



Biographical Affidavit

Report #1

Group 1:

Hipolito Bautista

Justin Chuc

Michael Gomez

Rene Allen

Dennis Arnold

Jahmur Lopez





CMPS4131

Manual Medina

February 02, 2023



TEAM SIGNATURES

SIGNATURE BLOCK			
Statement	I did my share of the work, and I have a general understanding of the contents of the assignment		
Team Member	Contribution	Signature	Date
Dennis Arnold	<ul style="list-style-type: none"> • Functional, Non-functional and Onscreen requirements. • On-screen appearance • Global Control Flow • Preliminary Design • Traceability Matrix • Created Use Cases 		26/2/2023
Jahmur Lopez	<ul style="list-style-type: none"> • Document template Creation for final report • Fully Dressed description • References • Glossary of Terms • Cover Page • Proof read • Use Effort estimation • Document Management 		26/2/2023
Justin Chuc	<ul style="list-style-type: none"> • Hardware Requirements • Connectors and Network Protocols • Proof Reading 	<i>J. Chuc</i>	26/2/2023
Hipolito Bautista	<ul style="list-style-type: none"> • Created all Use Cases • Functional, Non-functional, and on-screen Requirments • Identifying subsystems • UML Diagram 		26/2/2023
Michael Gomez	<ul style="list-style-type: none"> • Team Leader, logistics • Problem Statement • Document Management 		26/2/2023

	<ul style="list-style-type: none">• Plan of work, gantt chart• Edit Use Cases• Proof Read		
Rene Allen	<ul style="list-style-type: none">• Spell & Grammar Check• Proof Read• Edit Use Cases• Use Case Diagram• Scenario Sequence Diagrams	<i>Rene Allen</i>	26/2/2023

INDIVIDUAL CONTRIBUTION BREAKDOWN

Responsibility Level	Team Member Names					
	Dennis Arnold	Jahmur Lopez	Justin Chuc	Hipolito Bautista	Michael Gomez	Rene Allen
Project Management (10 points)	10%	10%	5%	15%	30%	30%
Sec. 1 Customer Statement of Requirements (9 points)		45%			55%	
Sec. 2 System Requirements (6 points)	48%	1%	1%	48%	1%	1%
Sec. 3 Functional Requirements Specification (30 points)	30%	10%		35%		25%
Sec. 4 User Interface Specs (15 points)	50%	50%				
Sec. 5 Domain Analysis (25 points)	15%	15%	15%	20%	20%	15%
Sec. 6 Plan of work (6 points)					100%	

TABLE OF CONTENTS

TEAM SIGNATURES	2
INDIVIDUAL CONTRIBUTION BREAKDOWN	4
TABLE OF CONTENTS	5
CUSTOMER STATEMENT OF REQUIREMENTS	7
Problem Statement:	7
Glossary of Terms:	9
SYSTEM REQUIREMENTS	10
Functional Requirements:	10
Non-Functional Requirements:	10
On-Screen Appearance Requirements:	11
FURPS Table:	11
FUNCTIONAL REQUIREMENTS SPECIFICATION	15
Stakeholders:	15
Actors and Goals:	15
Use cases:	16
Casual Description:	16
Use Case Diagram:	18
Traceability Matrix	19
Fully-Dressed Description:	20
USER INTERFACE SPECIFICATION	44
Preliminary Design:	44
User Effort Estimation:	49
SYSTEM ARCHITECTURE	52
Identifying Subsystems & Architecture Styles:	52
UML Package Diagram	52
Mapping Subsystems to Hardware:	53
Connectors and Network Protocols:	53
Global Control Flow:	54
Hardware Requirements:	54
PLAN OF WORK	55
Gantt Chart for the development of the System:	56

CUSTOMER STATEMENT OF REQUIREMENTS

Problem Statement:

OSIPP(the Office of the Supervisor of Insurance and Private Pension), a part of the Ministry of Finance, which is a part of the Government of Belize. As a regulatory body, it's required to have information for the purpose of due diligence and ensuring insurance companies and pension administrators properly follow the regulations.



OFFICE OF THE SUPERVISOR OF
INSURANCE & PRIVATE PENSION

The primary purpose of the Biographical Affidavit is to gather information on the various people involved with entities that OSIPP is tasked with regulating. These people are usually:

- Shareholders
- Directors
- Senior Management
- Compliance Officers
- Insurance Agents
- Insurance Brokers
- Pension Administrators

The biographical affidavit is thorough, with the information needed for its completion. A form may be sent back multiple times because the information is often incomplete or incorrect. Leading to the submission of the biographical affidavit itself being time-consuming and bothersome since both parties would prefer to perform other tasks.



In addition to submitting the biographical affidavit taking quite a bit of time, the information is extracted for future reference and proper information management.

Since the form is currently being physically submitted, handwritten and typed submissions are received. Handwritten forms are problematic because often, the writing is illegible. While typed forms still need to be manually extracted. Both situations lead to being wasted on what should be a minor task at OSIPP.

OSIPP has already digitized some of its processes in the past; The electronic financial reports and AML(Anti-Money Laundering) risk reports are some of these two. The digitization of these forms has improved efficiency at OSIPP since these are done more frequently than the other forms. Thus saving space for the remaining forms that they process. It's now time for the biographical affidavits to be digitized; it's long overdue since the first attempt was back in 2018.



OSIPP is eagerly awaiting the improvements that this project will bring to the processing of the biographical affidavit. The digitization of this form will significantly increase the efficiency of the process since no physical paper processing will need to be done. Since no paper is involved throughout the process, it's less bothersome for the staff who will have to manage the files. Having the information available quickly will significantly improve the speed of the reviews when performed; no more digging through boxes of papers to find the correction file.

We'd like for the digital form to mimic the other digitized forms, where instead of scanning a document and sending it via email, the people who need to fill out the forms will be able to do so with a digital form over a traditional one. We'd also like to add the ability to add comments to note any changes made to the information; for internal use, of course. An example would be; a member of staff reviewing a previously processed biographical affidavit that needs to be updated. They'll note any changes that have been made so that, down the line, future staff will be able to observe what has been changed.

A current issue with the other digitized forms is that sometimes the information they contain will need to be made physical. We'd like to have the ability to print both a tabulated and

pdf version of the biographical affidavit information. They'll be seldom used but when the need arises, it's usually because the information is currently needed elsewhere. Composing files with that information manually is time-consuming; therefore, being able to print like that is important to us.

