System Analysis and Design Course Project

Group 5

Group Members:

Curtis Mintah (040994474)

Almaz Albanov (040985613)

Austin Oneil (040919419)

Kathleen Charbonneau

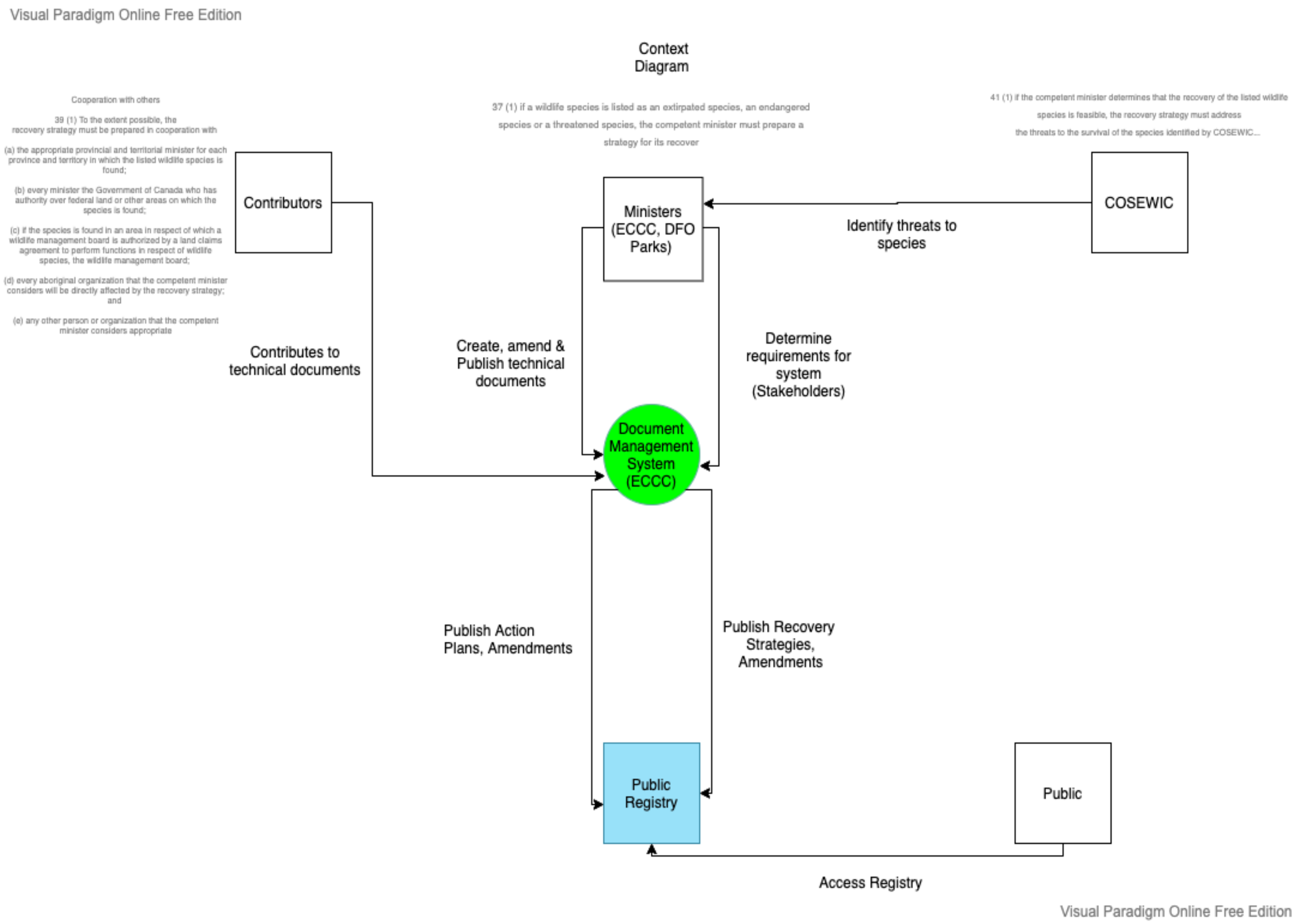
April 16th, 2021

Section 1: Project Plan

Project Description

The project that we have been tasked with is to create a document management system for Environment and Climate Change Canada (ECCC). This is to ensure the SARA act is enforced. Several deliverables will be completed to ensure the Document Management System is created in its entirety (e.g. a Gantt chart to lay out the steps required to complete the project). To reiterate, the system “manages recovery strategies and action plans for endangered species at risk in Canada” [1]. The requirements for the system will be displayed in the deliverables below. These diagrams and text will determine how our team will complete the project, and how several individuals will use the system (e.g. Competent Ministers). This system will ultimately process the necessary documents and publish them to the public registry.

Project scope


Visual Paradigm Online Free Edition
Context Diagram
Contributors
Ministers (ECCC, DFO Parks)
COSEWIC
Identify threats to species
Contributes to technical documents
Create, amend & Publish technical documents
Determine requirements for system (Stakeholders)
Document Management System (ECCC)
Publish Action Plans, Amendments
Publish Recovery Strategies, Amendments
Public Registry
Public
Access Registry
Visual Paradigm Online Free Edition

Cooperation with others

39 (1) To the extent possible, the recovery strategy must be prepared in cooperation with

(a) the appropriate provincial and territorial minister for each province and territory in which the listed wildlife species is found;

(b) every minister the Government of Canada who has authority over federal land or other areas on which the species is found;

(c) if the species is found in an area in respect of which a wildlife management board is authorized by a land claims agreement to perform functions in respect of wildlife species, the wildlife management board;

(d) every aboriginal organization that the competent minister considers will be directly affected by the recovery strategy; and

(e) any other person or organization that the competent minister considers appropriate

37 (1) if a wildlife species is listed as an extirpated species, an endangered species or a threatened species, the competent minister must prepare a strategy for its recover

41 (1) if the competent minister determines that the recovery of the listed wildlife species is feasible, the recovery strategy must address the threats to the survival of the species identified by COSEWIC...

Figure 1: Context Diagram

Draft of the Process model

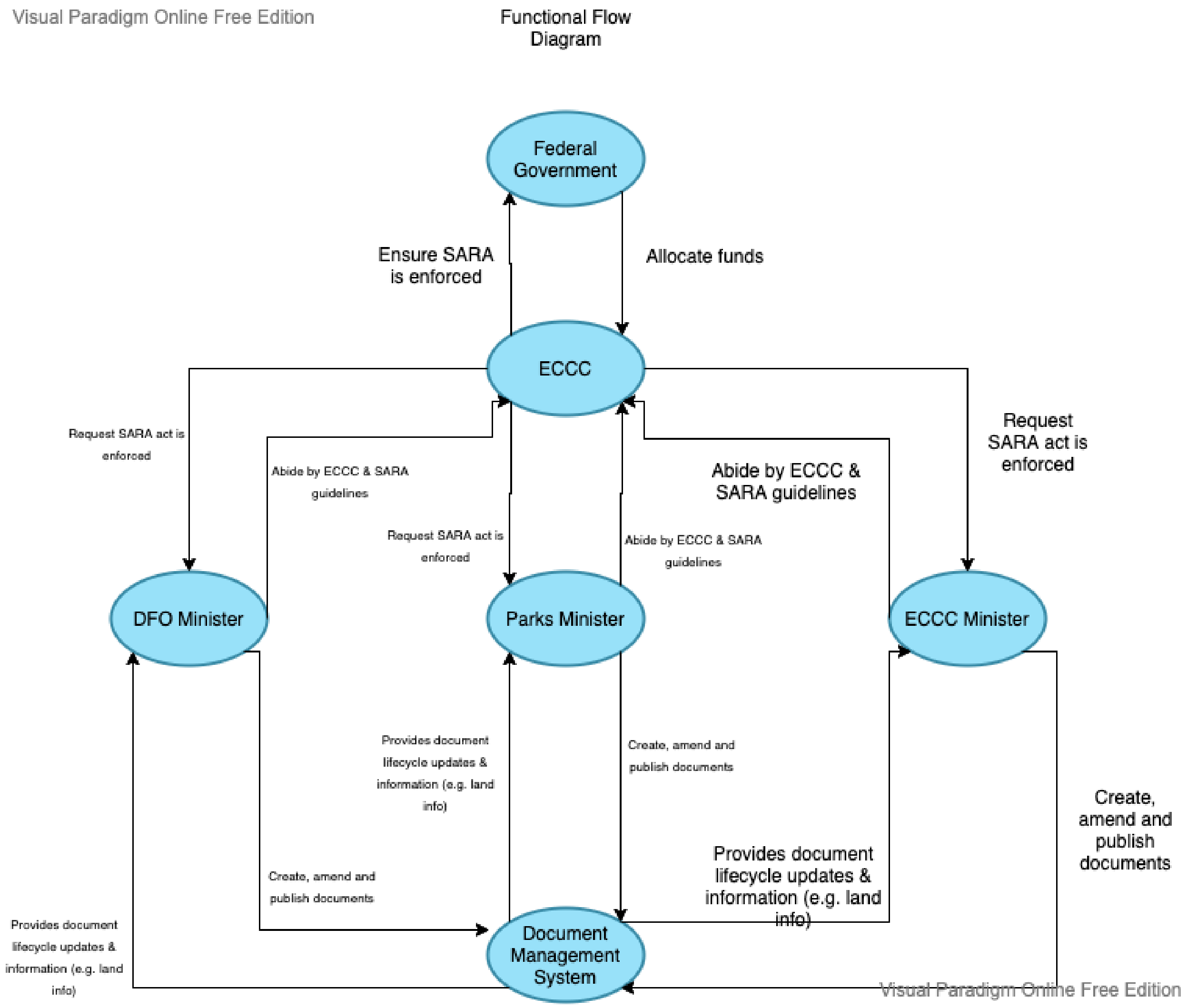

Visual Paradigm Online Free Edition
Functional Flow Diagram
Federal Government
Ensure SARA is enforced
Allocate funds
ECCC
Request SARA act is enforced
Abide by ECCC & SARA guidelines
Request SARA act is enforced
Abide by ECCC & SARA guidelines
Request SARA act is enforced
Abide by ECCC & SARA guidelines
DFO Minister
Parks Minister
ECCC Minister
Provides document lifecycle updates & information (e.g. land info)
Create, amend and publish documents
Provides document lifecycle updates & information (e.g. land info)
Create, amend and publish documents
Create, amend and publish documents
Provides document lifecycle updates & information (e.g. land info)
Document Management System
Visual Paradigm Online Free Edition

Figure 2: Functional Flow Diagram (Process Model)

Draft of the requirements plan

Requirements gathering process

The requirements will be gathered from the documentation provided to us by our client (ECCC). We will also consult with the stakeholders, contributors (e.g. Provincial governments), etc. This allows us to adapt to the ever changing requirements that will emerge as the project continues. We will also speak to competent ministers to ensure their needs are being met for the document management system.

Roles and responsibilities

All current members working on the document management system are considered project managers. They will ensure all requirements have been met. All members will also serve as lead analysts, ensuring the integrity of the requirements are kept as the project continues through the life cycle (e.g. SARA act is enforced).

Tools

The main tools used will be Visual Studio, Microsoft Word, Microsoft Excel, and more. There will be little to no use of automated tools.

Requirements traceability

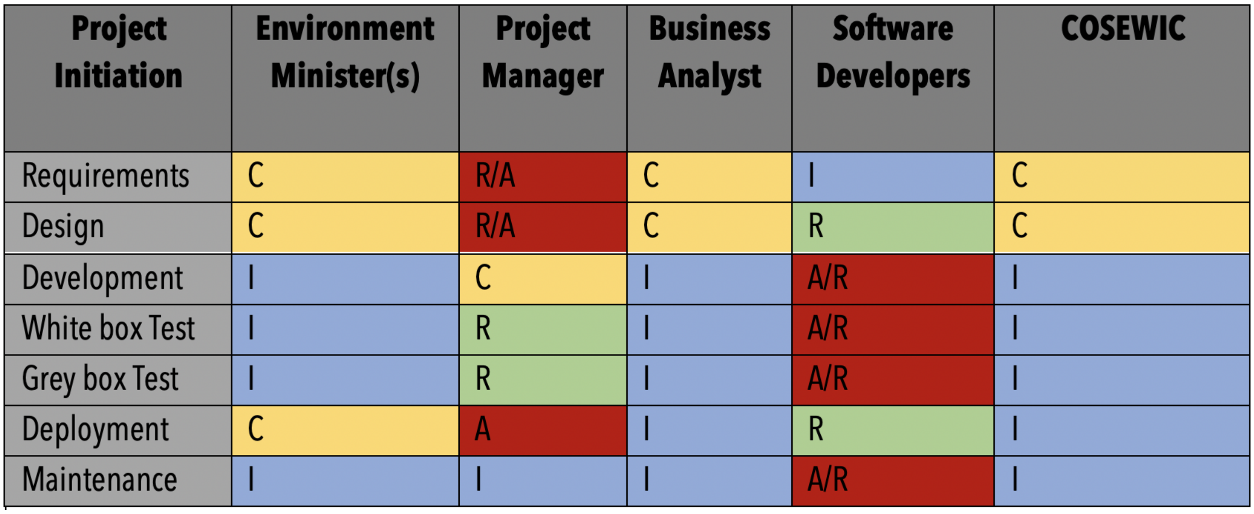
The requirements of the project are going to be tracked using a traceability matrix, mainly. The gantt chart will also aid in this process. The traceability matrix will include all stakeholder requirements. The gantt chart focuses on a simple, week-by-week approach.

It must be noted however that we expect delays in the deliverables. A maximum of 3 weeks will be provided at the end of the last week shown in the gantt chart to ensure the work is submitted properly. Using these diagrams, the deliverables will be provided in the allotted time.

Change Control

In order to ensure the document management system meets stakeholder requirements, we will actively accept changes to requirements, provided they are reasonable and within the scope of our timeline. The changes will be approved and processed by all project managers. The change control will work as follows: We will hold weekly digital meetings with stakeholders, accessing the requirements. They will make changes. We will agree or disagree to process the changes via a messaging service (e.g. Microsoft Teams). The changes will be added to our traceability matrix.

