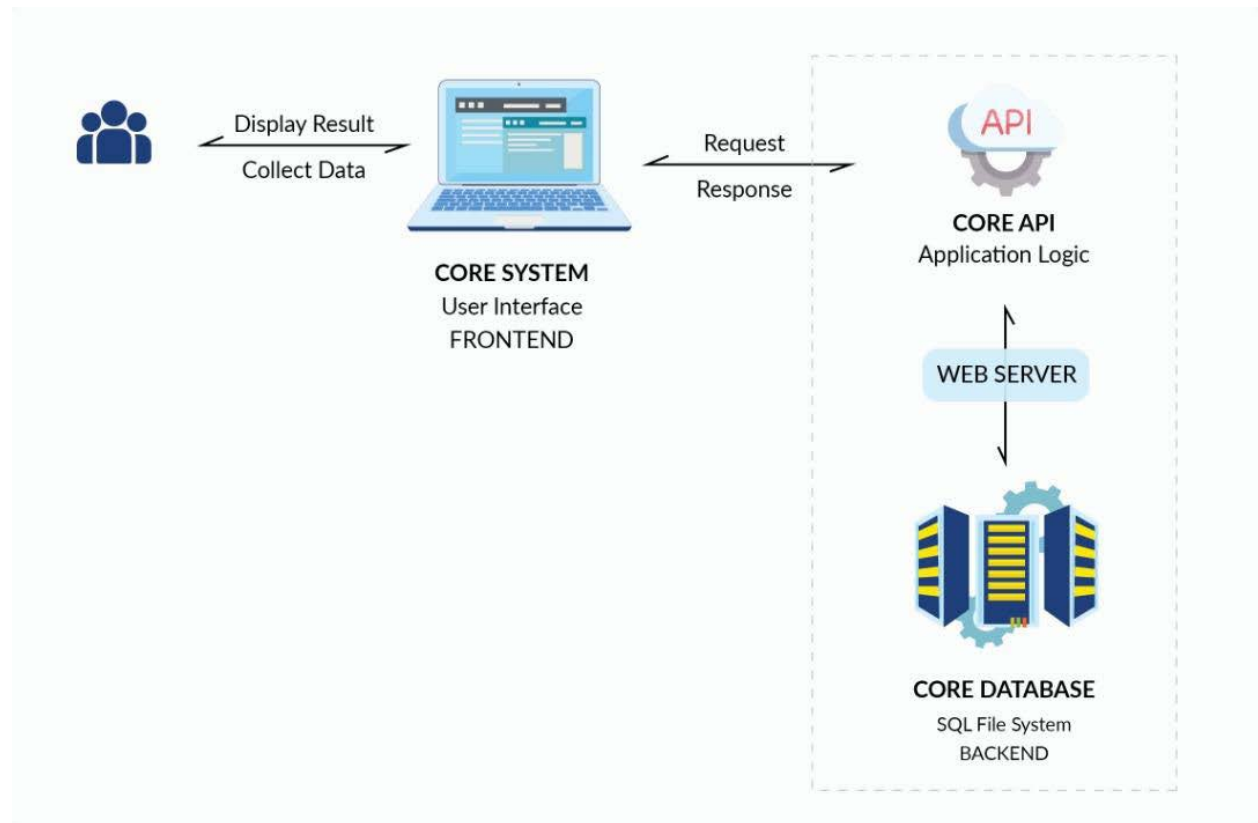
Figure 1 - System Architecture

3.2 Application Architecture
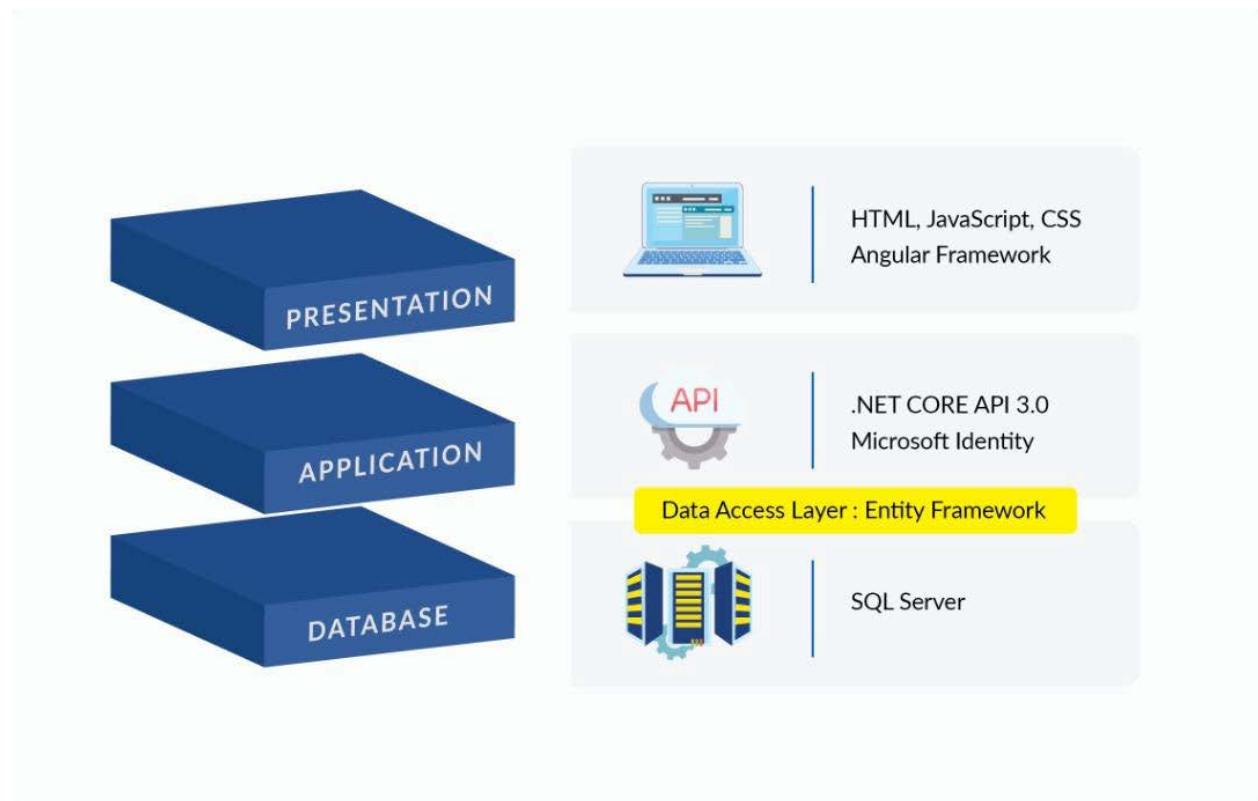
Figure 2 - Application Architecture

4. Functional Requirements

In software and systems engineering, a use case is a list of actions or event steps typically defining the interactions between a role (known in the Unified Modeling Language as an actor) and a system to achieve a goal. The actor can be a human or other external system.

In systems engineering use cases are used at a higher level within software engineering often representing missions or stakeholder goals. The detailed requirements may then be captured in the Systems Modeling Language (SysML) or as contractual statements.

Use Case Model was chosen to describe the functionality of CORE as it is easily understanding by Technical person or Non-Technical person. The Use Case Model content of Use Case Diagram, Use Case List, Actors Description and Use Case Description.

The CORE Use-Case Model is designing on micro view to make it clearer and self- explanatory.