Glossary of Terms:

Terms	Meaning
Shareholders	A person or institution that has invested money in a corporation in exchange for a "share" of the ownership
Directors	Someone elected or appointed to manage a company's business and affairs
Senior Management	Generally individuals at the highest level of management of an organization who have the day-to-day task of managing that organization; sometimes a company or a corporation.
Compliance Officers	are responsible for ensuring that all corporate processes and procedures comply with the law
Insurance Agents	a person who solicits, negotiates, or instigates insurance contracts on behalf of an insurer and can be independent or an employee of that insurer
Insurance Brokers	An intermediary who sells, solicits, or negotiates insurance on behalf of a client for compensation
Pension Administration	The act or performing various types of yearly service on an organizational retirement plan
Internal use	A system used within an organization which data or processes are not intended for the public.
External use	A system used by actors outside the organization
Digitization	The process of converting information to a digital format
OSIPP	Acronym for "The Office of the Supervisor of Insurance & Private Pension"

SYSTEM REQUIREMENTS

Functional Requirements:

Identifier	PW	Requirement
REQ-1	5	The system shall allow the internal user to login to access the Digital Biographical Affidavit.
REQ-2	5	The system shall allow the external user to fill in the digitized form with the necessary information.
REQ-3	5	The system shall allow the external user to send the form electronically after all information is filled out.
REQ-4	4	The system shall allow the internal user to verify if the form was filled out correctly.
REQ-5	3	The system will archive outdated forms to preserve the records.
REQ-6	5	The system shall allow OSIPP public officers to search for Affidavit data.

Non-Functional Requirements:

Identifier	PW	Requirement
NONREQ-1	2	The system shall provide a non-cluttered, user-friendly, easy-to-understand web application.
NONREQ-2	4	The system shall use an algorithm to transfer user's information to the database and back to the user.
NONREQ-3	5	The system shall provide a unique code to their work email to access the web application.
NONREQ-4	3	The system shall have a holding database to store a form for verification.
NONREQ-5	5	The system shall allow the administrator to verify the forms before having them stored in the database.
NONREQ-6	5	The system shall generate a unique code to associate each form created.
NONREQ-7	2	The system shall store information from the electronic forms on a

		database.
NONREQ-8	5	The system shall allow the user to be warned about incorrect information on the form to be fixed or the field being left blank.

On-Screen Appearance Requirements:

Identifier	PW	Requirement
ONSREQ-1	5	The design of the system shall be related to the color scheme of OSIPP.
ONSREQ-2	4	The system shall display a unique code on the top right of the form.

FURPS Table:

Functionality	<ul style="list-style-type: none"> Users will be able to include their signature at the end of the form.
Usability	<ul style="list-style-type: none"> The web application will have a login screen to access the form. Input boxes are clearly labeled and formatted
Reliability	<ul style="list-style-type: none"> Checks are done to verify if the information has been entered correctly or if there's missing information Version control of the application will be done to for upkeep of the
Performance	<ul style="list-style-type: none"> Multiple users can use the application simultaneously
Supportability	<ul style="list-style-type: none"> Website Application runs on many supported browsers.



Welcome

Username

Password

Log In Cancel

[Forgot your password?](#)

Figure 1. Showing the on-screen requirements: ONSREQ-1



OFFICE OF THE SUPERVISOR OF
INSURANCE & PRIVATE PENSION



Form ID: 2045783820

BIOGRAPHICAL AFFIDAVIT FORM

1. Affiant's Full Name:

2. Other names used at any time:

3. Have you ever had your name changed? ☐ Yes ☒ No
If "Yes", (a) provide previous name(s) and (b) give the reason for the change:

4. Affiant's Identification No. applied to Government Record Systems. Two certified copies of picture ID must be submitted. The documents must be current and valid.

Document	Number of Document	Date of issuance	Country of Issuance
Social Security	<input type="text" value="23262988"/>	<input type="text" value="2/15/21"/>	<input type="text" value="Belize"/>
Passport	<input type="text" value="P00035684"/>	<input type="text" value="10/28/20"/>	<input type="text" value="Belize"/>
National Health Insurance	<input type="text" value="N/A"/>	<input type="text" value="N/A"/>	<input type="text" value="N/A"/>
Other, specified	<input type="text"/>	<input type="text"/>	<input type="text"/>

5. Date of Birth: (DD/MM/YYYY)

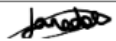
6. Place of Birth: include District/State and Country)

7. Nationality: Indicate how acquired.
☒ Birth
☐ Naturalization
☐ Marriage
☐ Other, specify

8. Spouse's Name:

9. Affiant's Address:

Physical Address	Telephone No.	Fax No.	Email Address
Residential	<input type="text" value="501-223-4567"/>	<input type="text" value="N/A"/>	<input type="text" value="johndoe@gmail.com"/>
Business	<input type="text" value="N/A"/>	<input type="text" value="N/A"/>	<input type="text" value="N/A"/>

Signature: 

Date:

SUBMIT

Figure 2. Showing the on-screen requirements: ONSREQ-1, and ONSREQ-2

OFFICE OF THE SUPERVISOR OF
INSURANCE & PRIVATE PENSION**Form Review & Approval**

ID: 2045783820

Comment

1. Affiant's Full Name:
2. Other names used at any time
3. Have you ever had your name changed? ☐ Yes ☒ No
If "Yes", (a) provide previous name(s) and (b) give the reason for the change:
4. Affiant's Identification No. applied to Government Record Systems. Two certified copies of picture ID must be submitted. The documents must be current and valid.

Document	Number of Document	Date of Issuance	Country of issuance
Social Security	23262998	2/15/21	Belize
Passport	P00035684	10/28/20	Belize
National Health Insurance	N/A	N/A	N/A
Other, specified			

5. Date of Birth: (DD/MM/YYYY)
6. Place of Birth: include District/State and Country
7. Nationality: Indicate how acquired.

☒ Birth☐ Naturalization☐ Marriage☐ Other, specify

8. Spouse's Name:
9. Affiant's Address:

Physical Address	Telephone No.	Fax No.	Email Address
Residential	501-223-4567	N/A	johnndoe@gmail.com
Business	N/A	N/A	N/A

Signature:

Date:

Figure 3. Showing the on-screen requirements: ONSREQ-1

FUNCTIONAL REQUIREMENTS SPECIFICATION

Stakeholders:

A stakeholder is an individual or group with an interest in the operations of an organization. For improved file administration within the sub-department of the Ministry of Finance, the digitized version of the Biographical Affidavit has been created for the Office of the Supervisor of Insurance & Private Pension. Consequently, the following individuals are involved in this process:

- 1.) A site administrator that shall manage all system's data (database and front-end information).
- 2.) The agent shall interact with the system to complete the form.
- 3.) Public Officers of OSIPP shall verify the form that was filled out, to ensure all information is in order

Actors and Goals:

- **Site Administrator**- can change the entire system and manipulate every piece of data. These comprise database details, the front-end interface, and any system logs produced.
- **3rd Party Agents** - can access the system to fill out the biographical affidavit form on the system so it can be sent for verification before having it stored in the database.
- **Public Officers of OSIPP**- can access customer's form to verify if all fields were entered correctly and none were left blank; they'll also be able to access the information stored in the database for their own purposes.