Stakeholder Management plan (RACI matrix) (updated)



Project Initiation	Environment Minister(s)	Project Manager	Business Analyst	Software Developers	COSEWIC
Requirements	C	R/A	C	I	C
Design	C	R/A	C	R	C
Development	I	C	I	A/R	I
White box Test	I	R	I	A/R	I
Grey box Test	I	R	I	A/R	I
Deployment	C	A	I	R	I
Maintenance	I	I	I	A/R	I

Figure 3: RACI matrix

Part 2: Work Breakdown Structure (WBS)

WBS (Updated)

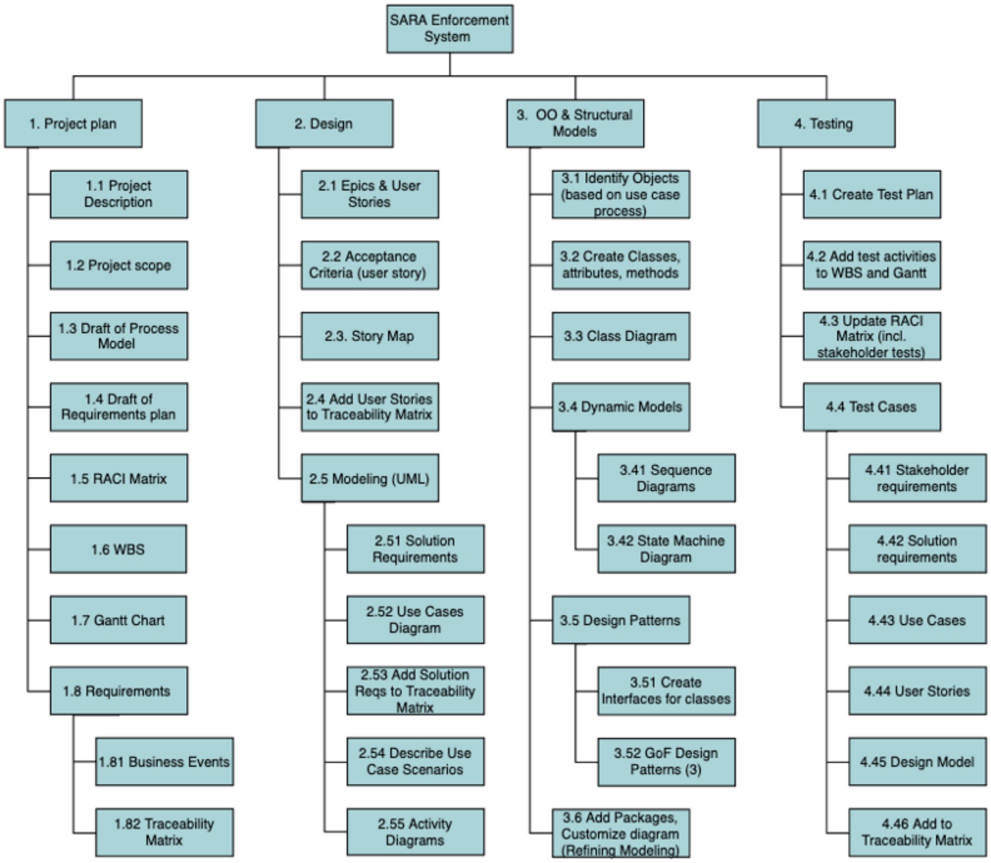

SARA Enforcement System
1. Project plan
1.1 Project Description
1.2 Project scope
1.3 Draft of Process Model
1.4 Draft of Requirements plan
1.5 RACI Matrix
1.6 WBS
1.7 Gantt Chart
1.8 Requirements
1.81 Business Events
1.82 Traceability Matrix
2. Design
2.1 Epics & User Stories
2.2 Acceptance Criteria (user story)
2.3. Story Map
2.4 Add User Stories to Traceability Matrix
2.5 Modeling (UML)
2.51 Solution Requirements
2.52 Use Cases Diagram
2.53 Add Solution Reqs to Traceability Matrix
2.54 Describe Use Case Scenarios
2.55 Activity Diagrams
3. OO & Structural Models
3.1 Identify Objects (based on use case process)
3.2 Create Classes, attributes, methods
3.3 Class Diagram
3.4 Dynamic Models
3.41 Sequence Diagrams
3.42 State Machine Diagram
3.5 Design Patterns
3.51 Create Interfaces for classes
3.52 GoF Design Patterns (3)
3.6 Add Packages, Customize diagram (Refining Modeling)
4. Testing
4.1 Create Test Plan
4.2 Add test activities to WBS and Gantt
4.3 Update RACI Matrix (incl. stakeholder tests)
4.4 Test Cases
4.41 Stakeholder requirements
4.42 Solution requirements
4.43 Use Cases
4.44 User Stories
4.45 Design Model
4.46 Add to Traceability Matrix

Figure 4: Work Breakdown Structure

Gantt Chart (Updated)

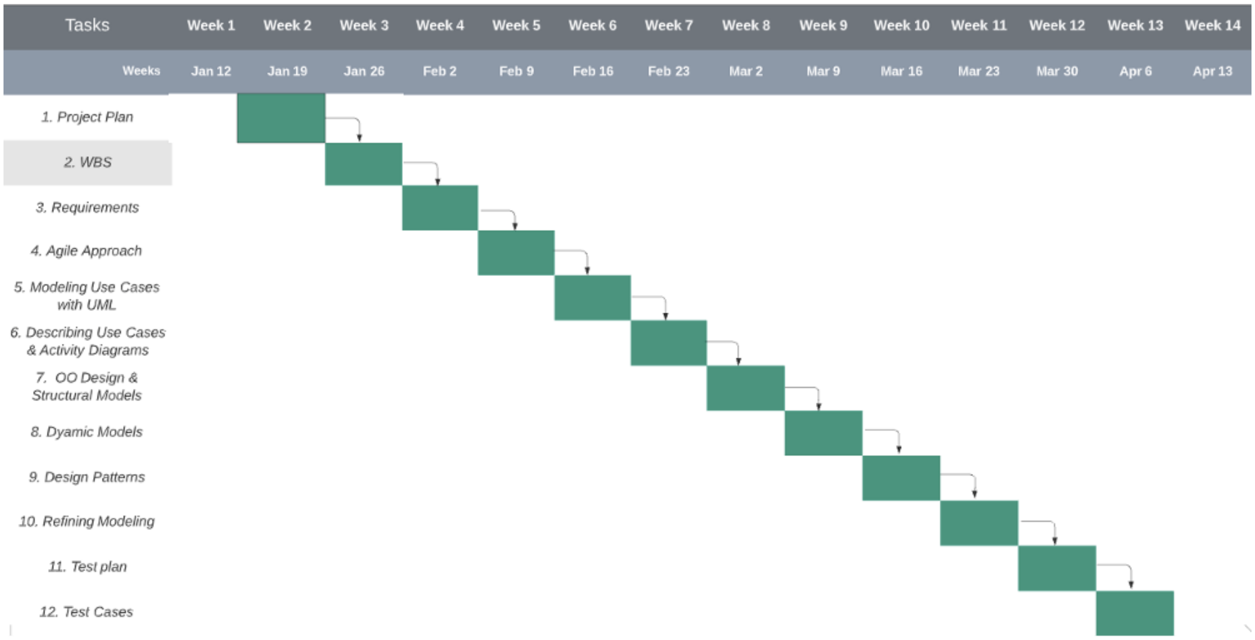


Figure 5: Gantt Chart. Please note: The main testing will occur at week 12. The stakeholder requirements, solution requirements, use cases, user stores and design model will be tested. Monthly testing (i.e week 4, 8 and 12) will also occur, with week 12 being the main week to fulfill requirements.

Section 3: Requirements

Identify Business Events (based on context diagram)

The following business events were procured from analyzing our context diagram.

- Publish Action Plans and Recovery Strategies to Public Registry
- The Public accesses Registry
- Create amend, track and publish technical documents
- COSEWIC identifies threats to species; consults with Ministers (this affects the quality of document submitted)
- Contributors consult with Ministers (this affects the quality of document submitted)

Stakeholder Requirements:

Stakeholder requirements are important requirements that determine how the project is completed. From the perspective of the system analyst, to the project manager, and finally to creating the document management system, we must consider what the stakeholder needs.

Requirement Number	StakeHolder Requirements
001	The competent ministers (ECCC, DFO, and Parks Canada) shall be able to create, amend, track and publish Recovery Strategies and Action Plans to the Document Management System.
002	The competent ministers (ECCC, DFO, and Parks Canada) shall be able to communicate with COSEWIC members within the system to ensure technical document quality
003	The competent ministers (ECCC, DFO, and Parks Canada) shall be able to communicate with Contributors within the

	system to ensure technical document quality
004	The competent ministers (ECCC, DFO, and Parks Canada) shall be able to access information to aid them in document creation (e.g. Contact lists, land information, species location, etc.)

Refining High Level business approach (process model)

After requirements have been specified, it is time to update the process model to a higher level. A cross-functional flowchart was create to demonstrate this.

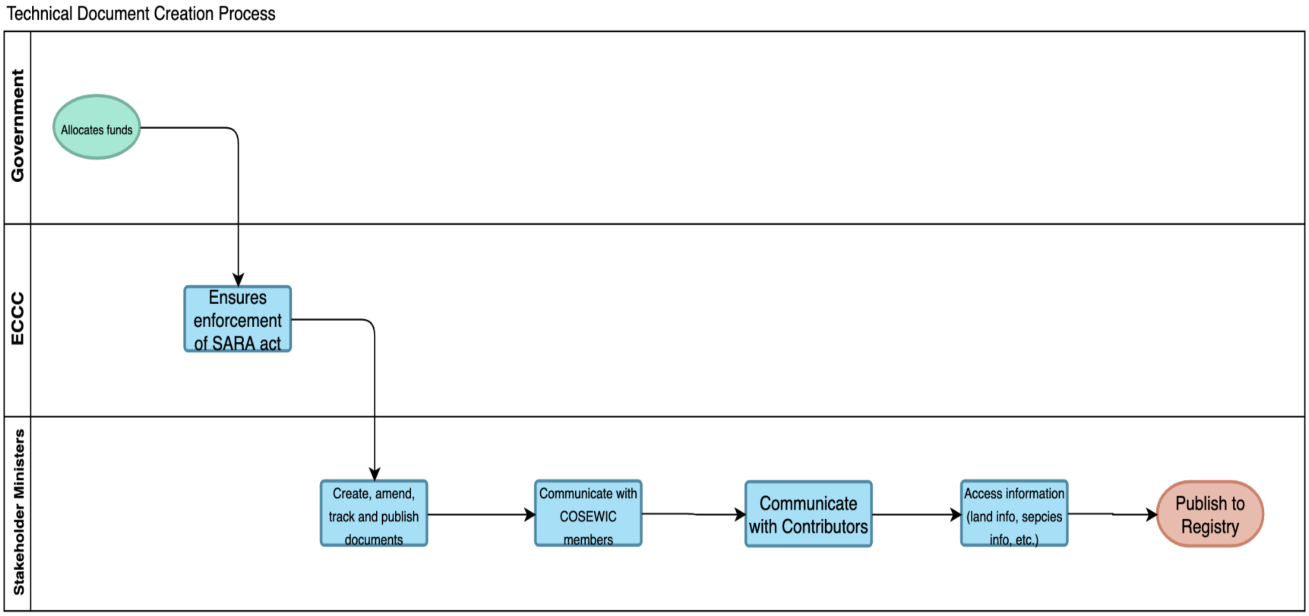

Technical Document Creation Process
Government
ECCC
Stakeholder Ministers
Allocates funds
Ensures enforcement of SARA act
Create, amend, track and publish documents
Communicate with COSEWIC members
Communicate with Contributors
Access information (land info, sepcies info, etc.)
Publish to Registry

Figure 6: Cross-functional flowchart

Traceability Matrix

The traceability matrix is provided in an excel file submitted with this assignment.

Section 4: Agile Approach

Epics and User stories

Epics:

As a stakeholder (ECCC, DFO, Parks Canada) I want to be able to create, amend, track and publish documents to a document management system. It will fulfill the SARA act and allow the public to access the technical documents through the public registry.

User stories:

User Story-001: As a stakeholder, I want to be able to create, amend, track and publish technical documents to the public registry so that the public (and other individuals) can access them.

User Story-002: As a stakeholder, I want to be able to communicate with COSEWIC members so that the SARA act is enforced.

User Story-003: As a stakeholder, I want to be able to access the necessary information required (Contact lists, land information, species location) in the document management system so that the work process is efficient and precise.

User Story-004: As a stakeholder, I want to be able to communicate with contributors so that the technical document quality is ensured.

User Story-005: As a stakeholder, I want to be able to receive suggestions to improve technical documents from the ECCC minister and other parties in the form of amendments so that the SARA act will be enforced.

User Story-006: As a stakeholder, I want to be able to give suggestions to the project manager and developers on the document management system so that the quality of the system is upheld.

Acceptance Criteria

Acceptance criteria for User Story-001:

- The document management system must be able to create, amend, track and publish documents.
- The public must be able to access the public registry via website.
- ECCC minister must be able to communicate with individuals who are creating technical documents. After confirmation, the document can be published.

Acceptance criteria for User Story-002:

- COSEWIC members must have access to the technical documents and have the ability to send suggestions to enforce amendments.
- COSEWIC members must be able to track progress of documents and aid in the approval of documents
- Technical document writers must have access to the SARA act within the document management system.
-

Acceptance criteria for User Story-003:

- Contact lists must be integrated in the document management system.
- Land information must be integrated in the system.
- Water information must be integrated in the system (e.g. water temperature)
- Park information must be integrated in the system (e.g. Park size)
- Species information must be integrated in the system.

Acceptance criteria for User Story-004:

- Contributors (e.g. indigenous land owners) must be able to access technical documents (provided they are given permission).
- Contributors must be able to affect the amendment process of the technical document.

Acceptance criteria for User Story-005:

- ECCC minister must be able to give permission to workers to publish documents to the registry.
- ECCC minister must be able to publish documents to the registry.

- ECCC minister must be able to communicate with other stakeholders to ensure the technical documents are worth publishing
- ECCC minister must be in contact with COSEWIC members to ensure SARA act is enforced.

Acceptance criteria for User Story-006:

- Stakeholder (ECCC, DFO, Parks) must be able to communicate and offer suggestions to project manager
- Stakeholder (ECCC, DFO, Parks) must be able to communicate and offer suggestions to system developers.
- The document management system must have a forum section in which stakeholders can submit complaints or issues to developers & project manager.

Story Map

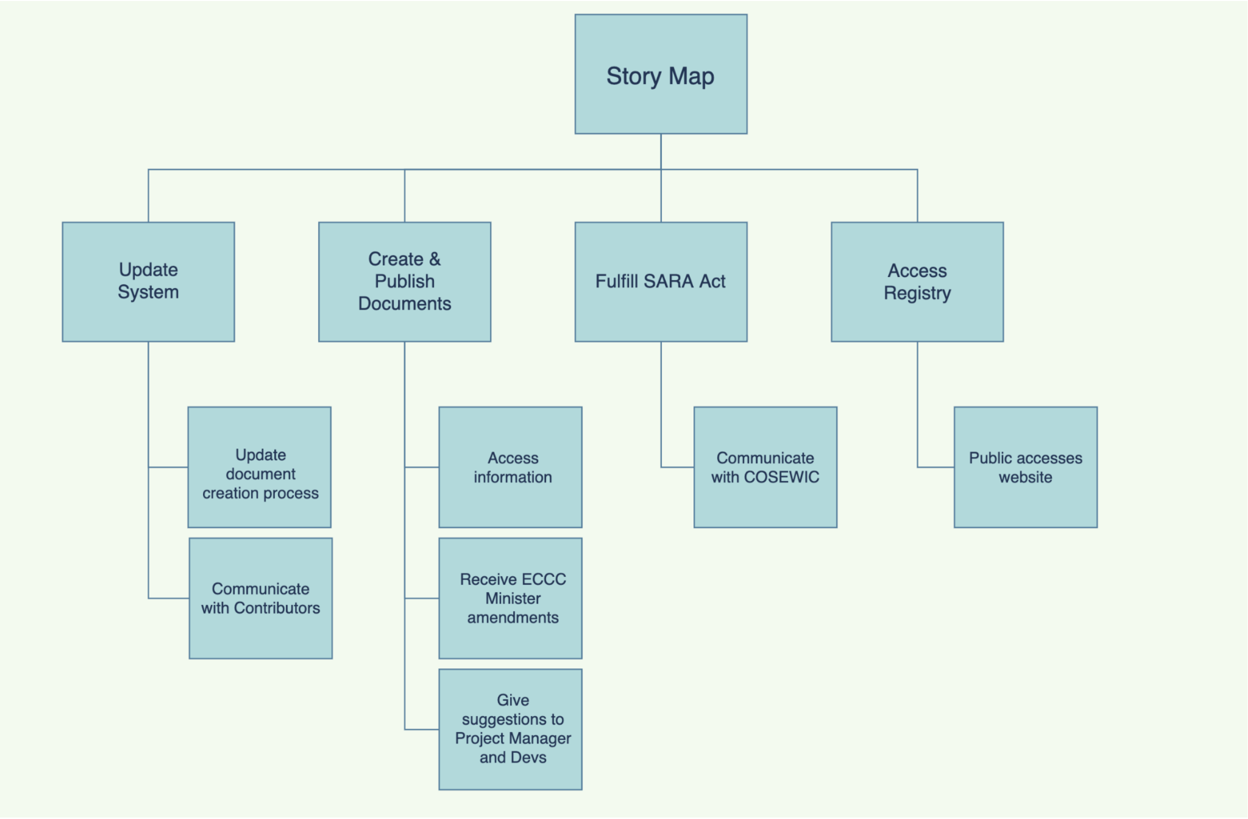

Story Map
Update System
Create & Publish Documents
Fulfill SARA Act
Access Registry
Update document creation process
Communicate with Contributors
Access information
Receive ECCC Minister amendments
Give suggestions to Project Manager and Devs
Communicate with COSEWIC
Public accesses website

Figure 7: Story map create from epics and user stories.