4.1 Use Case Diagram
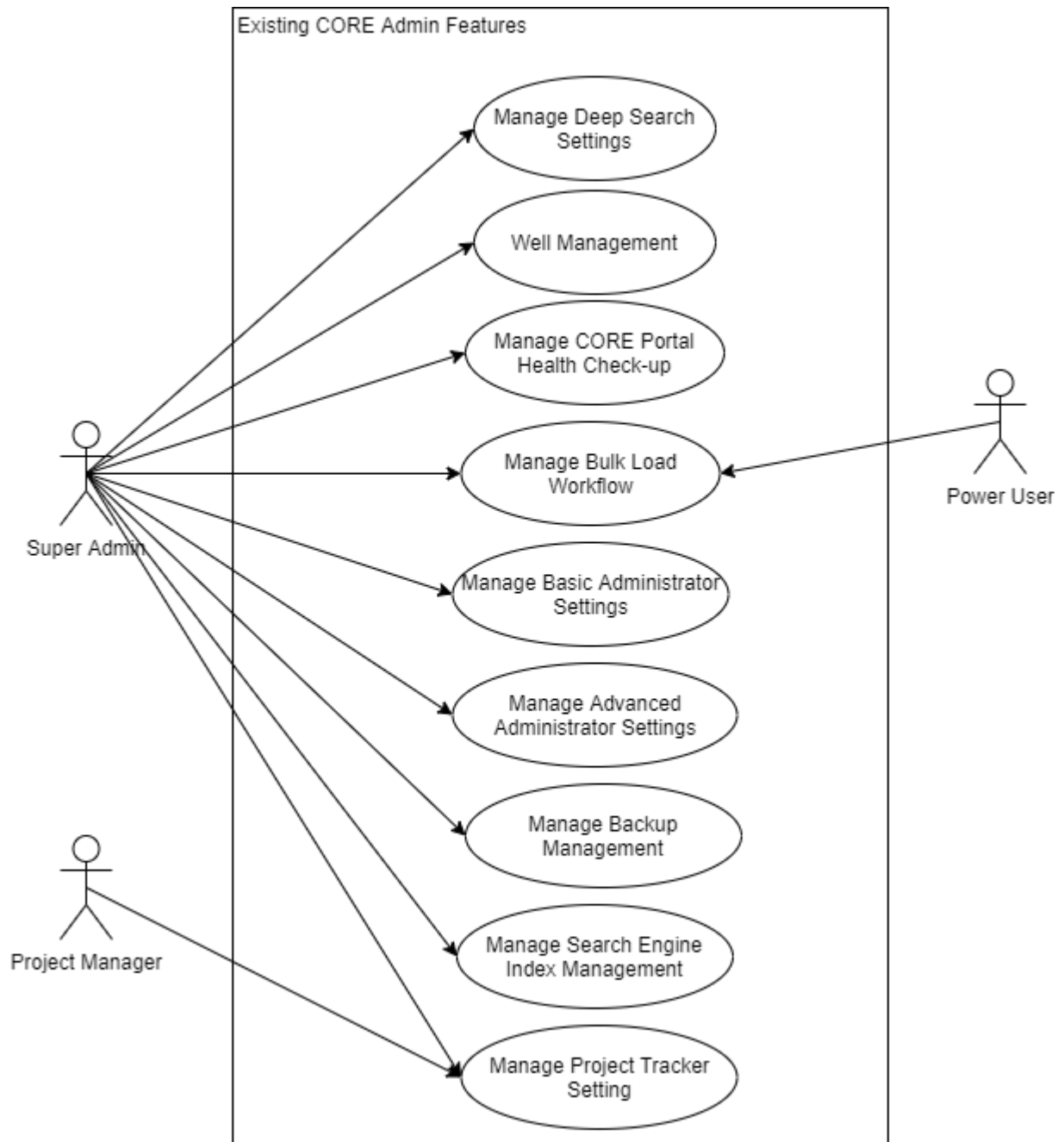
Figure 3 - Use Case Diagram

4.2 Use Case List

Use-Case ID	Use-Case Name	Description
CORE_ECA-01	Manage Deep Search Setting	This Use Case is for the actor to manage deep search setting (add, edit, delete and sort metadata)
CORE_ECA-02	Well Management	This Use Case is for the actor to manage well management (add, edit, delete and sort well)
CORE_ECA-03	Manage CORE Portal Health Check-up	This Use Case is for the actor to monitor and maintain the good document management practices and quality of the core Portal.
CORE_ECA-04	Manage Bulk Load Workflow	This Use Case is for the actor to manage bulk load workflow. • Document attachment is made easy and faster for administrator to load several documents at once. • Bulk Uploader provide the ability to bulk uploading document to CORE by given network folder.
CORE_ECA-05	Manage Basic Administrator Settings	This is the Basic Setting for CORE and control CORE functionality.
CORE_ECA-06	Manage Advanced Administrator Settings - Provide pattern to detect date from the network paths	This Use Case is for the actor to manage advanced administrator settings. • For Advanced Setting is to extract text for auto complete • The advance settings control two specific capabilities of the portal. It allowed admin (with REGEX expertise) to provide

		- pattern to detect date from the network paths. - It also provides admin to customize the email contents that CORE send to the user.
CORE_ECA-07	Manage Backup Management	This Use Case is for the actor to manage backup for the system, comprising both database and application backup
CORE_ECA-08	Manage Search Engine Index Management	This Use Case is for the actor to manage search engine index
CORE_ECA-09	Manage Project Tracker Setting	This Use Case is for the actor to manage the project tracker setting

Table 4 - Use Case List

4.3 Use Case Description

I.   CORE_ECA-01 Manage Deep Search Setting

Use Case Component:	Description
Use Case ID:	CORE_ECA-01
Use Case Name:	Manage Deep Search Setting
Actor:	Super Admin
Pre-Condition:	1. The actor must exist in the CORE. The actor must have permission to manage deep search setting. 2. The actor can only add subcategory after successfully adding new category.
Basic Path:	1. Actor login the system. 2. System will check the actor permission. 3. Actor navigate to CORE Menu. 4. Actor click on the Deep Search Setting button. **Category** a) Actor select the Category tab. b) Actor create a new category by clicking on Add New button. c) Actor edit the existing category by clicking on Edit button. d) Actor delete the existing category by clicking on Delete button. e) In the confirmation page, the system will check if there is any existing document belongs to that category. f) Actor needs to assign different Category before proceeding to Confirm Delete. g) Actor can control the order of the Category metadata by clicking on Update Sorting button.

Subcategory

    a) Actor click on the Subcategories button under Category tab.

    b) Actor create a new subcategory by clicking on Add New button.

    c) Actor edit the Subcategories by clicking on Edit button.

    d) Actor delete the Subcategories by clicking on Delete button.

    e) In the confirmation page, the system will check if there is any existing document belongs to that subcategory.

    f) Actor needs to assign different Subcategory before proceeding to Confirm Delete.

    g) Actor can control the order of the Subcategory metadata by clicking on Update Sorting button.