Use cases:

Casual Description:

Name	Description	Requirement Covered
UC 1 - SubmitForm	<ul style="list-style-type: none">• To allow the system to capture all fields in the form that was filled out by the user	REQ-1, REQ-2, REQ-3
UC 2 - AcceptingForm	<ul style="list-style-type: none">• Confirm the form is properly filled out	REQ-3, REQ-4, NONREQ-5
UC 3 - DenyingForm	<ul style="list-style-type: none">• Deny form if not properly filled out	REQ-3, REQ-4, NONREQ-5, NONREQ-6, ONSREQ-1
UC 4 - AddingComments	<ul style="list-style-type: none">• Indicate missing requirements for form	REQ-1, REQ-4, NONREQ-5, ONSREQ-1
UC 5 - StoringData	<ul style="list-style-type: none">• Saving forms to database if approved	REQ-1,REQ-4
UC 6 - TogglingTableView	<ul style="list-style-type: none">• Displays approved affidavits	REQ-1, NONREQ-1
UC 7 - View Archive	<ul style="list-style-type: none">• View all archived affidavits.	REQ-2,REQ-3
UC 8 - ArchivingFile	<ul style="list-style-type: none">• Archive affidavit currently being accessed.	REQ-2,REQ-3

UC 9 - RestoringForm	<ul style="list-style-type: none">• Restore the form using the restoration code they received via email.	REQ-1, NONREQ-1
UC 10 - Download Form	<ul style="list-style-type: none">• To download the affidavit as a PDF	REQ-1, NONREQ-1
UC 11 - Register Account	<ul style="list-style-type: none">• Register Accounts for OSIPP Officers	REQ-1, NONREQ-3
UC 12 - Affidavit Search	<ul style="list-style-type: none">• Allow public officers of OSIPP to search Affidavit's data on the web application	REQ-6 NONREQ-2

Use Case Diagram:

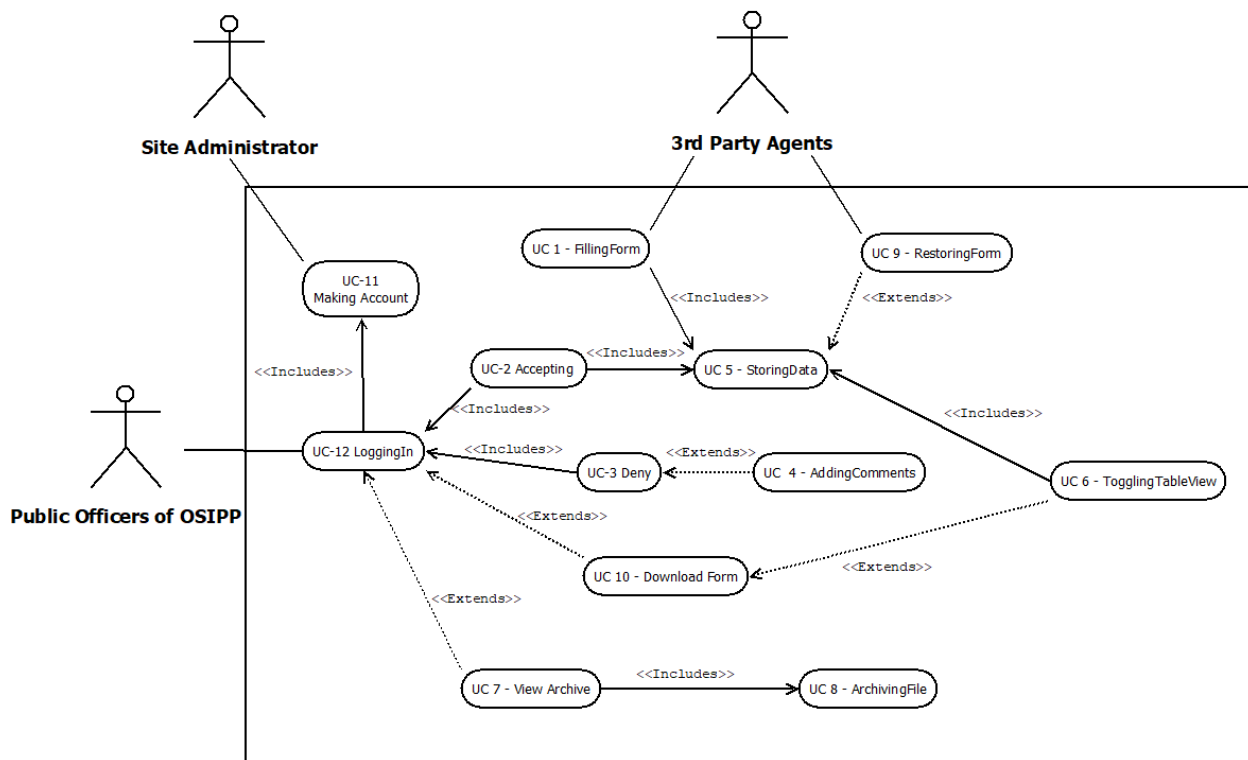


Figure 4. The system's Use Case diagram

Traceability Matrix

PW Value		UC 1 - FillingForm	UC 2 - AcceptingForm	UC 3 - DenyingForm	UC - 4 AddingComments	UC 5 - StoringData	UC 6 - TogglingTableView	UC 7 - View Archive	UC 8 - ArchivingFile	UC 9 - RestoringForm	UC 10 - DownloadForm	UC 11 - MakingAccount	UC 12 - LoggingIn
5	REQ-1	X			X	X	X	X	X			X	X
5	REQ-2	X								X	X		
5	REQ-3	X	X	X									
4	REQ-4		X	X	X	X							
3	REQ-5												
2	NONREQ-1						X	X	X				
4	NONREQ-2												
5	NONREQ-3												
3	NONREQ-4												
5	NONREQ-5		X	X	X								
5	NONREQ-6			X						X	X		
2	NONREQ-7												
5	NONREQ-8									X	X		
5	ONSREQ-1			X	X								
4	ONSREQ-2												
	Total Weight	15	14	19	19	9	7	7	7	15	15	5	5

Table 1. Traceability Matrix according to system requirements and use cases weight

Fully-Dressed Description:

The use cases that received the highest priority weight in the traceability matrix underwent the fully-dressed description. UC-3, UC-4, UC-6 ,UC-7, UC-8, UC-9, UC-10 ,UC-12 and UC-13 were not considered because other use cases such as UC-1 include the processes carried out in these use cases.