Section 5: Modeling Use Cases with UML

Identify Solution Requirements

Functional

Solution ID	Solution Requirements (Functional)
SR-001	The system must be able to create technical documents.
SR-002	The system must be able to amend technical documents.
SR-003	The system must be able to track technical documents.
SR-004	The system must be able to publish technical documents.
SR-005	The system must be able to import documents
SR-006	The system must be able to export documents
SR-007	The system must be connected to the public registry.
SR-008	The system must have a communication interface where COSEWIC workers can communicate with technical document writers.
SR-009	The system must contain a database with the necessary information (e.g. species data, land data)
SR-010	The system must have a front end page for the information in the database
SR-011	The system must have a communication interface where Contributors can communicate with technical document writers.

Non-Functional:

Solution ID	Solution Requirements (Non-Functional)
SRN-001	The system must have a simple, interactive UI.
SRN-002	The system must have a well-featured writing tool integrated within the system.
SRN-003	The system must have icons that indicate the status of the technical document (e.g. green circle = submitted to registry, yellow circle= amended)
SRN-004	The system must be available from 4am to 10PM Eastern time (considers other time zones adequately).
SRN-005	The system must follow federal government accessibility standards
SRN-006	The system must have separated data sets for each main component of Environment Canada (ECCC, DFO, Parks).

Use Case Diagram

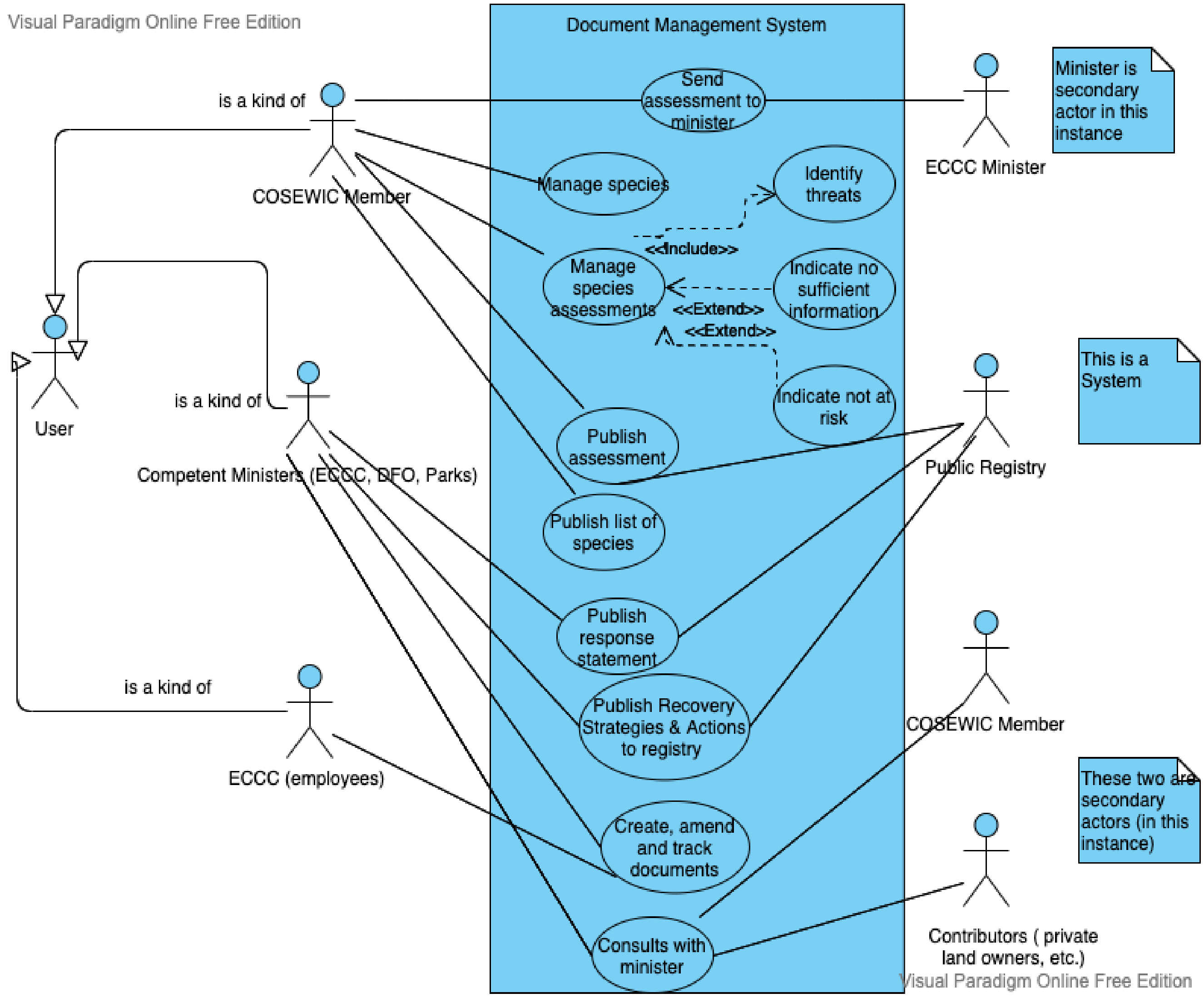

Visual Paradigm Online Free Edition
Document Management System
is a kind of
COSEWIC Member
Send assessment to minister
ECCC Minister
Minister is secondary actor in this instance
Manage species
Identify threats
<<Include>>
Manage species assessments
<<Extend>>
<<Extend>>
Indicate no sufficient information
Indicate not at risk
User
is a kind of
Competent Ministers (ECCC, DFO, Parks)
Publish assessment
Public Registry
This is a System
Publish list of species
Publish response statement
is a kind of
ECCC (employees)
Publish Recovery Strategies & Actions to registry
COSEWIC Member
These two are secondary actors (in this instance)
Create, amend and track documents
Consults with minister
Contributors (private land owners, etc.)
Visual Paradigm Online Free Edition

Figure 8: Use case diagram.

Section 6: Describing Use Cases and representing using Activity Diagrams

Describe Use Case scenarios

Use Case 001:

Title: Send assessment to minister

Primary Actor: COSEWIC Member

Brief Description: This use case describes how the COSEWIC member can send an assessment to the minister.

Trigger: COSEWIC member is consulted for appropriate data. He/she must send this to the minister to ensure the SARA act is enforced (i.e. the technical document meets Government guidelines)

Relationships:
Association*:* COSEWIC member to ECCC Minister
Include: None
Extend: None
Generalization: None

Pre-conditions: Assessment must be created

Post-conditions: Assessment is reviewed my minister & technical document is amended.

Main flow:

1. COSEWIC member acquires the necessary information to create the assessment from scientists, etc.
2. COSEWIC member creates the assessment document
3. COSEWIC member uploads the document onto the document management system
4. Double check if the assessment follows SARA act requirements
5. Send assessment to minister.

Use Case 002:

Title: Manage Species

Primary Actor: COSEWIC Member

Brief Description: This use case describes how the COSEWIC member can manage species in the system.

Trigger: COSEWIC member is given a set of data for species. It needs to be organized such that Ministers can confirm the status of a species (Endangered, Extirpated, Threatened) and acquire other information. Member must ensure the species data is managed and accessible.

Relationships:
Association*:* COSEWIC member
Include: None
Extend: None
Generalization:None

Pre-conditions: The species must be inputted into the database within the system. The COSEWIC member must also have reference to any other Government tools that will aid him manage species within the current document management system.

Post-conditions: The users of the system (e.g. ministers) will have access to a database of species that were managed by COSEWIC member. The COSEWIC member will have to continually update this process as conditions change for a species. This will also update documents.

Main flow:

1. COSEWIC member determines which species need to be managed in the system.
2. If there are any anomalies in the system, the COSEWIC member updates the species data in accordance to guidelines.
3. Species is updated and accessible for users of the system.

Use Case 003:

Title: Manage Species assessments

Primary Actor: COSEWIC Member

Brief Description: This use case describes how the COSEWIC member can manage species accessments in the system.

Trigger: COSEWIC member acquires an assessment of species from a scientist and begins to record the data into the system.

Relationships:
Association*:* COSEWIC member
Include: Identify threats
Extend: Indicate no sufficient information, Indicate not at risk.
Generalization: None

Pre-conditions: Species must be within the system.

Post-conditions: Species assessment is created in the system.

Main flow:

1. COSEWIC member looks for a species to assess and add details to.
2. The system brings up information of species searched
3. COSEWIC member shares their findings from the assessment
4. If the species is at risk
    a. COSEWIC member will add details of risk
    b. Execute use case: identify threats
5. If the species doesn't have enough details
    a. Execute use case: no sufficient information
6. If the species is not at risk
    a. Execuse use case: indicate not at risk
7. COSEWIC member records assessment details in system.

Use Case 004:

Title: Publish assessments

Primary Actor: COSEWIC Member

Brief Description: This use case describes how the COSEWIC member can publish assessments into the public registry.

Trigger: The government, ministers and the public require professional input from organizations such as COSEWIC. The request is issued by upper management and the government.

Relationships:
Association*:* COSEWIC member
Include: None
Extend: None
Generalization: None

Pre-conditions: member must have completed the assessment in full.

Post-conditions: Assessments created by COSEWIC member will be available in the public registry in conjunction to the technical documents published by ministers and ECCC workers. The COSEWIC assessment can be seen as a reference or additional document to the Recovery Strategies and Action Plans.

Main flow:

1. COSEWIC member acquires the necessary information to procure the assessment
2. He/she confirms it follows the SARA act and government guidelines
3. Member publishes assessment to the system
4. System publishes document to the public registry.

Use Case 005:

Title: Publish list of species

Primary Actor: COSEWIC Member

Brief Description: This use case describes how the COSEWIC member can publish lists of species into the public registry.

Trigger: COSEWIC member receives demands from users of the system to create a list of species that is comprehensive and has the correct information on each species.

Relationships:
Association*:* COSEWIC member
Include: None
Extend: None

Generalization: None

Pre-conditions: The document management system has an interactive UI in which the COSEWIC member can publish the list of species.

Post-conditions: The list of species is available on the public registry.

Main flow:

1. A request is made for a list of species by higher ups
2. COSEWIC member procurs data from scientists and government to create the list of species
3. List of species is processed through the system once complete
4. List of species is published to the public registry through the system.

Use Case 006:

Title: Publish response statement

Primary Actor: Competent Ministers (ECCC, DFO, Parks)

Brief Description: This use case describes how the Competent Ministers can publish a response statement to the public registry.

Trigger: The competent minister is creating a technical document. During the process of work for the competent minister, a series of amendments will occur from COSEWIC members, Contributors, etc. to ensure the quality of the technical document is upheld.

Relationships:
Association*:* Competent Ministers (ECCC, DFO, Parks) to Public Registry
Include: None
Extend: None
Generalization: None

Pre-conditions: Minister must be working on a technical document that will be published into the system.

Post-conditions: A response statement will be submitted with any necessary amendments in the technical document. These amendments will account for any changes to strategies and action plans (e.g. a species status has changed from Threatened to Endangered).

Main flow:

1. Competent Minister begins to work on a Recovery Strategy or Action Plan
2. Minister submits document into the system and requests amendments
3. The document is tracked and minister receives amendments in due time.
4. The necessary amendments are done.
5. The response statement is submitted into the system again and is prepared for publishing.
6. The response statement is published in the public registry.

Use Case 007:

Title: Publish Recovery Strategies & Action plans to registry

Primary Actor: Competent Ministers (ECCC, DFO, Parks)

Brief Description: This use case describes how the Competent Ministers can publish recovery strategies & action plans to registry.

Trigger: Ministers are required to enforce the SARA act through technical documents.

Relationships:
Association*:* Competent Ministers (ECCC, DFO, Parks) to Public Registry
Include: None
Extend: None
Generalization: None

Pre-conditions: The new document management system must be functional. Otherwise the old system will be used.

Post-conditions: Recovery strategies & action plans are available on the registry.

Main flow:

1. Government requests SARA act is enforced.
2. ECCC tasks project manager to create a document management system.
3. If the new system is incomplete

    a. Use the old method of submitting documents to the registry
4. If the new system is complete
    a. Continue the main flow
5. Competent Minister abides by all guidelines and creates the technical document
6. Document is inputted into the system
7. Document is uploaded to the public registry.

Use Case 008:

Title: Create, amend and track documents

Primary Actors: Competent Ministers (ECCC, DFO, Parks) & ECCC (employees)

Brief Description: This use case describes how the Competent Ministers & ECCC employees can create, amend and track documents.

Trigger: The government and their subdivisions (ECCC, DFO, Parks, etc.) demand that the SARA act is enforced. Ministers and employees of these subdivisions must comply. One of the ways they can do this is by creating, amending and tracking technical documents.

Relationships:
Association*:* Competent Ministers (ECCC, DFO, Parks), ECCC (employees)
Include: None
Extend: None
Generalization: None

Pre-conditions: The system must have the necessary information in order for the ministers and employees to do the work.

Post-conditions: For the employee: the document will be submitted to higher ups (e.g. ministers, employees directly under ministers) and determined if it should be published. For the ministers: the document will continue to be amended and tracked. If it complies with requirements it will be published to the registry.

Main flow (generalized for both actors):

1. The employee creates the technical document required to enforce the SARA act
2. The necessary information is procured from the system.
3. The first iteration is sent. The document will be amended and tracked as needed.
4. If the employee is a minister
    a. Submit the document to the registry if it meets conditions

5. If the employee is a ECCC employee
    a. Submit the document into the system and await the approval of the minister

<u>Use Case 009:</u>

Title: Consults with minister

Secondary Actor: COSEWIC Member & Contributors
The secondary actor initiates the act in this instance.

Brief Description: This use case describes how the COSEWIC Member & Contributors can consult with the ministers.

Trigger: COSEWIC members and contributors are tasked with reviewing the technical documents the ministers are attempting to get published. In the chain of command within the document management system, it would make sense to have individuals who check the work even without being requested directly. This is where the COSEWIC member and contributor come in.

Relationships:
Association*:* COSEWIC Member & Contributors to Competent Ministers (ECCC, DFO, Parks)
Include: None
Extend: None
Generalization: None

Pre-conditions: The COSEWIC member and contributor must be members within the system.