Stakeholder

    a) Actor select the Stakeholder tab

    b) Actor create a new stakeholder by clicking on 'Add New' button

    c) Actor edit the existing stakeholder by clicking on Edit button.

    d) Actor delete the existing stakeholder by clicking on Delete button.

    e) In the confirmation page, the system will check if there is any existing document belongs to that stakeholder.

    f) Actor needs to assign different stakeholder before proceeding to Confirm Delete.

    g) Actor can control the order of the Stakeholder metadata by clicking on Update Sorting button.

Department

    a) Actor select the Department tab

    b) Actor create a new department by clicking on 'Add New' button

    c) Actor edit the existing department by clicking on Edit button.

d) Actor delete the existing department by clicking on Delete button.

e) In the confirmation page, the system will check if there is any existing document belongs to that department.

f) Actor needs to assign different department before proceeding to Confirm Delete.

g) Actor can control the order of the Department metadata by clicking on Update Sorting button.

Location

a) Actor select the Location tab

b) Actor create a new location by clicking on 'Add New' button

c) Actor edit the existing location by clicking on Edit button.

d) Actor delete the existing location by clicking on Delete button.

e) In the confirmation page, the system will check if there is any existing document belongs to that location.

f) Actor needs to assign different location before proceeding to Confirm Delete.

g) Actor can control the order of the Location metadata by clicking on Update Sorting button.

Event

a) Actor select the Category tab

b) Actor create a new subcategory by clicking on 'Add New' button.

c) Actor edit the existing event by clicking on Edit button.

d) Actor delete the existing event by clicking on Delete button.

e) In the confirmation page, the system will check if there is any existing document belongs to that event.

f) Actor needs to assign different event before proceeding to Confirm Delete.

g) Actor can control the order of the Event metadata by clicking on Update Sorting button.

Alternative Path(s):	N/A
Exception Path(s):	1. If the category/sub-category/departments/ stakeholders/event/location already exists, system will prompt message "Category/Subcategory/departments/ stakeholders/ event/location already exist." 2. If the actor deletes category that contained sub-category, the system prompts message "You cannot delete a category which have sub-categories. Please delete sub-categories first."
Post-Conditions:	1. Newly created category, stakeholder, department, event and location will appear in the tab. 2. The documents belonged to the deleted category, subcategory, departments, stakeholders, events and locations will be assigned to the other respective metadata, as selected by the user. 3. Deleted category/subcategory, departments, stakeholder, department, event and location will disappear from the tab. 4. Metadata will be sorted accordingly based on the actor's preference.
Business Rules:	1. As the actor, I want to add, edit, delete and sort category or subcategory. 2. As the actor, I want to add, edit, delete and sort department. 3. As the actor, I want to add, edit, delete and sort stakeholders. 4. As the actor, I want to add, edit, delete and sort event. 5. As the actor, I want to add, edit, delete and sort location.
Wireframe:	Refer Section 1.1, Figure 1 - Figure 8 ECA wireframe booklet
Data Table:	Refer Section 1.1.2, ECA wireframe booklet

Supporting Documents:	1. URS ECA
	2. ECA Wireframe Booklet
Usage Frequency:	Often
Criticality	High
Dependencies	N/A

Change History	Author	Version	Explanation
	Zainoor Asykin Zainol Abidin	1.4	Added sorting metadata flows in basic path

Table 5 - CORE_ECA-01 Manage Deep Search Setting

II. CORE_ECA-02 Well Management

Use Case Component:	Description
Use Case ID:	CORE_ECA-02
Use Case Name:	Well Management
Actor:	Super Admin
Pre-Condition:	1. The actor must exist in the CORE. The actor must have permission to manage well management.
Basic Path:	1. Actor login the system.
	2. System will check the actor permission.
	3. Actor navigate to CORE Menu.
	4. Actor click on the Well Management button.
	5. Actor create a new well by clicking on Add New button.
	6. Actor edit the existing well by clicking on Edit button.
	7. Actor delete the existing well by clicking on Delete button.
	8. In the confirmation page, the system will check if there is any existing document belongs to that well.