Use Case UC-1	SubmitForm
Related Requirements	REQ-1, REQ-2, REQ-3
Initiating Actor	3rd party agents
Actor's Goal	To allow the system to capture all fields in the form that was filled out by the user
Participating Actors	None
Preconditions	The 3rd party agent has all required documents and must be already logged into the application.
Postconditions	The form is now completed
Flow of Events for Main Success Scenario: →1. Third party fills out each field of the form. ←2. Admin office Checks to make sure no fields are not empty. ←3. System Makes Submit button available to the user	

Use Case UC-2	AcceptingForm
Related Requirements	REQ-3, REQ-4, NONREQ-5
Initiating Actor	Public Officers of OSIPP
Actor's Goal	Confirm the document is properly filled out.
Participating Actors	3rd Party Agents
Preconditions	Form has been submitted for verification by a 3rd party user. Public Officers of OSIPP are logged in.
Postconditions	Form will be transferred from “unverified forms” to the “verified forms” table in the database.
Flow of Events for Main Success Scenario: →1. Public officer of OSIPP opens the form that was submitted by a 3rd-party agent for revision. →2. Public officer of OSIPP Clicks on the “Accept” button available to him. ←3. System will move form from “unverified forms” to “verified forms” in the database. ←4. System will display a “success” message to the OSIPP Officer	

Use Case UC-3	DenyingForm
Related Requirements	REQ-3, REQ-4, NONREQ-5, NONREQ-6, ONSREQ-1
Initiating Actor	Public Officers of OSIPP
Actor's Goal	Informs the 3rd party agent that the form has been denied via email with a restoration code. Officers of OSIPP are logged in.
Participating Actors	3rd party agents
Preconditions	Form has been submitted for verification by a 3rd party user. Public Officers of OSIPP are logged in.
Postconditions	
Flow of Events for Main Success Scenario: →1. Public officer of OSIPP opens the form that was submitted by a 3rd-party agent for revision. →2. Public officer of OSIPP include::AddingComments(UC4) →3. Public officer of OSIPP clicks on “deny” button. ←4. System will display a “form has been denied” message to the OSIPP Officer ←5. System Sends email to the 3rd party agent that the form has been denied. Within this email a restoration code will be included along with steps on how to utilize it.	

Use Case UC -4	AddingComments
Related Requirements	REQ-1, REQ-4, NONREQ-5, ONSREQ-1
Initiating Actor	Public Officers of OSIPP
Actor's Goal	Indicate fields which have been improperly filled out. Officers of OSIPP are logged in.
Participating Actors	None
Preconditions	Form has been submitted for verification by a 3rd party user. Public Officers of OSIPP are logged in.
Postconditions	Form will be commented by Public officers of OSIPP
Flow of Events for Main Success Scenario: →1. Public officer of OSIPP opens the form in order to use the “add comment” button to place remarks beside the chosen field(s). ←2. System will add a comment field beside the field for the OSIPP officer to input into →3. Public officer of OSIPP will add comments ←4. System will save data inputted in the new comments field by the OSIPP officer.	

Use Case UC -5	StoringData
Related Requirements	REQ-1,REQ-4
Initiating Actor	Public Officers of OSIPP
Actor's Goal	To submit form to be saved in the database
Participating Actors	None
Preconditions	Form fields have been entered correctly and verification was successful.
Postconditions	Form information will be in the database for viewing.
Flow of Events for Main Success Scenario: →1. Public officer of OSIPP Clicks on the “Approve” button available to him. ←2. System will display a “success” message to the OSIPP Officer ←3. System will send record to the stored in database →4. Public officer of OSIPP can view stored records in database	

Use Case UC -6	TogglingTableView
Related Requirements	REQ-1, REQ-6
Initiating Actor	Public Officers of OSIPP
Actor's Goal	Displays approved affidavits
Participating Actors	None
Preconditions	The user is logged into their OSIPP officer account
Postconditions	The affidavits the user has searched for will be listed.
Flow of Events for Main Success Scenario: →1. Public officer of OSIPP searches for an affidavit (UC-13) ←2. System will search through the data in the “approved” affidavits table. ←2. System will display all affidavits which match the users search.	

Use Case UC -7	ViewArchive
Related Requirements	REQ-1, REQ-6
Initiating Actor	Public Officers of OSIPP
Actor's Goal	View all archived affidavits.
Participating Actors	None
Preconditions	OSIPP officer is logged in.
Postconditions	Show all archived affidavits
Flow of Events for Main Success Scenario: →1. Public officer of OSIPP selects “View Archived Affidavits” in the menu. ←2. System Loads user into new page ←2. System Loads all affidavits saved in the “Archived affidavits” table from the database.	

Use Case UC -8	ArchivingFile
Related Requirements	REQ-1, REQ6
Initiating Actor	Public Officers of OSIPP
Actor's Goal	Archive affidavit currently being accessed.
Participating Actors	None
Preconditions	View list of files that were approved.
Postconditions	Remove file from the “approved” table and place it in the “archived affidavits” table.
<p>Flow of Events for Main Success Scenario:</p> <p>→1. Public officer of OSIPP selects “View Archived Affidavits” in the menu (UC-7).</p> <p>→2. Public officer of OSIPP presses “archive” button next to the affidavit they wish to archive</p> <p>←3. System Moves the affidavit from the “approved” table into the “archived affidavits” table</p> <p>←4. System Deletes the affidavit from the “approved” table.</p> <p>←5. System Informs the OSIPP office that the affidavit was successfully archived.</p>	

Use Case UC-9	RestoringForm
Related Requirements	REQ-2,NONREQ-6, NONREQ-8
Initiating Actor	3rd party agents
Actor's Goal	Restore the form using the restoration code they received via email.
Participating Actors	None
Preconditions	Form must be submitted for review
Postconditions	Previously rejected form is loaded with it's comments.
Flow of Events for Main Success Scenario: →1. 3rd party agent presses the “Restore Previous Form” button. ←2. System Prompts user for Form ID via popup with instructions. →3. 3rd party agent enters ID /registration code. ←4. System Loads the form associated with the Form ID with it's comments.	

Use Case UC-10	DownloadForm
Related Requirements	REQ-2,NONREQ-6, NONREQ-8
Initiating Actor	Public Officers of OSIPP
Actor's Goal	To download the affidavit as a PDF
Participating Actors	None
Preconditions	OSIPP Officer is in PDF view
Postconditions	Affidavit has been downloaded as a PDF
Flow of Events for Main Success Scenario: →1. Public officer of OSIPP Presses “Download” button available beside all affidavits. ←2. System Gets all data associated with affidavit and converts to PDF. ←3. System Initiates download to OSIPP officer’s local machine.	

Use Case UC-11	Making Account
Related Requirements	REQ-1
Initiating Actor	Site Administrator
Actor's Goal	Register Accounts for OSIPP Officers
Participating Actors	Public officer of OSIPP
Preconditions	Must have the work issued email address of OSIPP officer
Postconditions	Registered OSIPP officer account
Flow of Events for Main Success Scenario: →1. Site Administrator clicks “Create Account” to add OSIPP officer’s email address into the “email address” section of the “users” table. →2. Site Administrator adds OSIPP officer’s password into the “password” section of the “users” table. ←3 System stores accounts created by the site administrator on the database.	

Use Case UC-12	Affidavit Search
Related Requirements	REQ-6, NONREQ-2
Initiating Actor	Public Officers of OSIPP
Actor's Goal	Search of Affidavit in the database
Participating Actors	Public officer of OSIPP
Preconditions	Affidavit form must passed the verification process
Postconditions	Affidavit's data will be stored after successful verification
Flow of Events for Main Success Scenario: →1. OSIPP Public Officers search for Affidavit by either first name, last name or with ID. ←2 System will communicate with the database to verify if the searched item exist or not.	