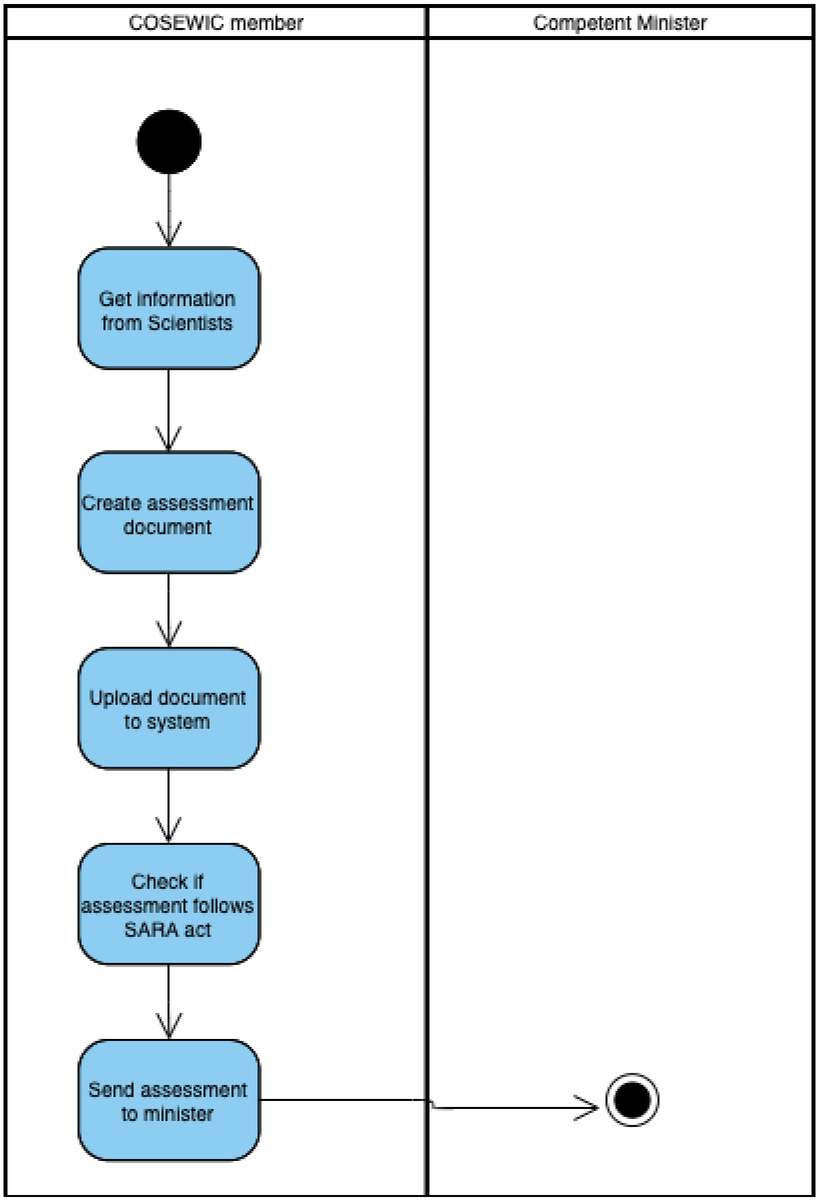
Post-conditions: The minister will receive amendments to their document in the form of conversation and documentation.

Main flow:
1. ECCC Minister submits document in the system and mentions the status of the document.
2. Depending on the status of the document (how far the minister has gotten on the document), the COSEWIC member and contributor will offer their expertise to ensure the document meets guidelines
3. The amendment suggestions is sent back to the minister
4. The minister updates the document

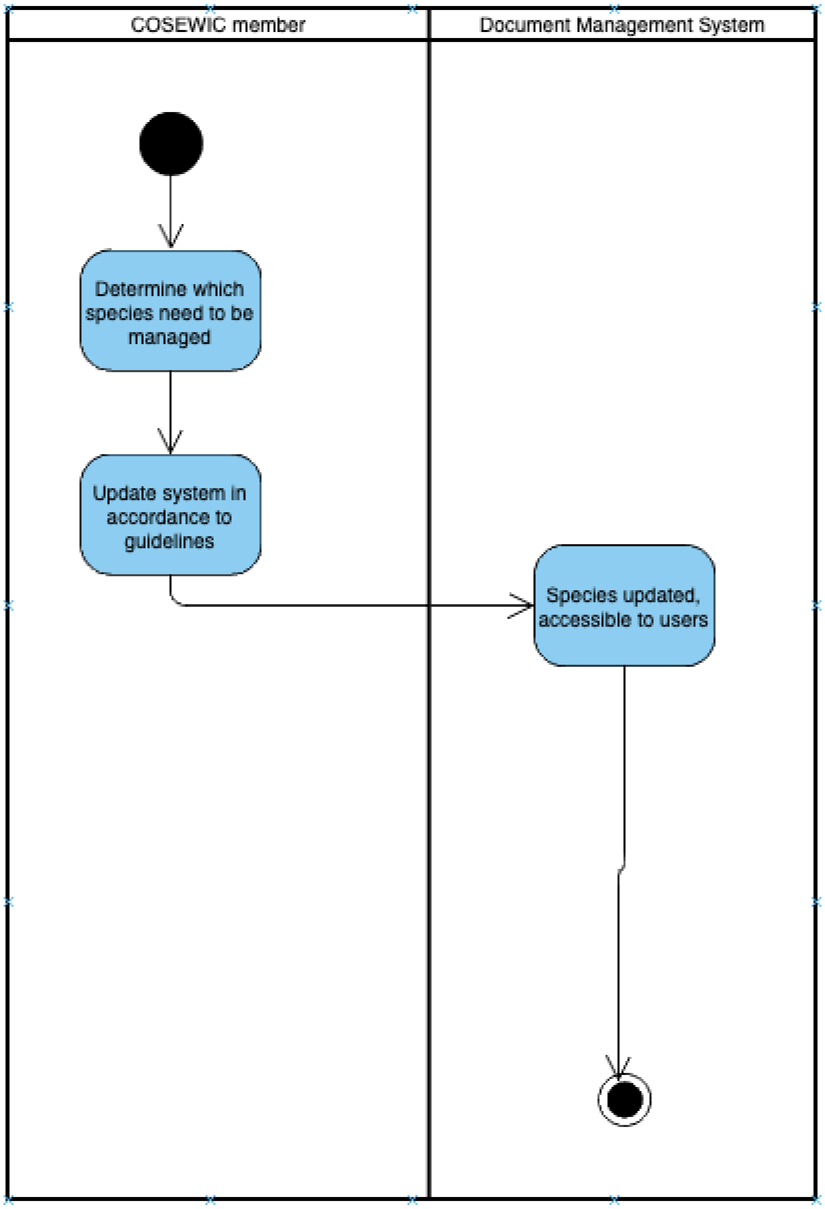
5. The document can be uploaded to the registry.

Activity Diagrams (based on use cases)

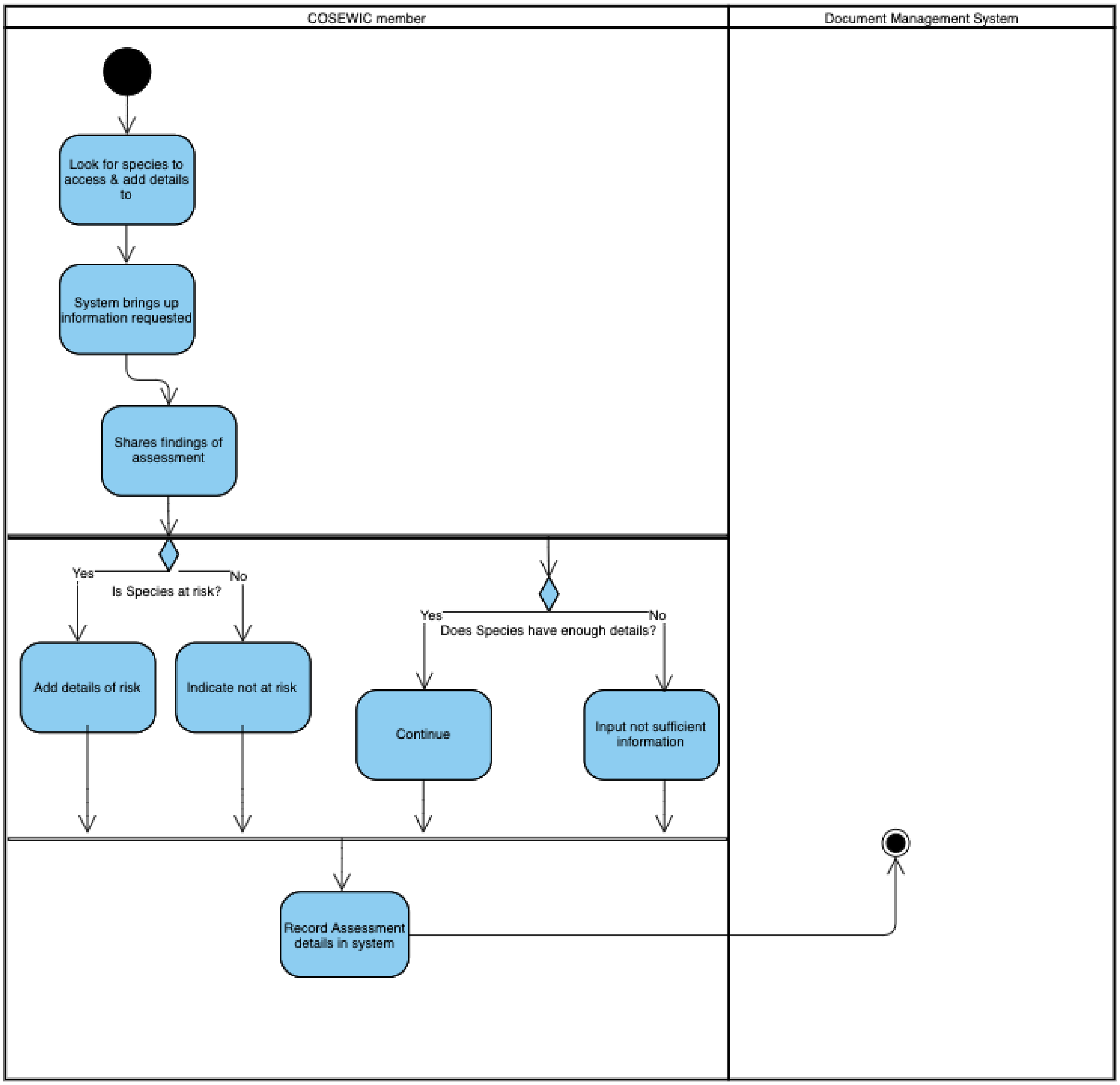
Use Case 001: Send Assessment to Minister


COSEWIC member
Competent Minister
Get information from Scientists
Create assessment document
Upload document to system
Check if assessment follows SARA act
Send assessment to minister

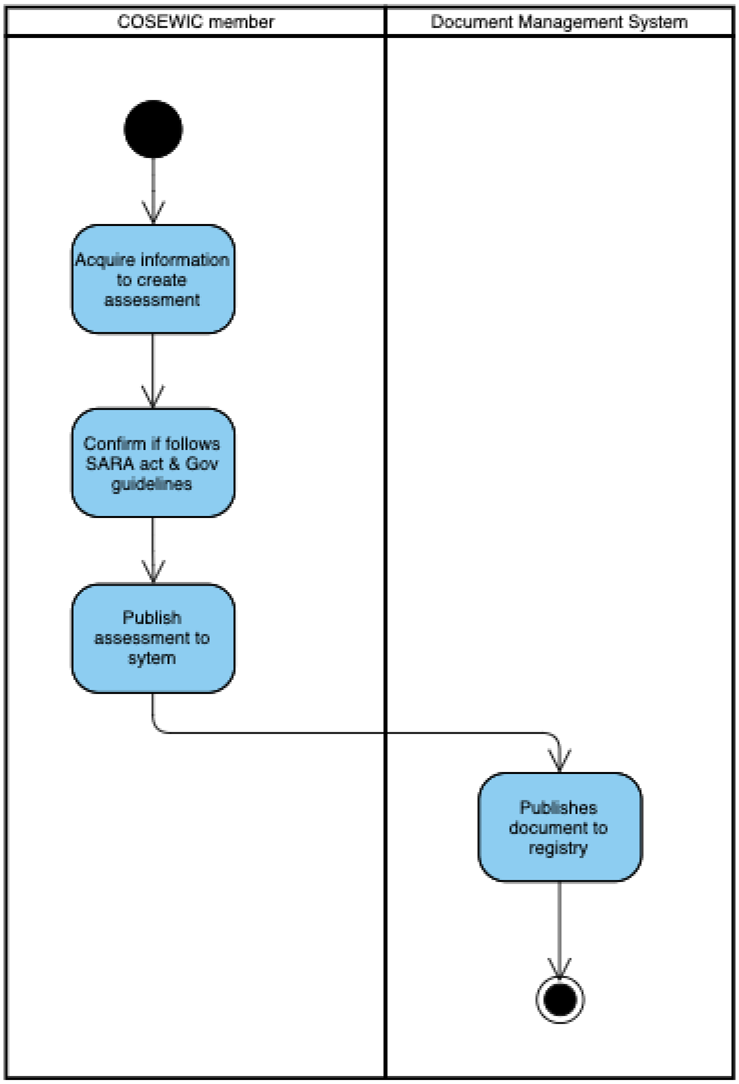
Use Case 002: Manage Species


COSEWIC member
Document Management System
Determine which species need to be managed
Update system in accordance to guidelines
Species updated, accessible to users

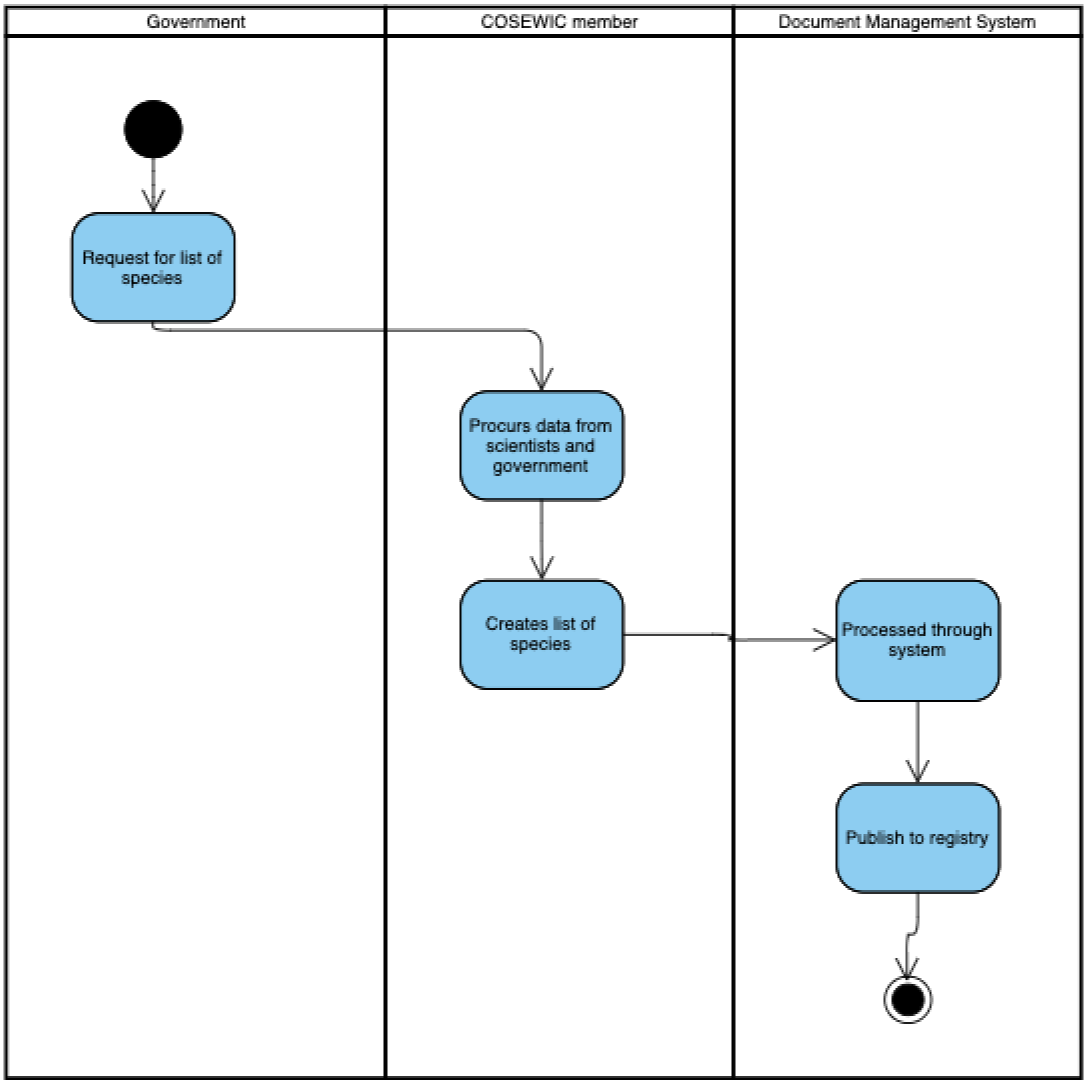
Use Case 003: Manage Species Assessments