	9. Actor needs to assign different well before proceeding to Confirm Delete. 10. Actor can control the order of the Well metadata by clicking on Update Sorting button.
Alternative Path(s):	N/A
Exception Path(s):	1. If the well already exists, system will prompt message "Well already exist."
Post-Conditions:	1. Newly created well will appear in the tab. 2. The documents belonged to the deleted well will be assigned to the other respective metadata, as selected by the user. 3. Deleted well will disappear from the tab. 4. Metadata will be sorted accordingly based on the actor's preference.
Business Rules:	1. As the actor, I want to create, edit, delete and sort well.
Wireframe:	Refer Section 1.1, Figure 9 - Figure 11 ECA wireframe booklet
Data Table:	Refer Section 1.2.2, ECA wireframe booklet
Supporting Documents:	1. URS ECA 2. ECA Wireframe Booklet
Usage Frequency:	Often
Criticality	High
Dependencies	N/A

Change History	Author	Version	Explanation
	Zainoor Asykin Zainol Abidin	1.4	Changed use case name, add sorting metadata flows in basic path

Table 6 – CORE_ECA-02 Well Management

III.   CORE_ECA-03 Manage CORE Health Check-up

Use Case Component:	Description
Use Case ID:	CORE_ECA-03
Use Case Name:	Manage CORE Health Check-up
Actor:	Super Admin
Pre-Condition:	The actor must exist in the CORE. The actor must have permission to manage CORE Health Check-up.
Basic Path:	1. Actor login the system. 2. System will check the actor permission. 3. Actor navigates to CORE Menu. 4. Actor click on the CORE Health Check-up button 5. Actor can check documents with broken links, special characters, document without content, duplicate documents and too long paths.
Alternative/Exception Path(s):	N/A
Post-Conditions:	N/A
Business Rules:	As the actor, I want to manage CORE portal Health Check-up.
Wireframe:	Refer Section 1.3, Figure 12 ECA wireframe booklet
Data Table:	Refer Section 1.3.2, ECA wireframe booklet
Supporting Documents:	1. URS ECA 2. ECA Wireframe Booklet
Usage Frequency:	Often
Criticality	High

Dependencies	N/A		
Change History	Author	Version	Explanation
	Zainoor Asykin Zainol Abidin	1.2	Added missing use cases and amended based on the feedbacks

Table 7 - CORE_ECA-03 Manage CORE Health Check-up

IV.CORE_ECA-04 Manage Bulk Load Workflow

Use Case Component:	Description
Use Case ID:	CORE_ECA-04
Use Case Name:	Manage Bulk Load Workflow
Actor:	Super Admin, Power User
Pre-Condition:	The actor must exist in the CORE. The actor must have permission to manage Bulk Load Workflow
Basic Path:	1. Actor login the system. 2. System will check the actor permission. 3. Actor navigate to CORE Menu 4. Actor click on the Bulk Uploader button (refer CORE_DM-01, CORE_DM-02 in Document Management DM SRS)
Alternative/Exception Path(s):	N/A
Post-Conditions:	N/A
Business Rules:	As the actor, I want to manage Bulk Load Workflow.

Wireframe:	Refer Section 1.1, Figure 1 - Figure 4 DM wireframe booklet
Data Table:	Refer Section 1.1.2, DM wireframe booklet
Supporting Documents:	1. URS ECA 2. ECA Wireframe Booklet 3. SRS DM 4. DM Wireframe Booklet
Usage Frequency:	Often
Criticality	High
Dependencies	N/A

Change History	Author	Version	Explanation
	Zainoor Asykin Zainol Abidin	1.2	Added missing use cases and amended based on the feedbacks

Table 8 - CORE_ECA-04 Manage Bulk Load Workflow

V.   CORE_ECA-05 Manage Basic Administrator Settings

Use Case Component:	Description
Use Case ID:	CORE_ECA-05
Use Case Name:	Manage Basic Administrator Settings
Actor:	Super Admin
Pre-Condition:	The actor must exist in the CORE. The actor must have permission to manage Basic Administrator Settings

Basic Path:	1. Actor login the system. 2. System will check the actor permission. 3. Actor navigates to CORE Menu. 4. Actor clicks on the Admin Settings button. 5. Actor click on the Basic Settings tab. **CORE** a) Actor click on the CORE tab under Basic Settings tab. b) Actor can make changes on the CORE (main folder on network, duplicate document detection, insert special characters, document path max length, max document size (MB), minimum characters for extracted document content, default department and default location) and click on Save Changes to save the changes made. **Portal** a) Actor click on the Portal tab under Basic Settings tab. b) Actor can make changes on the company logo, portal logo, company name, login page background picture, site title and CORE portal short name.
Alternative/Exception Path(s):	N/A
Post-Conditions:	Changes in the settings will be reflected
Business Rules:	As the actor, I want to manage basic admin settings
Wireframe:	Refer Section 1.4, Figure 13 – Figure 14 ECA wireframe booklet
Data Table:	Refer Section 1.4.2, ECA wireframe booklet
Supporting Documents:	1. URS ECA 2. ECA Wireframe Booklet
Usage Frequency:	Often

Criticality	High		
Dependencies	N/A		
Change History	Author	Version	Explanation
	Zainoor Asykin Zainol Abidin	1.3	Added portal settings in basic path