State Sequence Diagram

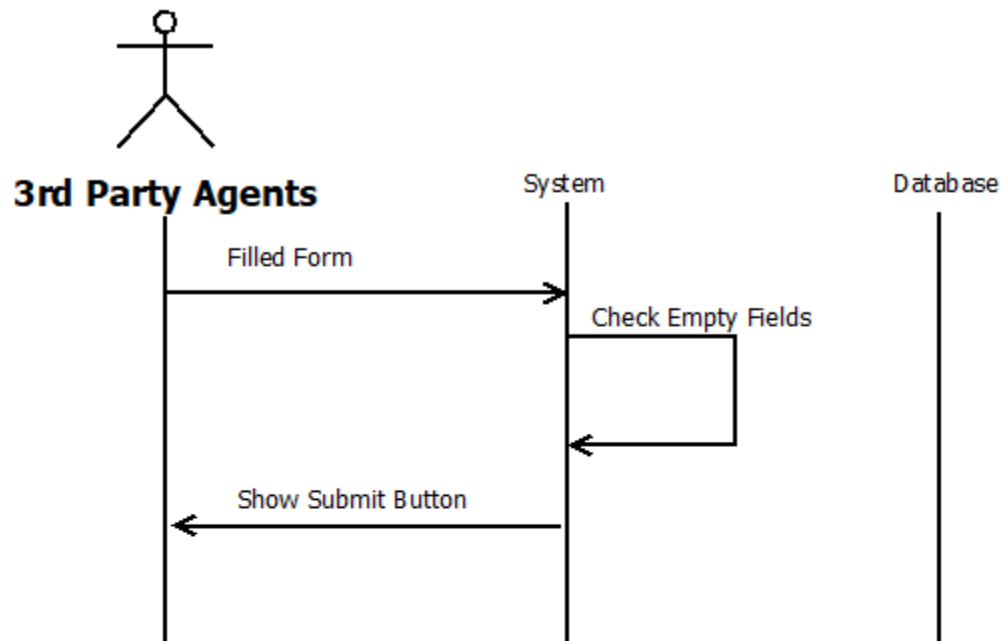


Figure 5. The sequence diagram for UC-1

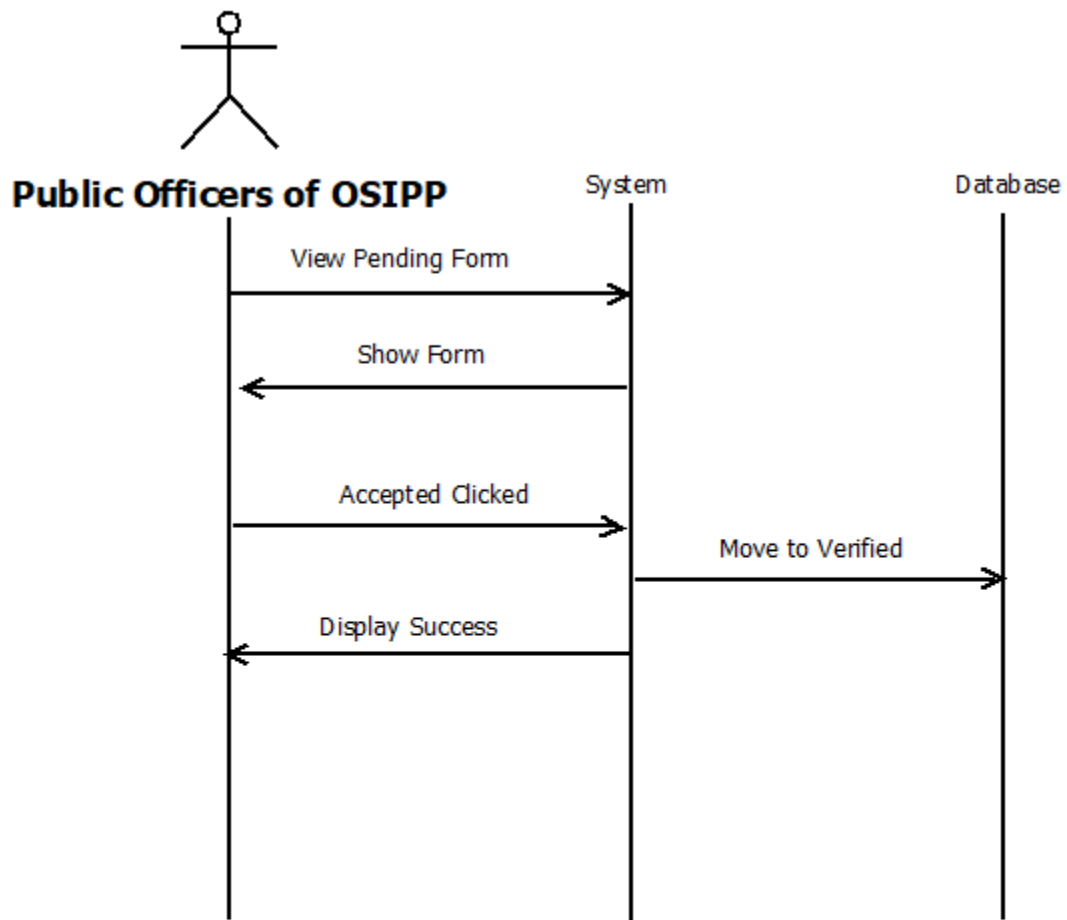


Figure 6. The sequence diagram for UC-2