COSEWIC member
Document Management System
Look for species to access & add details to
System brings up information requested
Shares findings of assessment
Yes
No
Is Species at risk?
Add details of risk
Indicate not at risk
Yes
No
Does Species have enough details?
Continue
Input not sufficient information
Record Assessment details in system

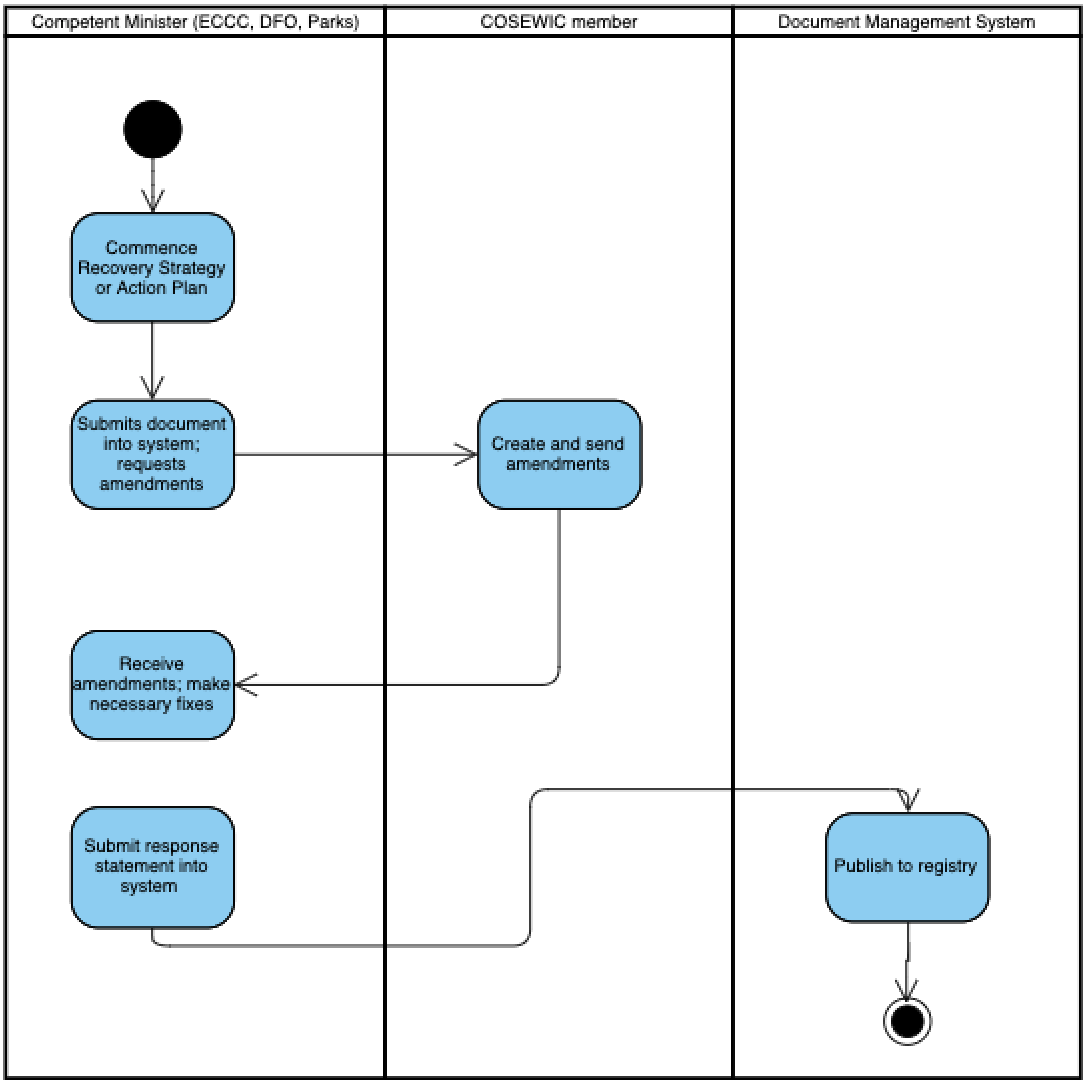
Use Case 004: Publish Assessments


COSEWIC member
Document Management System
Acquire information to create assessment
Confirm if follows SARA act & Gov guidelines
Publish assessment to sytem
Publishes document to registry

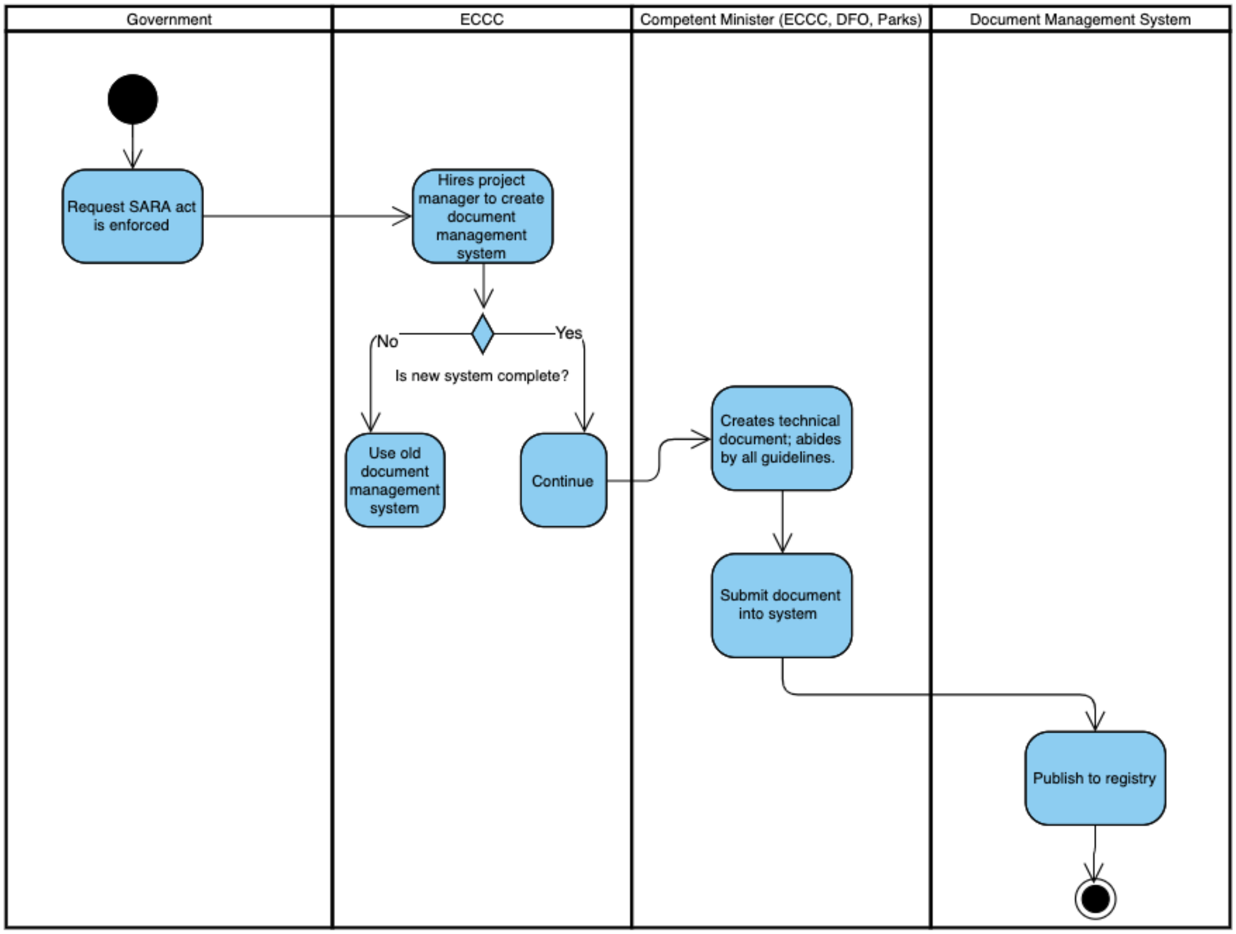
Use Case 005: Publish List of Species


Government
COSEWIC member
Document Management System
Request for list of species
Procurs data from scientists and government
Creates list of species
Processed through system
Publish to registry

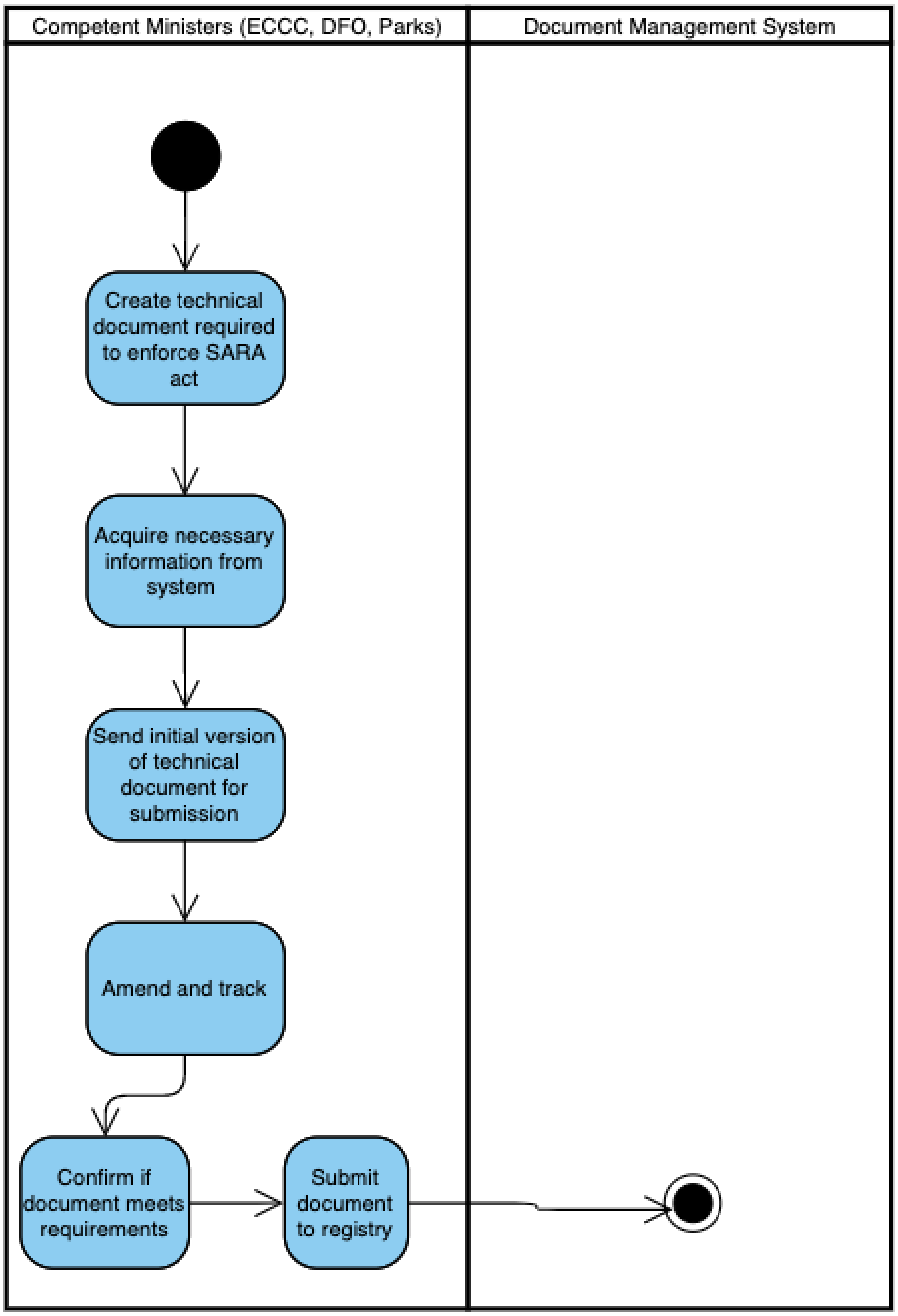
Use Case 006: Publish Response Statement


Competent Minister (ECCC, DFO, Parks)
COSEWIC member
Document Management System
Commence Recovery Strategy or Action Plan
Submits document into system; requests amendments
Create and send amendments
Receive amendments; make necessary fixes
Submit response statement into system
Publish to registry

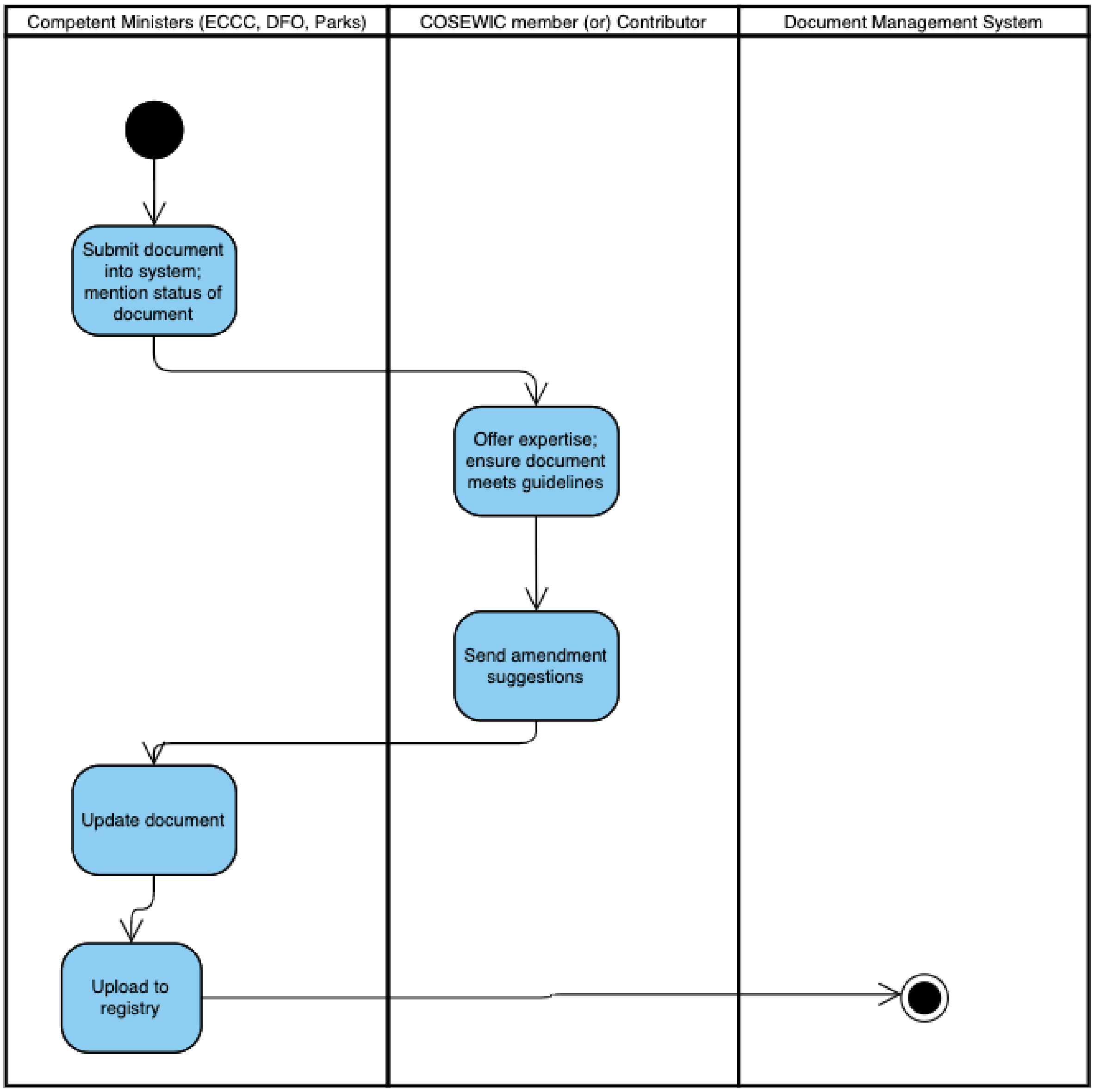
Use Case 007: Publish Recovery Strategies & Action plans to registry


Government
ECCC
Competent Minister (ECCC, DFO, Parks)
Document Management System
Request SARA act is enforced
Hires project manager to create document management system
No
Yes
Is new system complete?
Use old document management system
Continue
Creates technical document; abides by all guidelines.
Submit document into system
Publish to registry

Use Case 008: Create, amend and track documents


Competent Ministers (ECCC, DFO, Parks)
Document Management System
Create technical document required to enforce SARA act
Acquire necessary information from system
Send initial version of technical document for submission
Amend and track
Confirm if document meets requirements
Submit document to registry

Use Case 009: Consults with minister


Competent Ministers (ECCC, DFO, Parks)
COSEWIC member (or) Contributor
Document Management System
Submit document into system; mention status of document
Offer expertise; ensure document meets guidelines
Send amendment suggestions
Update document
Upload to registry

Section 7: Introduction to OO and structural models

List of objects (Nouns)

The following objects were determined by using the "noun, behaviour and attribute" method. We analyzed the activity diagrams, use case scenarios and use case diagram. This is a draft of how we would organize the system.