<center>Table 9 - CORE_ECA-05 Manage Basic Administrator Settings</center>

VI.   CORE_ECA-06 Manage Advanced Administrator Settings

Use Case Component:	Description
Use Case ID:	CORE_ECA-06
Use Case Name:	Manage Advanced Administrator Settings
Actor:	Super Admin
Pre-Condition:	The actor must exist in the CORE. The actor must have permission to manage advanced admin settings
Basic Path:	1. Actor login the system. 2. System will check the actor permission. 3. Actor navigate to CORE Menu. 4. Actor click on the Admin Settings button. 5. Actor click on Advanced Settings tab. 6. Actor can make changes on date auto-match regexes, email templates (Sign Up, Create User/Update Profile, Forgot Password, Project Creation/Project Update, Backup, Document Upload and New Page/Post Created) and email configuration

	(admin email, Smtp Host, Smtp Enable SSL, UseDefaultCredential, Smtp Port, Smtp Username and Smtp Password) by clicking on the Edit button (refer CORE_EM-01).
Alternative/Exception Path(s):	N/A
Post-Conditions:	Changes will be reflected in the advanced admin settings
Business Rules:	As the actor, I want to manage advanced admin settings
Wireframe:	Refer Section 1.4, Figure 15 ECA wireframe booklet
Data Table:	Refer Section 1.4.2, ECA wireframe booklet
Supporting Documents:	1. URS ECA 2. ECA Wireframe Booklet 3. SRS EM
Usage Frequency:	Often
Criticality	High
Dependencies	N/A

Change History	Author	Version	Explanation
	Zainoor Asykin Zainol Abidin	1.2	Added missing use cases and amended based on the feedbacks

Table 10 - CORE_ECA-06 Manage Advanced Administrator Settings

VII.   CORE_ECA-07 Manage Backup Management

Use Case Component:	Description
Use Case ID:	CORE_ECA-07
Use Case Name:	Manage Backup Management
Actor:	Super Admin
Pre-Condition:	The actor must exist in the CORE. The actor must have permission to manage backup.

Basic Path:	1. Actor login the system.
	2. System will check the actor permission.
	3. Actor navigates to Settings Menu.
	4. Actor click on the Backup Management button.
	Local Backup
	a) Actor click on the Local Backups tab.
	b) Actor view the list of created backups and click on the icon under Action;
	    i. Arrow down icon to download the backup.
	    ii. Bin icon to delete the old backup.
	c) Actor can run a backup by clicking on Create Backup button, choose to run the existing backup profile or add a new profile.
	d) If the actor clicks on Add Backup Profile button, the actor needs to complete the details by inserting profile title and choosing 'Complete Backup' or 'Database Only'.
	e) Actor can tick on the checkbox to notify user on the backup status.
	f) Actor can tick on the checkbox on 'Send to remote destination as part of backup process' and choose the destination (Shared Folder on Intranet or FTP).

g) Actor click on Select Destination button and choose any existing backup profile to run the backup.

h) Overview on the backup will automatically displayed when running the backup.

i) Actor can click on Cancel Backup button to cancel the backup and view status log by clicking on the Status Log tab.

j) In the status log tab, it will display time elapsed and memory message.

k) Actor can click on Status log (.txt) button to download the status log.

Restore Backup

a) Actor click on the Restore Backup tab.

b) Actor view the list of created backups and click on the icon under Action to restore the backup.

Schedules

a) Actor click on the Schedules tab.

b) Actor view the list of schedules backup and click on the icon under Action to delete the schedule.

c) Actor create a new backup schedule by clicking on Add Schedule button.

d) Actor complete the details by entering the Schedule Name, choosing the existing backup profile, the backup interval, date/time of next run and remote backup destination.

e) Actor can choose to tick the checkbox on 'Delete local backup file after remote send success' and 'Enable schedule to run', then click Add Schedule button to create the schedule.

Destination

a) Actor click on the Destination tab.

b) Actor click on Add New under Destination tab.

c) If the actor chooses FTP, the actor needs to fill in the details (destination name, server address, username, password, remote path, migration URL

	and archive limit), then click Save Setting button to save the destination. d) If the actor chooses Local Network, the actor needs to fill in the details (destination name, local file path, migration URL and archive limit), then click Save Setting button to save the destination. e) The new destination will be saved in the tab under destination.
Alternative Path(s):	N/A
Exception Path(s):	N/A
Post-Conditions:	1. If the backup failed, all errors will be logged with traces and important logging. 2. Actor have the ability to clean the log file, to retrieve and to download. 3. Local backups will automatically be deleted if reached archive limit numbers set in the Destination or the actor set to delete local backup file after remote send success.
Business Rules:	As the actor, I want to manage Backup Management
Wireframe:	Refer Section 1.5, Figure 16 – Figure 25 ECA wireframe booklet
Data Table:	Section 1.5.2, ECA wireframe booklet
Supporting Documents:	1. URS ECA 2. ECA Wireframe Booklet
Usage Frequency:	Often
Criticality	High
Dependencies	N/A