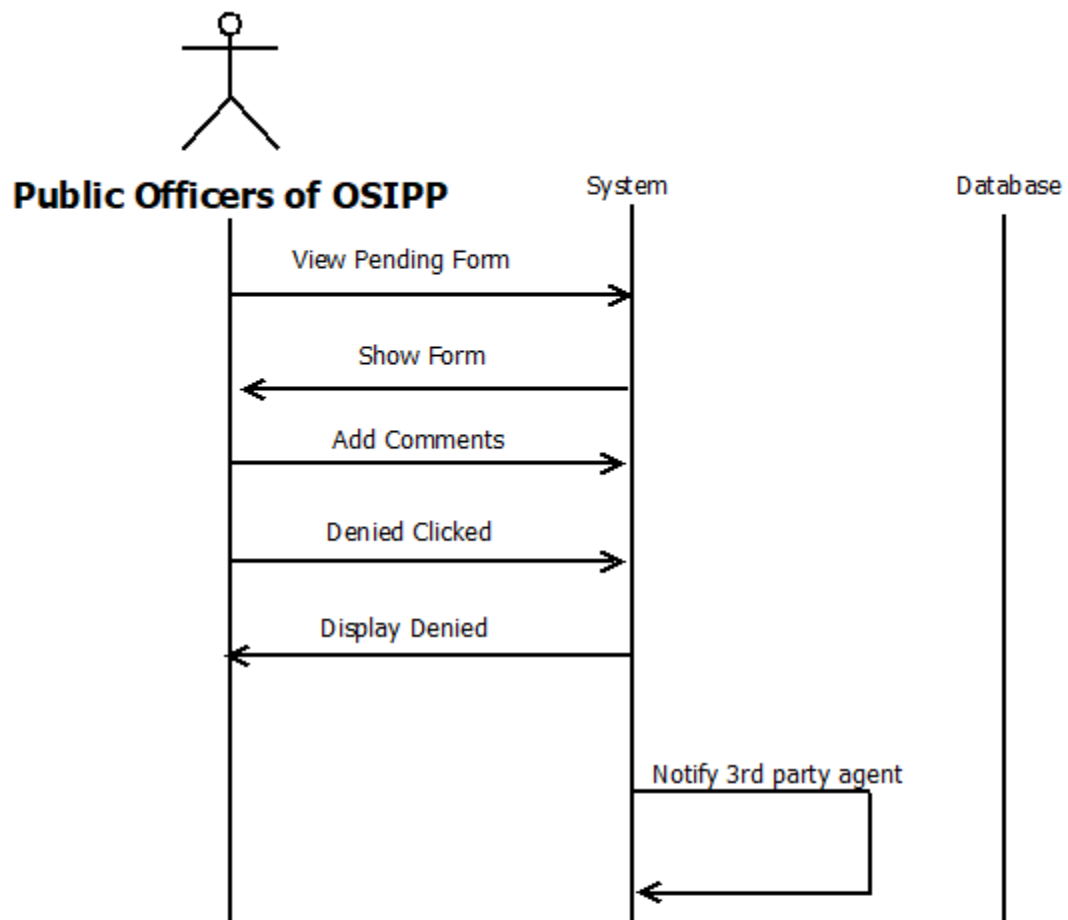


Figure 7. The sequence diagram for UC-3

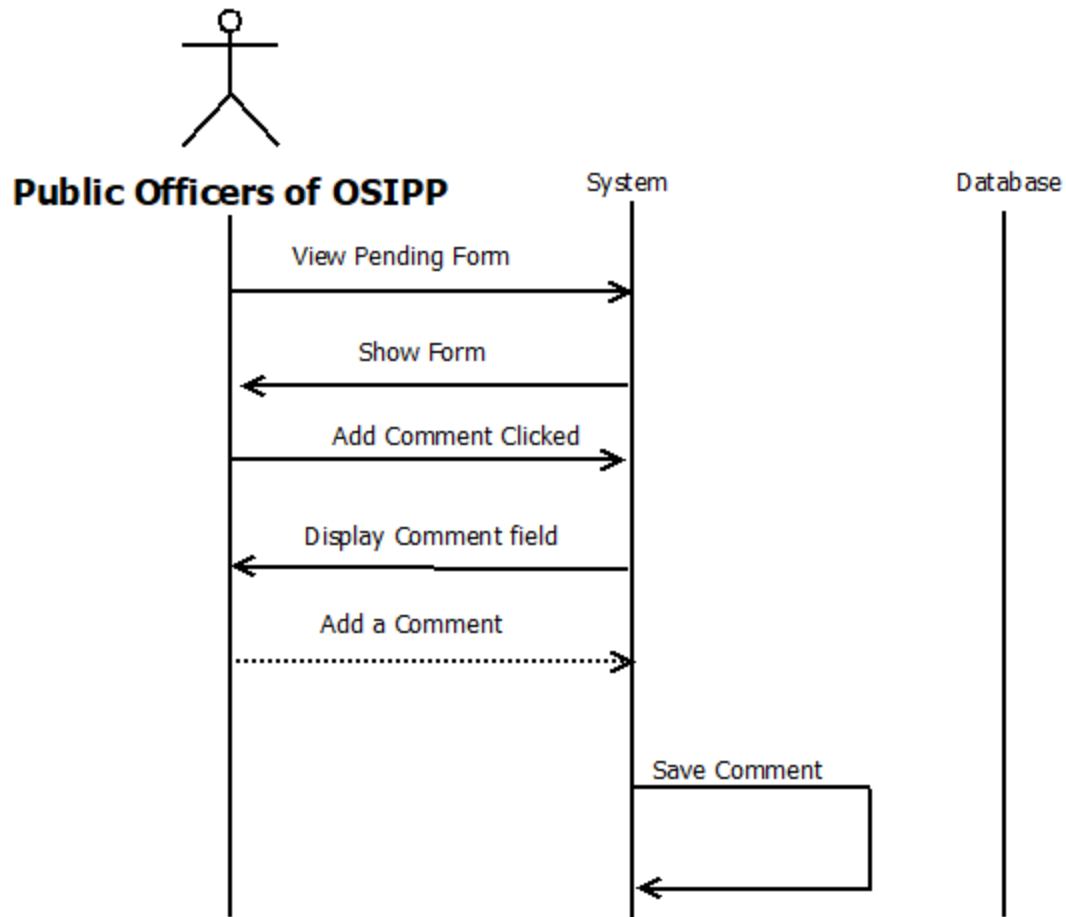


Figure 8. The sequence diagram for UC-4

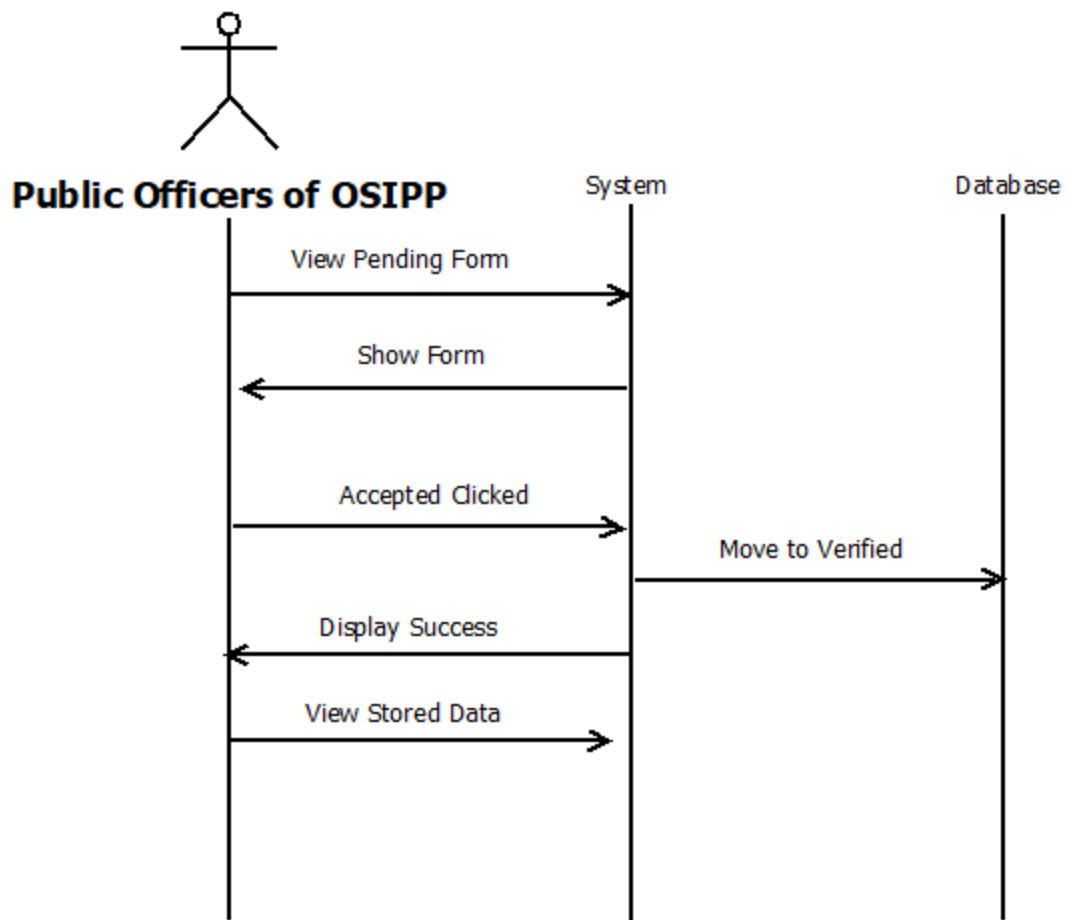