Red items will be used to create class diagrams with attributes and behaviour

Black items will not be considered in the diagram at the moment. This is to promote simplicity in the design for the moment (until more information is provided).

"Sub-objects" are shown in the list by ***sub-bullets.*** *It is likely we'll use inheritance on them.*

- User
    - COSEWIC member
    - Competent Minister
    - ECCC (employees)
    - Contributors
- System Features
    - Database
    - UI Design
    - Nofication System
    - Document Status System
    - Document Writer
    - Document Upload
    - Communication Hub
- Public registry
- Documents
    - Recovery Strategies
    - Action Plans
- Species
    - (e.g.) Bird
    - (e.g.) Mammal
- Assessment
- Threats

Create classes, attributes and methods (using as much of the 4 pillars)

We decided to meet this requirement by creating a table to relate the classes, methods and attributes. An interface was included in the table to use the 4 pillars.

Classes	*Methods*	*Attributes*
User	*+getUserName(),* *+setUserName(),* *+getPassword(),* *+setPassword(),* *+getFullName(),* *+setFullName(),* *+getStatus()* *+setStatus()* *+getEmail()* *+setEmail()*	*-name, -email,* *-username, -password,* *-userStatus,*

COSEWIC member
(Subclass of User)
	+amendDocument()
	+sendMessage()
		-accessId, cosewicId

Minister
(Subclass of User)
	+createDocument()
	+amendDocument()
	+publishActionPlan()
	+publishRecoveryStrat()
	+sendRequest()
		-ministerId, -documentId,
		-actionPlanId,
		-recoveryStratId

SystemFeatures	*+getSystemSpecs()*
	+getSystemFeatures()
		-systemId,
		-systemStatus,
		-isWorking

Document Writer *(Subclass of SystemFeatures)*	*-displayDocumentUI()* *-writeDocumentToSystem()* *-getDocumentFromDB()* *-setDocumentInfo()* *+getTemplateDoc()*	*-docWriterStatus*

Communication Hub
(Subclass of SystemFeatures)
	+displayCommHub()
	+readUserMetadata()
		-commHubStatus

Documents (*Interface****)***	*readDocument()*
	amendDocument()
	trackDocument()
	publishDocument()
		documentId,
		documentStatus,
		isPublished,

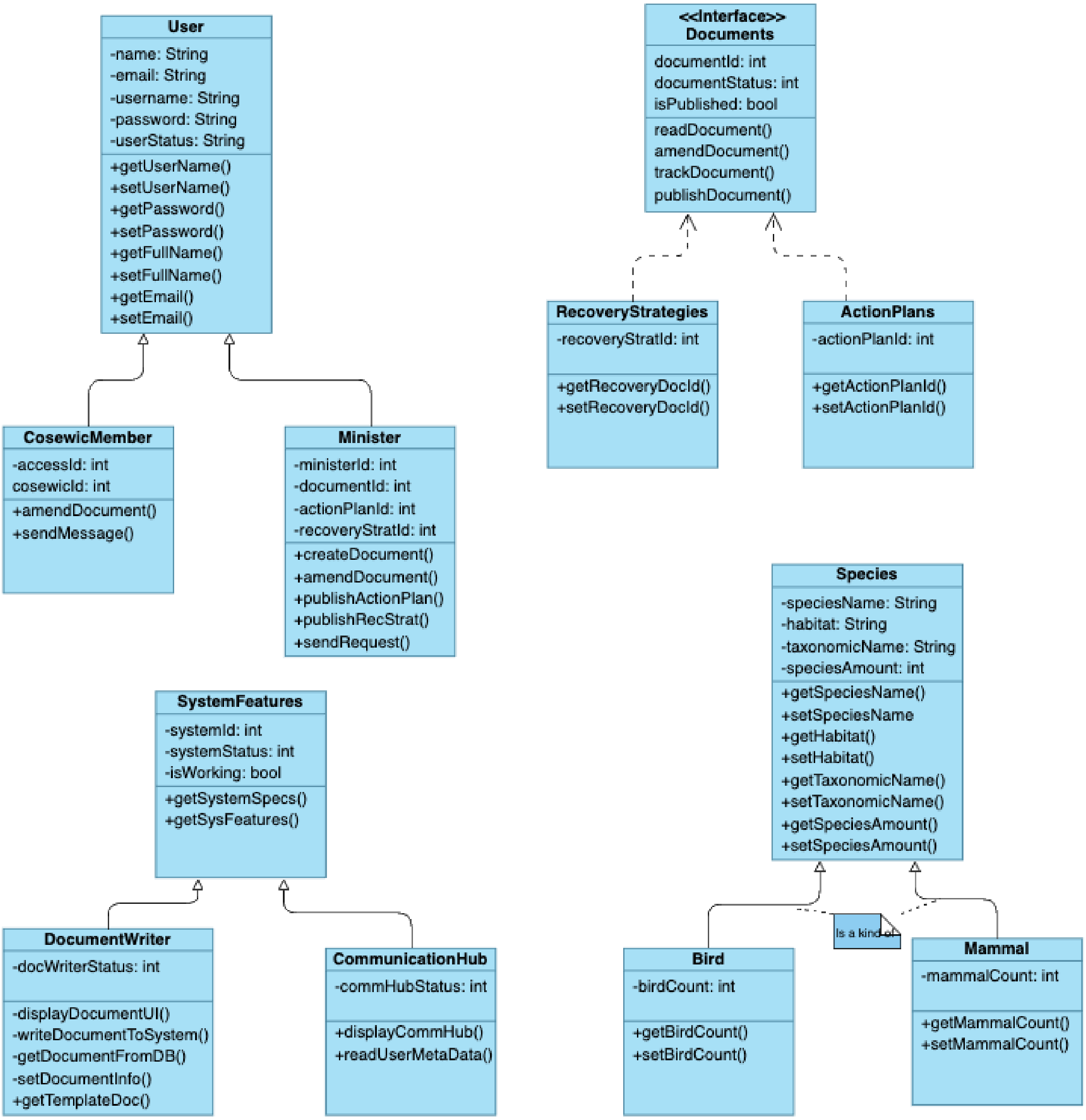
Recovery Strategies
(Implements Documents)
	implements Documents interface methods.
	+getRecoveryDocId()
	+setRecoveryDocId()
		implements Documents interface attributes.
		-recoveryStratId

Action Plans
(Implements Documents)
	implements Documents interface methods.
	+getActionPlanId()
	+setActionPlanId()
		implements Documents interface attributes.
		-actionPlanId

Species	*+getSpeciesName()*
	+setSpeciesName()
	+getHabitat()
	+setHabitat()
	+getTaxonomicName()
	+setTaxonomicName()
	+getSpeciesAmount()
	+setSpeciesAmount
		-speciesName, -habitat,
		-taxonomicName,
		-speciesAmount

Bird *(Subclass of Species) (Uses Polymorphism)*	*+getBirdCount* *+setBirdCount*	*-birdCount* *-status*
Mammal *(Subclass of Species) (Uses Polymorphism)*	*+getMammalCount* *+setMammalCount*	*-mammalCount*

Class Diagram



Section 8: Dynamic models

Sequence Diagrams

Diagram 1: Publish Action plans to registry

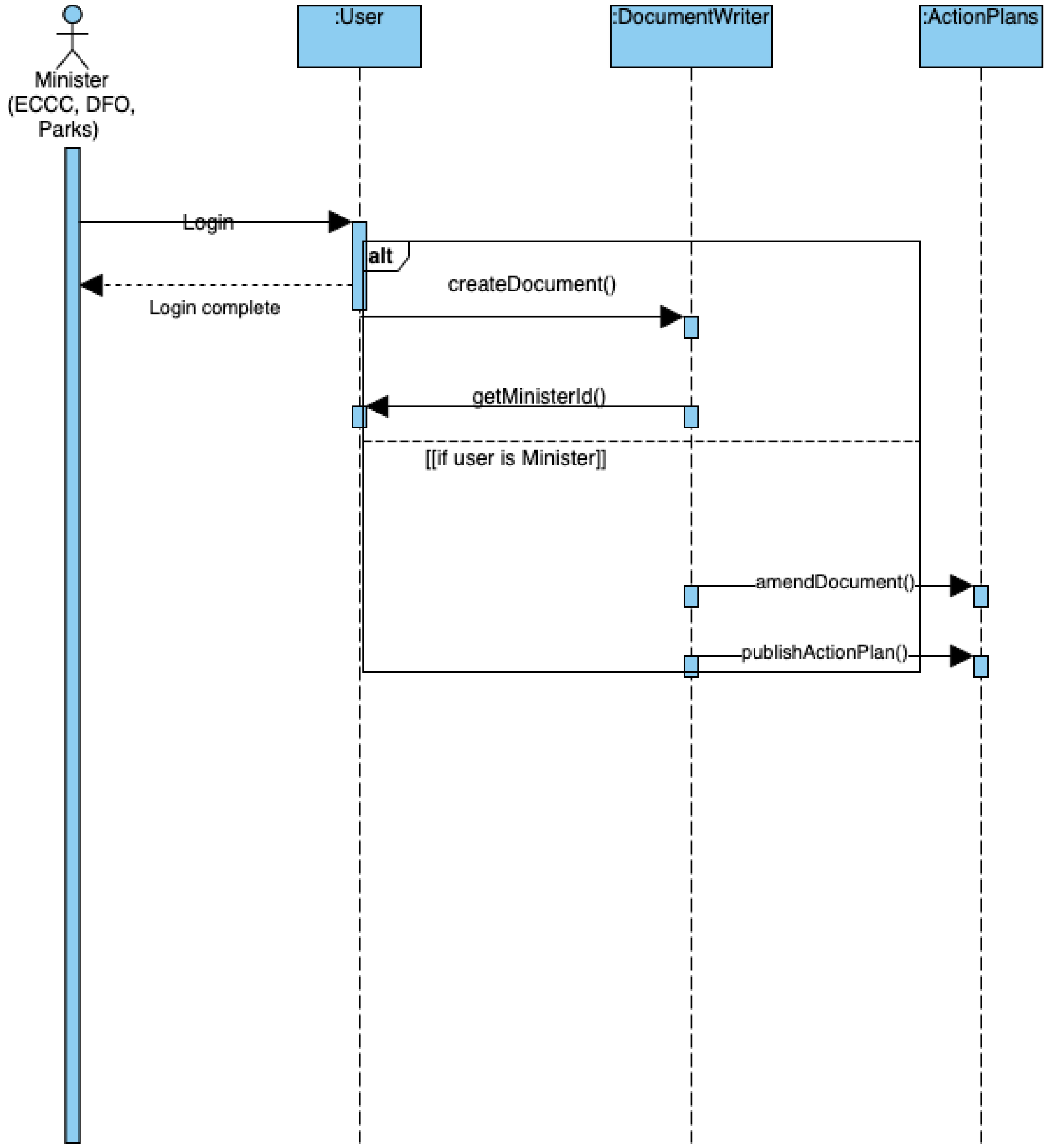
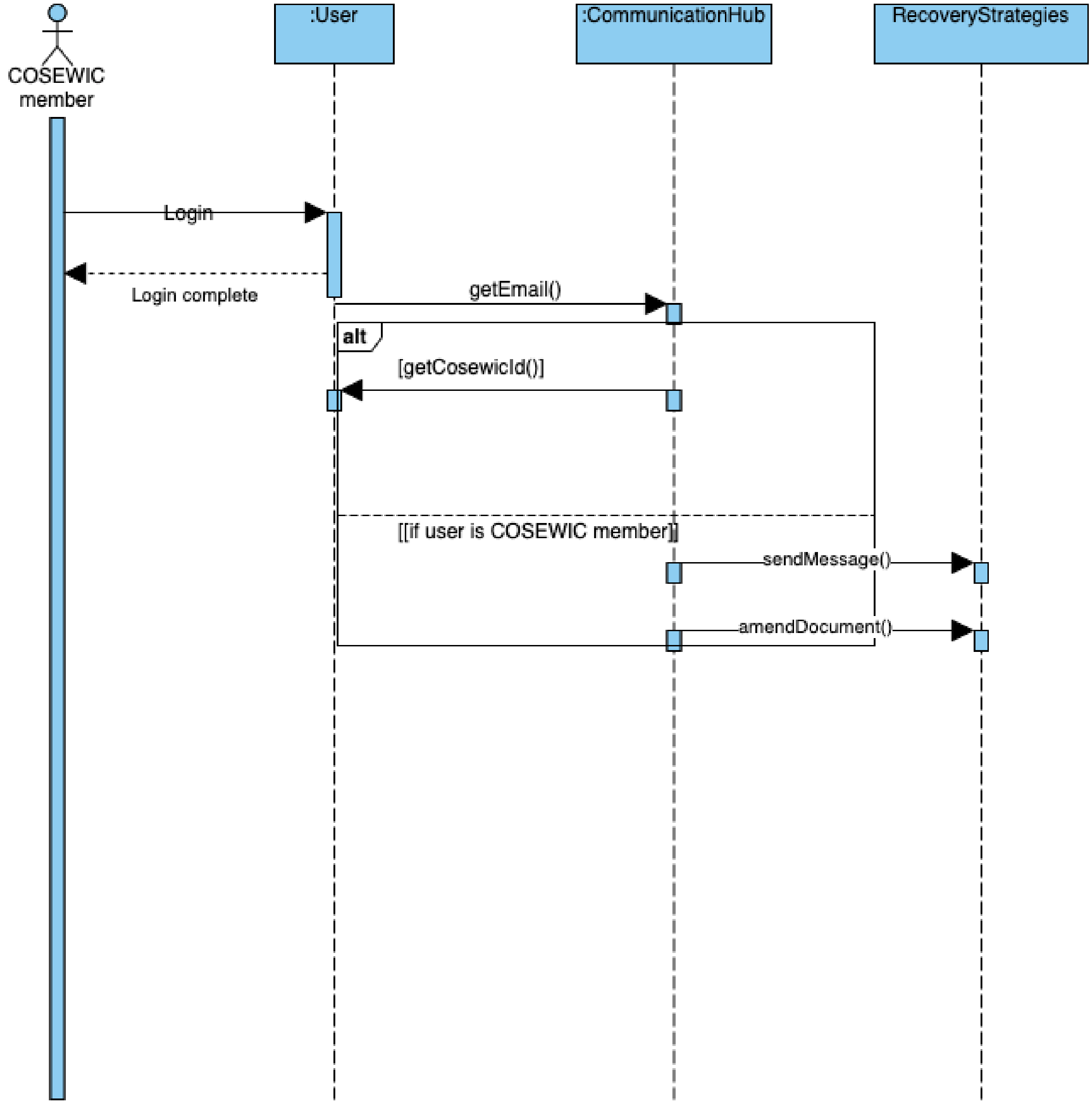
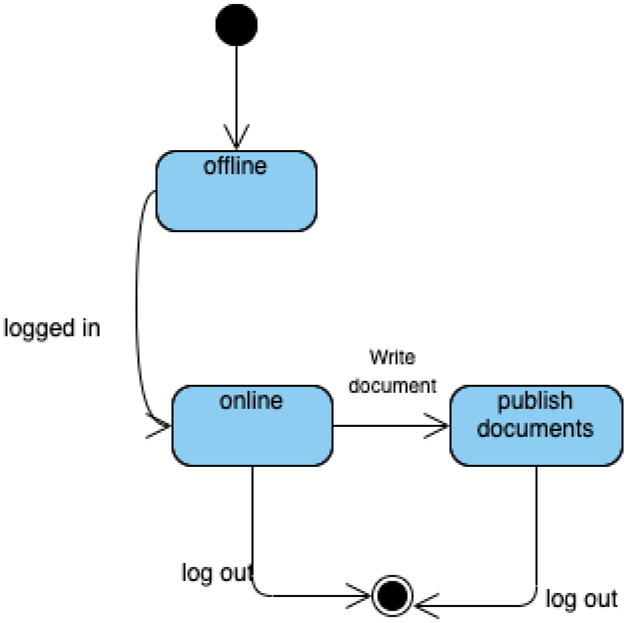