Change History	Author	Version	Explanation

	Zainoor Asykin Zainol Abidin	1.4	Update basic path and post condition for 'Local Backups' and 'Schedules'

Table 11 - CORE_ECA-07 Manage Backup Management

VIII.   CORE_ECA-08 Manage Search Engine Index Management

Use Case Component:	Description
Use Case ID:	CORE_ECA-08
Use Case Name:	Manage Search Engine Index Management
Actor:	Super Admin
Pre-Condition:	The actor must exist in the CORE. The actor must have permission to manage search engine index.
Basic Path:	1. Actor login the system. 2. System will check the actor permission. 3. Actor navigates to Settings Menu. 4. Actor clicks on the Search Engine Index button **Indexing** a) Actor click on the Indexing tab. b) Actor have the option to build index. c) Actor click on Save Options button to save the changes made. **Searching** a) Actor click on Searching tab. b) Actor have the option to set default operator search using AND or OR, enable fallback to OR, option for default order (relevance, date, etc), manage weightage (content, title, category, subcategory, custom content and description (excerpt), boost exact matches and category restriction.

	c) Actor click on Save Options button to save the changes made. **Synonyms** a) Actor click on Synonyms tab. b) Actor have the ability to add synonym word for a better search result. c) Actor click on Save Changes to save the changes made. **Stopword** a) Actor click on Stopword tab. b) Actor have the option to manage stopwords (add stopwords and exportable list of stopword), manage content stopwords (add content stopword and exportable list of content stopwords). c) Actor click on Save Changes to save the changes made.
Alternative/Exception Path(s):	N/A
Post-Conditions:	Changes in the indexing, searching, synonyms and stopword tabs will be reflected in search engine settings.
Business Rules:	As the actor, I want to manage Search Engine Index
Wireframe:	Refer Section 1.6, Figure 26 – Figure 29 ECA wireframe booklet
Data Table:	Refer Section 1.6.2, ECA wireframe booklet
Supporting Documents:	1. URS ECA 2. ECA Wireframe Booklet
Usage Frequency:	Often
Criticality	High
Dependencies	N/A

Change History	Author	Version	Explanation
	Zainoor Asykin Zainol Abidin	1.3	Removed unnecessary things in indexing and searching

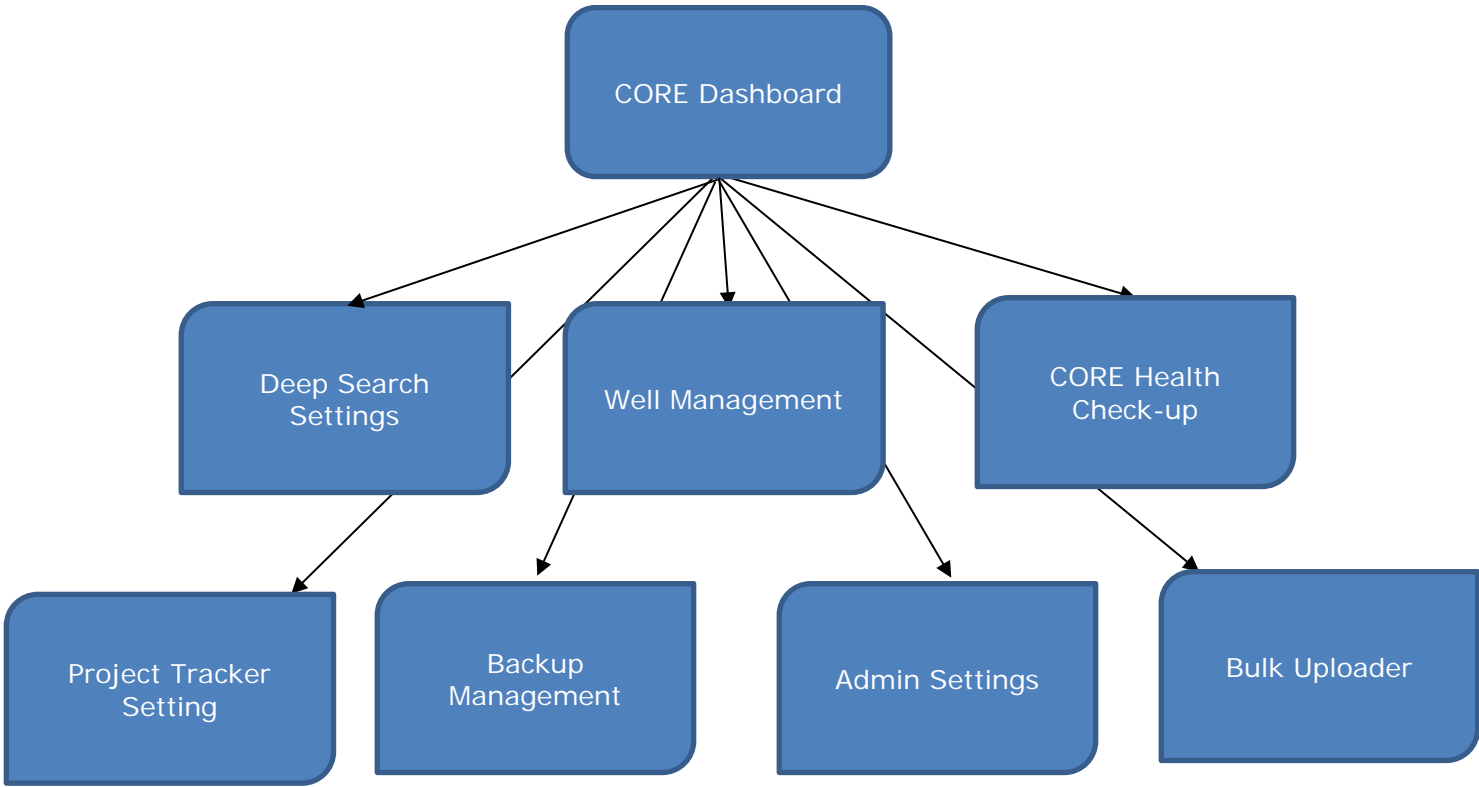Table 12 - CORE_ECA-08 Manage Search Engine Index Management