Figure 9. The sequence diagram for UC-5

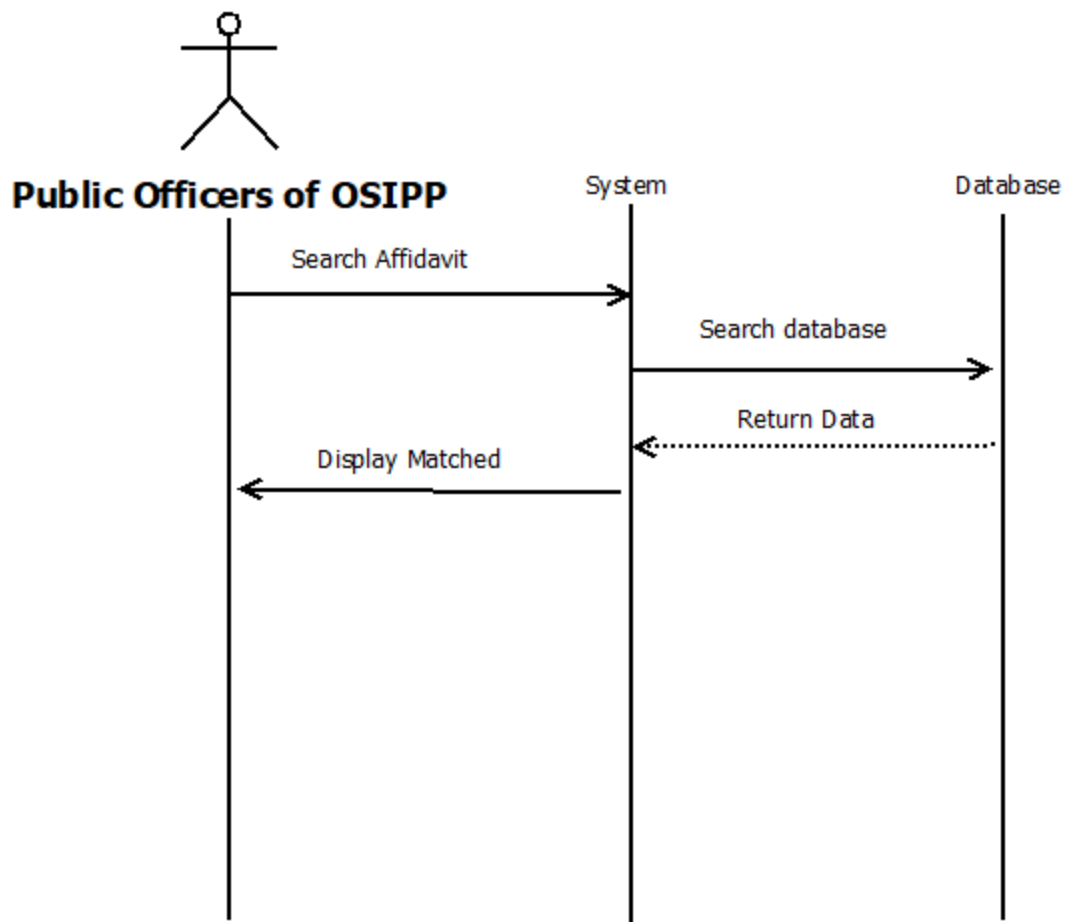


Figure 10. The sequence diagram for UC-6

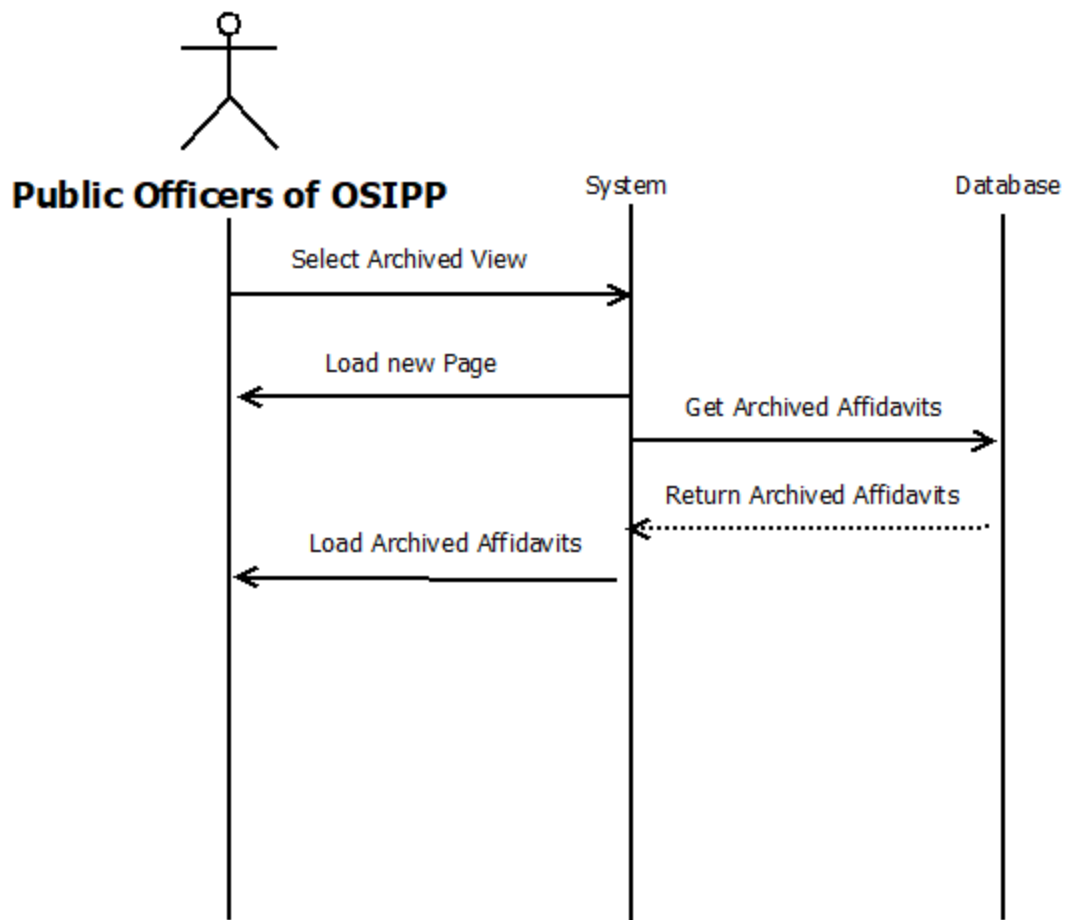