Diagram 2: Send assessment to minister

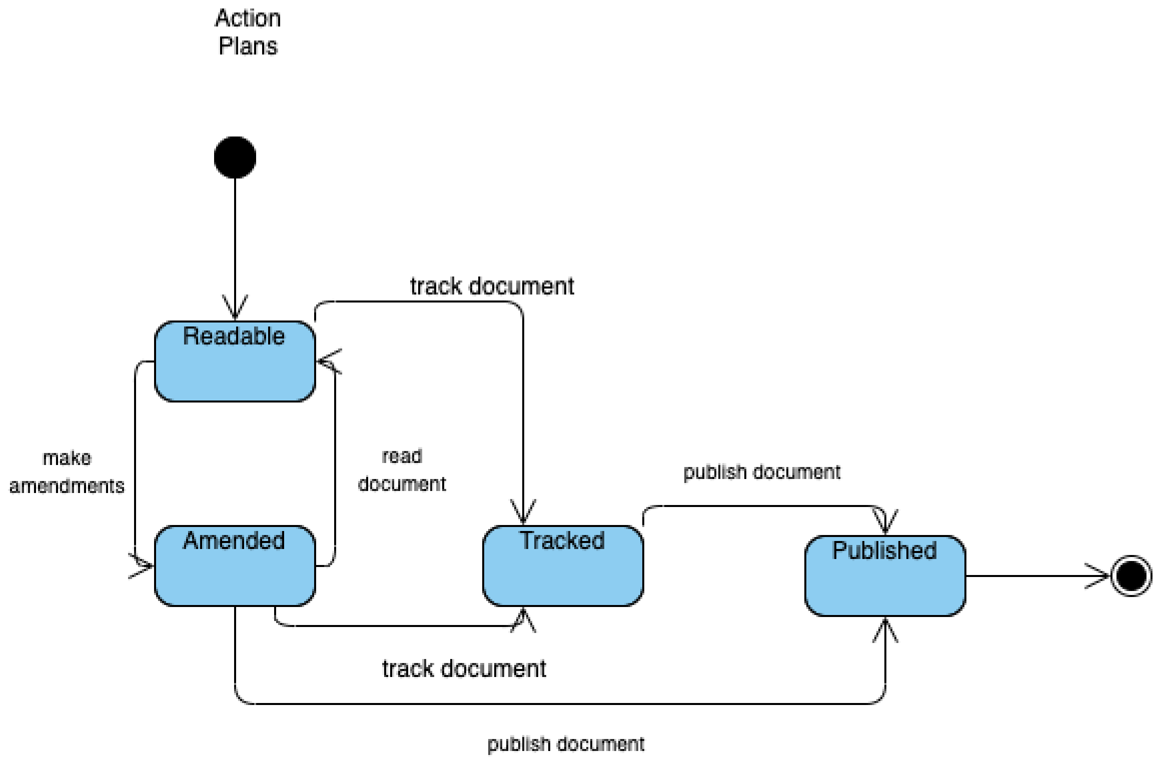


State Machine Diagram

Minister state diagram:



Action Plans state diagram (For action plan object implementing document interface):



Section 9: Design Patterns

Create interfaces to your classes to promote polymorphism

We have decided to preserve adding interfaces into our class diagram until we apply the GoF design patterns. We believe this to be best practice. I hope this is understandable.

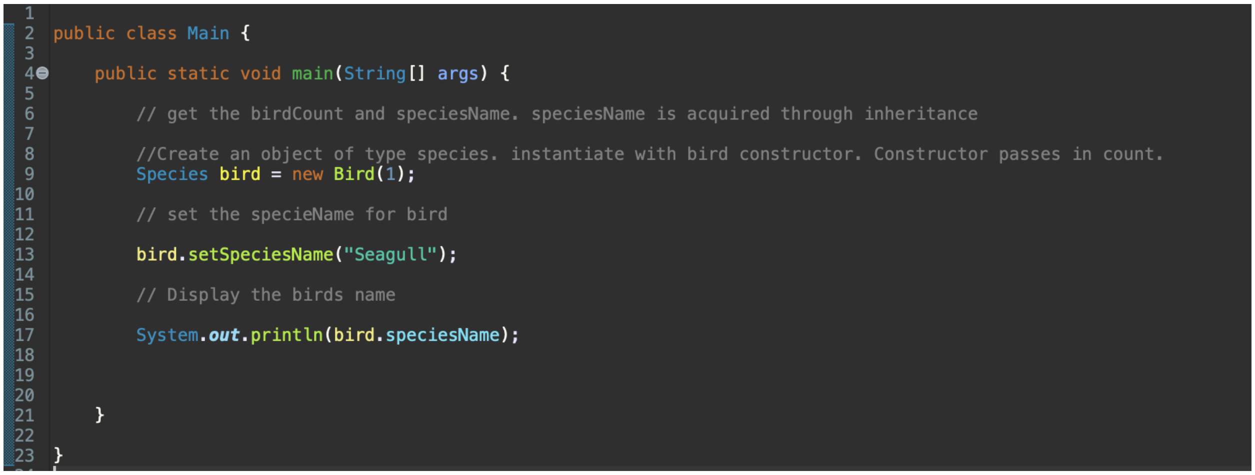
Please see the next requirement to see more interfaces applied to the class diagram.

Apply at least 3 of the GoF Design Patterns to your models

Factory Pattern

The factory pattern is a great pattern to implement for the Species class in our class diagram. Currently, the species class has two subclasses: Bird and Mammal. We know that in a real system, there will be a plethora of animals to access (from the perspective of the developers and the users). In order to ease the process of creating specific animal objects, we can use the factory pattern. Here is a snippet of code to explain the thought process.

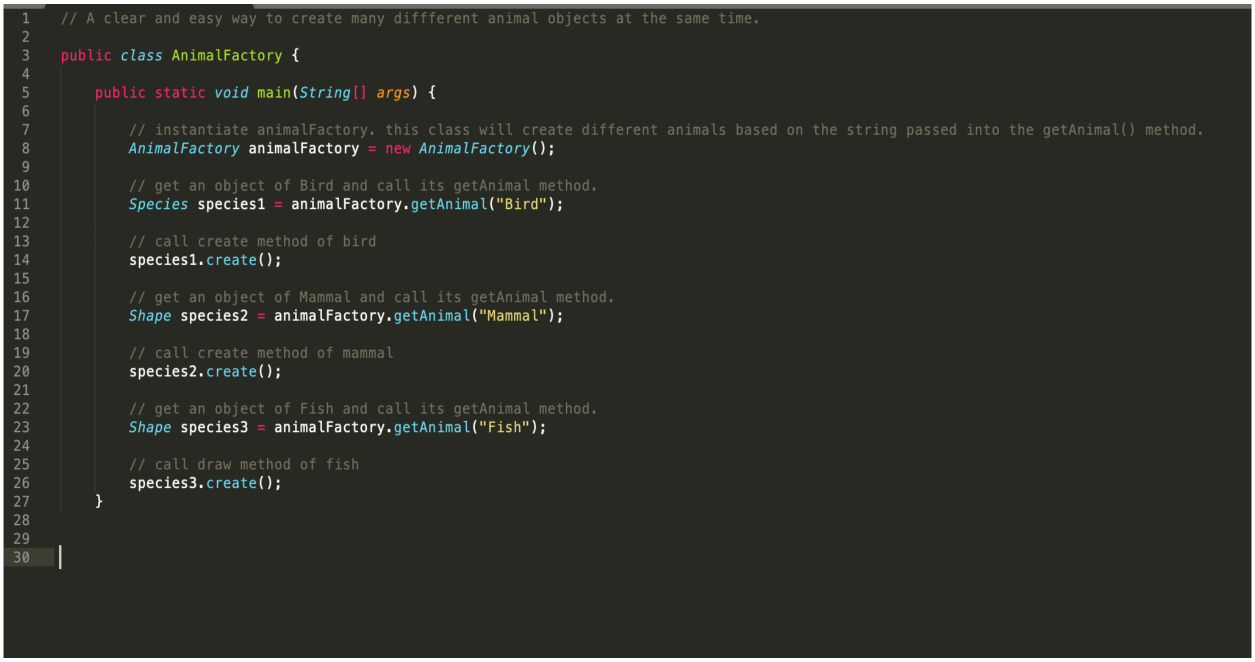
Code before using factory pattern:



```

public class Main {

    public static void main(String[] args) {

        // get the birdCount and speciesName. speciesName is acquired through inheritance

        //Create an object of type species. instantiate with bird constructor. Constructor passes in count.
        Species bird = new Bird(1);

        // set the specieName for bird

        bird.setSpeciesName("Seagull");

        // Display the birds name

        System.out.println(bird.speciesName);



    }

}
```

This code is alright, but you would have to specify the name of each animal every time you want to instantiate an animal in the system. If we use the factory method:

When you implement the **factory** pattern:



```java
// A clear and easy way to create many diffferent animal objects at the same time.

public class AnimalFactory {

    public static void main(String[] args) {

        // instantiate animalFactory. this class will create different animals based on the string passed into the getAnimal() method.
        AnimalFactory animalFactory = new AnimalFactory();

        // get an object of Bird and call its getAnimal method.
        Species species1 = animalFactory.getAnimal("Bird");

        // call create method of bird
        species1.create();

        // get an object of Mammal and call its getAnimal method.
        Shape species2 = animalFactory.getAnimal("Mammal");

        // call create method of mammal
        species2.create();

        // get an object of Fish and call its getAnimal method.
        Shape species3 = animalFactory.getAnimal("Fish");

        // call draw method of fish
        species3.create();
    }


```

You can create many animals simultaneously in a clean manner.

Facade Pattern

The Facade pattern was applied to the SystemFeatures class.

The SystemFeatures is meant to represent some of the core underlying features that would be available in the document management system. Originally it was a superclass and the subclasses were DocumentWriter and CommunicationsHub.

We implemented the Facade pattern because we knew how complex the SystemFeatures would get if all the necessary components were implemented in the class diagram (e.g. databases). The facade pattern serves as a medium to reduce complexity and allow users to access the SystemFeatures interface and its implementations with simple methods. As the stakeholder requirements continually evolve in the future, we must ensure the overarching complexity is limited.

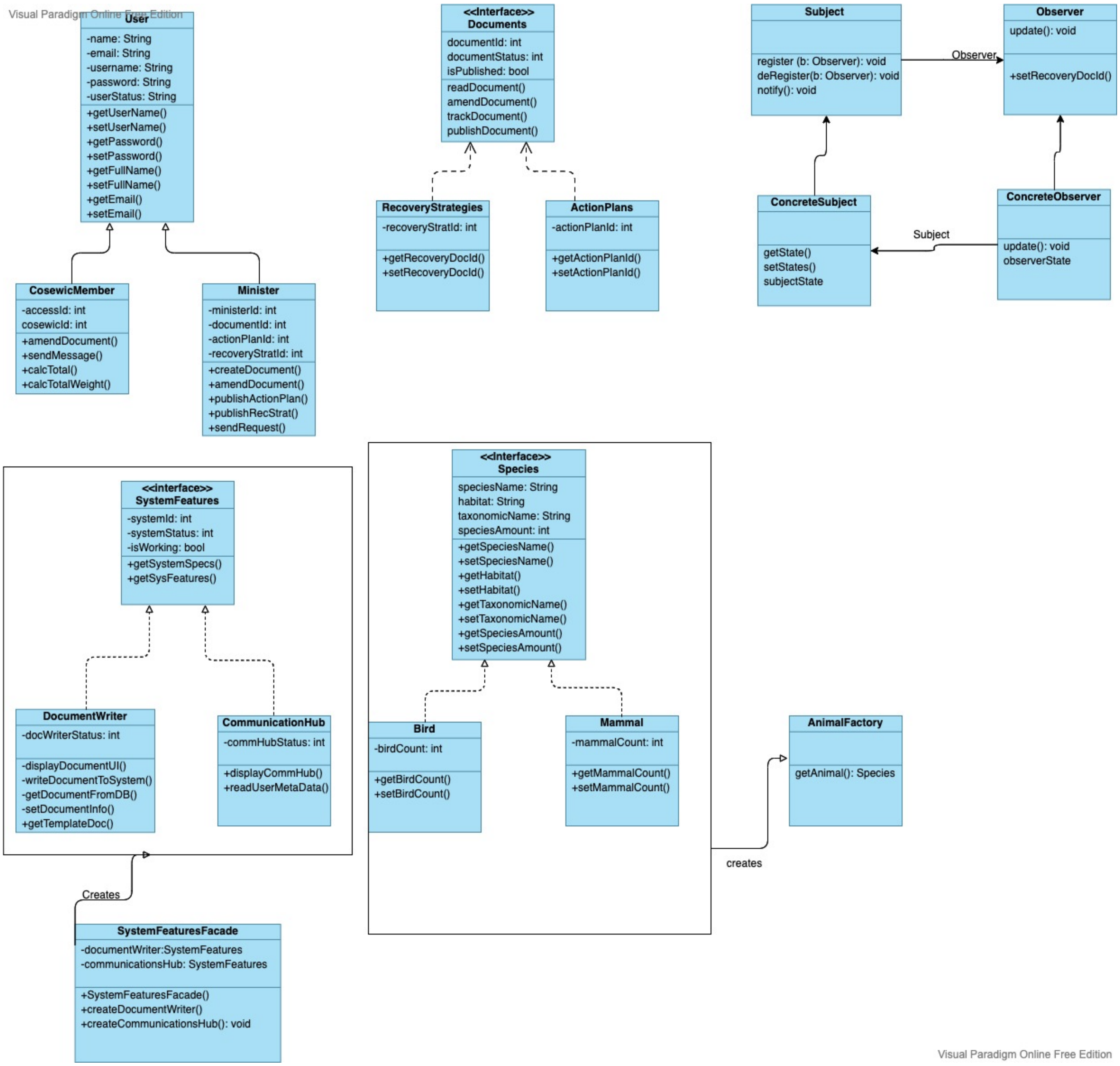
Observer Pattern

The observer pattern was implemented within the project. It is meant to serve the documents interface and respective classes. There will be several documents inputted into the system, each having their own unique specifications despite similarities. As the documents change, we want to be able to notify competent ministers, COSEWIC members, etc.

As one object changes state (documents), we want that to have an affect on other objects (users) without coupling the objects tightly together. The observer pattern is good for this.