IX.   CORE_ECA-09 Manage Project Tracker Setting

Use Case Component:	Description
Use Case ID:	CORE_ECA-09
Use Case Name:	Manage Project Tracker Setting
Actor:	Super Admin, Project Manager
Pre-Condition:	The actor must exist in the CORE. The actor must have permission to manage project tracker setting.
Basic Path:	1. Actor login the system. 2. System will check the actor permission. 3. Actor navigate to CORE Menu. 4. Actor click on the Project Tracker Setting button. 5. Actor click on the drop-down menu to select which project to track. 6. Actor click on the Next button, and the project details (milestone, date, days to ML, existing (%) and proposed (%)) will be displayed. 7. Actor can click on Update button to update the project settings.
Alternative/Exception Path(s):	N/A
Post-Conditions:	The project details will be displayed.

Business Rules:	As the actor, I want to manage Project Tracker Setting		
Wireframe:	Refer Section 1.7, Figure 30 ECA wireframe booklet		
Data Table:	Refer Section 1.7.2 ECA wireframe booklet		
Supporting Documents:	1. URS ECA 2. ECA Wireframe Booklet		
Usage Frequency:	Often		
Criticality	High		
Dependencies	N/A		
Change History	Author	Version	Explanation
	Zainoor Asykin Zainol Abidin	1.2	Added missing use cases and amended based on the feedbacks

Table 13 - CORE_ECA-09 Manage Project Tracker Setting

4.4 Site Structure

```
                        CORE Dashboard

   Deep Search          Well Management        CORE Health
   Settings                                    Check-up

Project Tracker    Backup            Admin Settings    Bulk Uploader
Setting            Management
```

4.5 Integration and Dependencies

ID	BRS ID	Description
INT-1	N/A	N/A

Table 14 - Integration and Dependencies

4.6 Access Matrix

Capability	ROLES			
	Member	Project Manager	Power User	Super Admin
Manage Deep Search Setting				✔
Well Management				✔
Manage CORE portal health check-up				✔
Manage Bulk Load Workflow			✔	✔
Manage Basic Administrator Settings				✔
Manage Advanced Administrator Settings				✔
Manage Backup Management				✔
Manage Search Engine Index Management				✔
Manage Project Tracker Setting		✔		✔

Table 15 - Access Matrix

4.7 Other Functional Requirements

Below are the functional requirements that are not able to be depicted by use case diagram.

ID	BRS ID	Description
FUNC-1	1.0	The system will provide filtering function for the user (whenever required).

Table 16 - Other Functional Requirements

5 Non-functional requirements

Non-functional requirement basically describes how the system works.

5.1 Performance requirements

ID	BRS ID	Description
PERF-1	1.0	Response time – The maximum response time to load will be 5 to 10 seconds

Table 17 - Performance Requirements

5.2 Security Requirements

ID	BRS ID	Description
SEC-1	1.0	To login the system only authorized user with registered ID can login into the system
SEC-2	2.0	The ability to view certain dashboard, menus, pages and do certain actions are dependent to the role and department given by the Super Admin

Table 18 - Security Requirements

5.3 Accessibility Requirements

ID	BRS ID	Description
DES-1	1.0	The accessibility would be depending on user, role, capabilities and department defined by the Super Admin

Table 19 - Accessibility Requirements