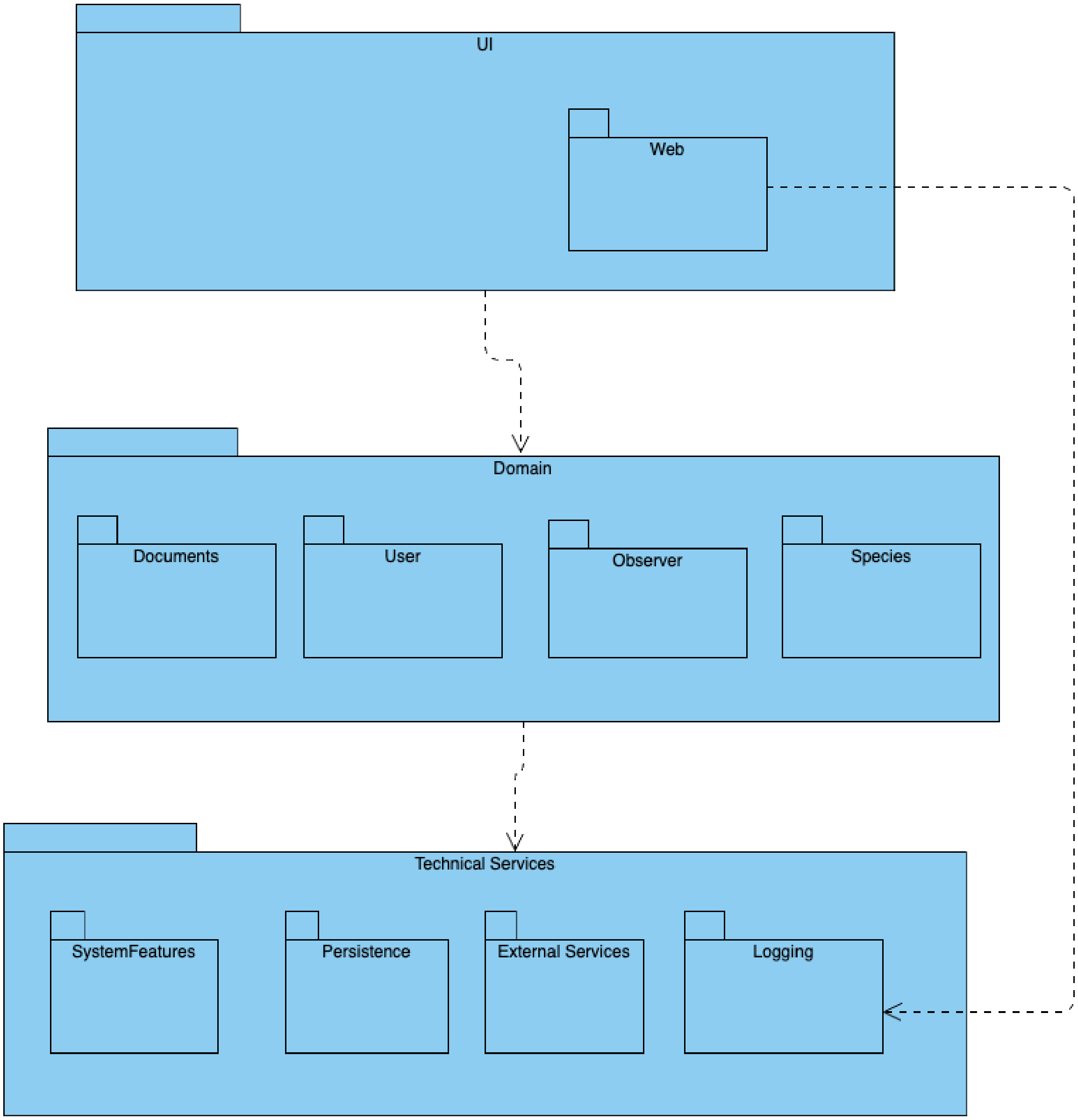
All design patterns are implemented in the class diagram below.

Updated Class Diagram (with 3 GoF design patterns + interfaces implementation):



Section 10: Refining Modeling

Package Diagram



Section 11: Test Plan

1. Executive Summary

As the document is being completed and the overall project is being completed by various members/stakeholders, we expect the fulfillment of the document management system to be completed. The project manager will have achieved his task given by ECCC (which was, of course, issued by the government) and all necessary deliverables outlined will be procured at the end of the time frame.

It is also expected that individuals like testers and software developers will be available to troubleshoot any bugs that emerge within the system.

2. Stakeholders

The individuals involved in test activities are going to be:

- Software Developers (test the system)
- Project Managers (ensure deliverables have been completed & lead team members)
- Competent Ministers (act as primary actors within the system before its official launch)
- Actors that will eventually use the system (e.g. COSEWIC, Contributors)
- Select members of the public accessing the front end (i.e. registry)
- System Analysts
- Business Analyst
- Selected members of public

3. Objectives

Objectives will be determined in the following table below.

For Interfaces with external applications, we are assuming this objective is referring to external applications that will be available on the public registry (web app). These applications can be API's or apps that link to the website.

Objectives	***Software***	***Hardware***	***Documentation***	***Features of Requirements***	***Interfaces with external applications***
001.	Test if system works on various OS (Windows, Mac)	Test how the system will work on lower end PC's (minimum i5 8g ram).	Ensure all the necessary official documents are completed before system launch	The various requirements specified in this document will have the features to meet the requirements (e.g. system can upload documents for the publish to system requirement)	The registry (that is connected to the system) will be able to integrate API's if necessary to meet the requirements of the ever changing web app world
002.	Test if the size of the file is appropriate	Test the load on the CPU and memory	Ensure that all aspects of the testing process is recorded.	The communication system within the document management system includes the features necessary to make conversation seamless.	The website will be designed such that government links that lead to an external web application (e.g. a form) will be easily integratable in the registry

003.	Test if the software is accessible to the necessary audiences (Government mandated)	Test the public registry (web app) on different hardwares (e.g. cell phones, tablets)	Ensure that all necessary documentation will be kept on the system until its mandated expiration date (e.g. documents submitted prior to amending them)	The system will have a seamless database of data users of the system require (e.g. taxonomic data).	The external applications will have a interface specifically designed for mobile.

Objectives can be continued for quite a while. This is a good representation of a set of objectives that would be achieved once the test plan is implemented and the system is deployed.

4. Schedule

Once the test cases are completed, we will consider adding test tasks into the WBC and Gantt chart. Since there are so many different types of testing, the tests will be determined such that it is the best for the system.

5. Testing Tools

The tools that will be used for test activities include:

- Cleverly written algorithms (e.g. if you need to test if .doc and .pdf files can be uploaded to the system, a proper algorithm can test both without hindering the quality of the test).
- Methodologies that are considered the industry standard for Government projects (e.g. automatic logging of errors within the system using an external test tool).

The scenarios for which the test activities will be recorded will be within a test management tool (e.g. Zephyr scale). If the test management tool proves to be too difficult to use (i.e. more time spent learning the application as opposed to working), we will return to original methods of testing.

Bugs will be tracked with management tools like Monday.com. Once again, if the software is unintuitive, we will opt for the most efficient and effective solution (a document with bugs shared amongst team members). Platforms like StackOverflow may be appropriate for some portions of the project.

Section 12: Test Cases

Stakeholder requirements test cases

ID	Requirement	Scenario	Expected Result	Result
TC001	REQ001	Can create documents within the system	Documents are created in the system	
TC002	REQ001	Can amend documents within the system	Documents can be amended in system	
TC003	REQ001	Can track documents within the system	Documents can be tracked in system	
TC004	REQ001	Can publish documents within system	May be difficulty with certain files	
TC005	REQ002	Can communicate with COSEWIC members using integrated communication service	Users will not use it as much; update UI and make easier to use	

TC006	REQ002	COSEWIC member can send message to ministers	COSEWIC member is able to access minister email and send a direct message	
TC007	REQ003	Minister is able to access email of contributors and contact them	Minister will be able to send email	
TC008	REQ003	Minister is able to communicate with contributors using integrated communication service	Minister would rather communicate via other means	
TC009	REQ003	Contributor is able to access technical documents	Provided the contributor has login, he/she will be able to access system.	
TC010	REQ004	Document creation interface is within system	Continue to iterate until the integrated writing system is more appealing than popular writers.	
TC011	REQ004	Database contains necessary information to create documents	Databases contain said information; properly organized	

TC012	REQ004	All information (e.g. contact lists, land information) is organized properly	The information is designed in accordance to government requirements; requirement is fulfilled	

Solution requirements test cases (Functional)

ID	Requirement	Scenario	Expected Result	Result
TC013	**SR-001**	user must be able to create .doc and .pdf files within system	.doc. and .pdf files can be created.	
TC014	**SR-001**	user must be able to save documents to system	database for documents is accessed when user clicks necessary button	
TC015	**SR-002**	user can send request to COSEWIC member for amendment	COSEWIC member responds with message created within system	
TC016	**SR-002**	user can send request to contributor for amendment	Contributor responds with message created in system	

TC017	**SR-003**	user must be able to view icons that mention document status	colored icons allow for easy tracking within system.	
TC018	**SR-003**	user gets notifications for different stages of document	message is sent to users profile page indicating status of documents	
TC019	**SR-004**	user can publish .pdf and .doc files to registry	.pdf and .doc files are parsed through a domain layer class	
TC020	**SR-004**	backend developers must test storage capacity of registry	developers will find they have sufficient space to store documents.	
TC021	**SR-005**	import feature in system has UI button that says “import”	once clicked, a file explorer opens within the OS	
TC022	**SR-005**	user must be able to cancel out from the import feature system window	once the x is clicked, the window will close.	

TC023	**SR-006**	Files should be able to be accessed on a page within system that has download button	when download button is clicked, a file explorer opens. Can download .pdf and .doc files.	
TC024	**SR-006**	a .doc file can be converted to .pdf during export. Vice versa	after file choice is chosen before export, user will get converted version of file. File is converted with system code.	
TC025	**SR-007**	developer can test connection to registry from the system	send data packet to server that hosts the registry	
TC026	**SR-007**	developer can submit test documents to registry	send documents through clickable button. Initiates code and connects to website.	
TC027	**SR-008**	developers can test communication interface's speed	messages are sent within the system instantly.	

TC028	**SR-008**	developers can test email functionality for COSEWIC workers and ministers	register as mock minister and COSEWIC worker. Send email between the two. Emails succeeds.	
TC029	**SR-009**	ask backend developers to submit a data dump of DB contents	data dump will confirm database holds necessary information	
TC030	**SR-009**	get many users to access a specific database at same time	database is able to manage traffic	
TC031	**SR-010**	user attempts to access front end page using multiple browsers	website works with all major browsers	
TC032	**SR-010**	user must be able to interact with webpage buttons	all clickable buttons on UI have a purpose and do what's intended.	
TC033	**SR-011**	developers can test communication interface's speed	messages are sent within the system instantly.	

TC034	**SR-011**	developers can test email functionalily for COSEWIC workers and ministers	register as mock minister and COSEWIC worker. Send email between the two. Emails succeeds.	

Solution requirements test cases (Non-Functional)

ID	Requirement	Scenario	Expected Result	Result
TC035	**SRN-001**	open document management system. Navigate through all of system.	UI is interactive.	
TC036	**SRN-001**	user uses system casually; is asked if system is easy to use	User responds with their opinion. Opinion affects iterations	
TC037	**SRN-002**	test if writing tool has all major features available in Word	writing tool may be missing some tools. Amend this issue.	
TC038	**SRN-002**	test database accessibility while writing within system	acquiring information from database is seamless. This makes	

			writer appealing.	
TC039	**SRN-003**	developer submits 4 documents, all of which have different states (created, amended, tracked and published)	developer should see different coloured icons for each document.	

Use Case Test Cases

ID	Use Case	Scenario	Expected Result	Result
TC040	UC-001	User clicks on publish button	user is provided with metadata of his/her submission	
TC041	UC-002	User clicks publish for Recovery strategy	Recovery Strategy document uploaded to system	
TC042	UC-002	User clicks publish for Action Plan	Action Plan submitted to system.	
TC043	UC-003	User is able to create, amend and track documents in succession	User is able to do so; receives icons indicating document status	

TC044	UC-003	User is able to click buttons to access said functions	said functions would be available on one of the main pages in the system	
TC045	UC-004	user can access species DB	species DB info displayed	
TC046	UC-004	user can change species data within system	code runs to change necessary data, provided security clearance is cleared	
TC047	UC-005	user can access species assessments	data is acquired from database	
TC048	UC-005	user can amend species assessments	system asks for necessary id, following that amendments can occur	
TC049	UC-006	test if ministers can access assessment	minister logs on, navigates to proper UI, accesses assessment	
TC050	UC-006	find assessment on registry	found on a specific page on website.	

TC051	UC-007	find list of species on registry	found on specific page on website.	
TC052	UC-007	check system database for list of species	list of species can be accessed within system.	
TC053	UC-008	test if minister can receive amendments	icons in system will confirm whether or not minister received them	
TC054	UC-008	user tests speed of email service within system	an acceptable number is acquired by developer	
TC055	UC-009	contributor should be able to log into system	logs into system	
TC056	UC-009	contributor should be able to log in as guest if he/she wishes to	contributor is able to so.	

User Stories Test Cases

ID	User Story	Given	When	Then	
TC057	US-001	Document cannot be created	User attempts to edit document	System display a message saying document access unavailable	
TC058	US-001	Public attempts to access documents	public cannot access documents	System displays message: document not available	
TC059	US-002	Communication system is down	User tries to communicate with cosewic	System display message: communication system down	
TC060	US-003	Necessary information is present	User access databases	data must be retrieved at an ideal speed determined by developers and project manager.	
TC061	US-003	Necessary information is not present	User accesses database	System message suggests an alternative for user (e.g. web link).	

TC062	US-004	Communication system is up	stakeholder attempts to communicate with contributor	the system connects the two individuals rapidly	
TC063	US-005	User receives no suggestions for document	User attempts to submit document	Automated document verification process will occur on document.	
TC064	US-005	User wants to access SARA act for document	User searches for SARA act within system	System provides a URL	
TC065	US-006	User suggestion for system is neglected	When the user attempts to resubmit suggestion	It is marked with a different icon to promote urgency (if needed).	
TC066	US-006	User gives suggestion to developer	Developer receives suggestion	It is saved in the system where all developers can access the suggestion (and/or work on it).	

Design Model Test Cases

ID	**Component**	**Scenario**	**Expected Result**	**Result**
TC067	Species	getAnimal() method is called to create a specific animal type	Animal object will be instantiated provided the correct is passed into the getAnimal() method	
TC068	Documents	amendDocument() method is called for ActionPlans class (implements Documents)	Observer will update() the User component that document has been amended	
TC069	CosewicMember	sendMessage() method is called by CosewicMember	ComminicationHub attribute called commHubStatus (the status of the communication hub) will change its state from 0 to 1 (i.e the hub is turned on)	
TC070	Minister	publishActionPlan() is called by minister object	code will execute. Also within the code publishDocument() method will be called (from Document class). Ensures document is published to system. (Minister does not have full code-related	

			functionality to publish document)	

References

Branch, L. (2021, April 07). Consolidated federal laws OF CANADA, species at Risk Act. Retrieved April 16, 2021, from https://laws.justice.gc.ca/eng/acts/S-15.3/

I. (n.d.). Babok Guide V3. Retrieved from https://book.akij.net/eBooks/2018/September/5b8a80dd494ce/BABOK_Guide_v3_Member.pdf

LinkedIn. (n.d.). Retrieved April 16, 2021, from https://www.linkedin.com/learning/onboarding?dest=%2Flearning%2Fme%3Fu%3D2199673&u=2199673

PMBOK. (n.d.). Retrieved from http://faspa.ir/wp-content/uploads/2017/09/PMBOK6-2017.pdf

Brightspace home. (n.d.). Retrieved April 16, 2021, from https://brightspace.algonquincollege.com/d2l/le/content/313771/Home