Figure 11. The sequence diagram for UC-7

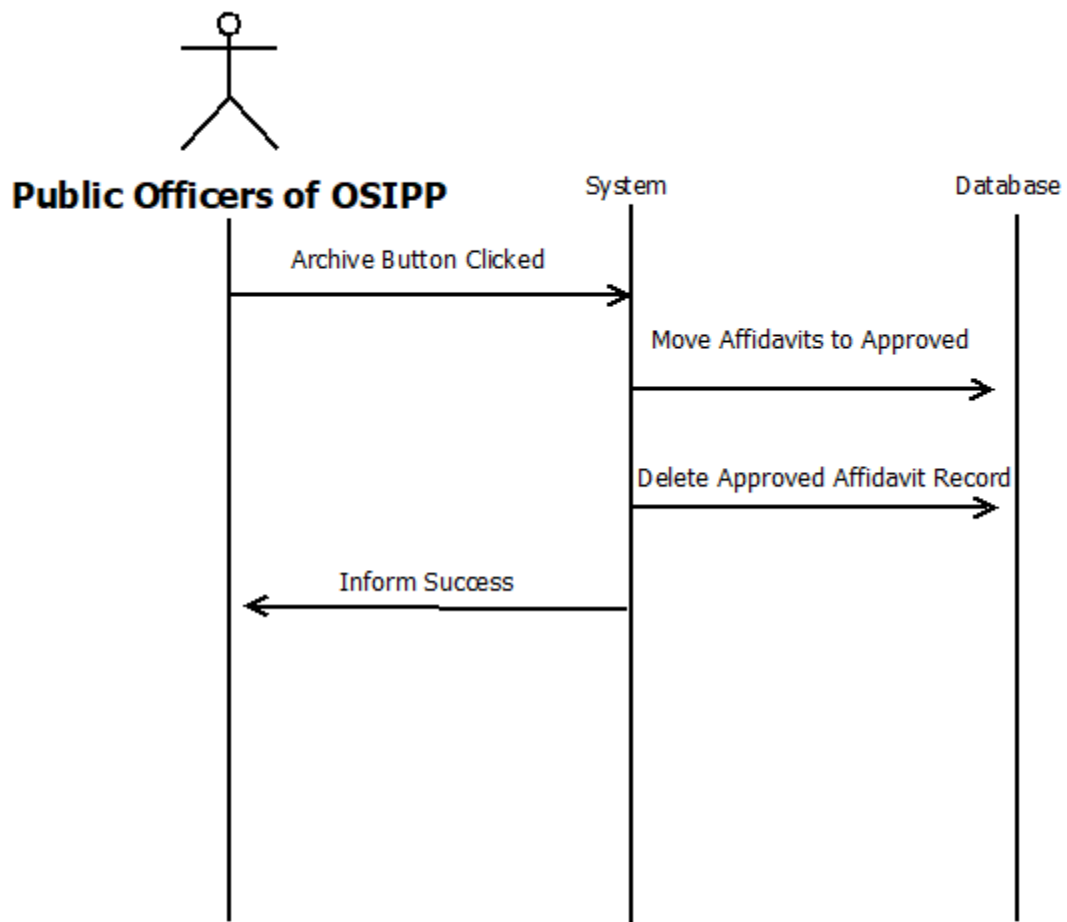


Figure 12. The sequence diagram for UC-8

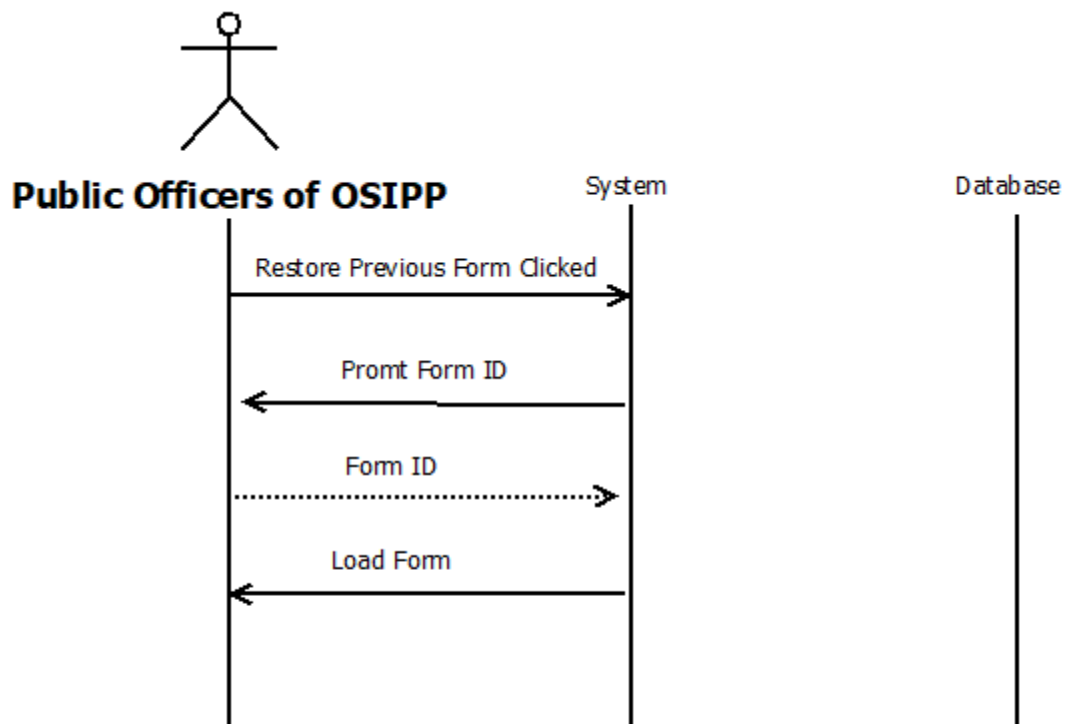


Figure 13. The sequence diagram for UC-9

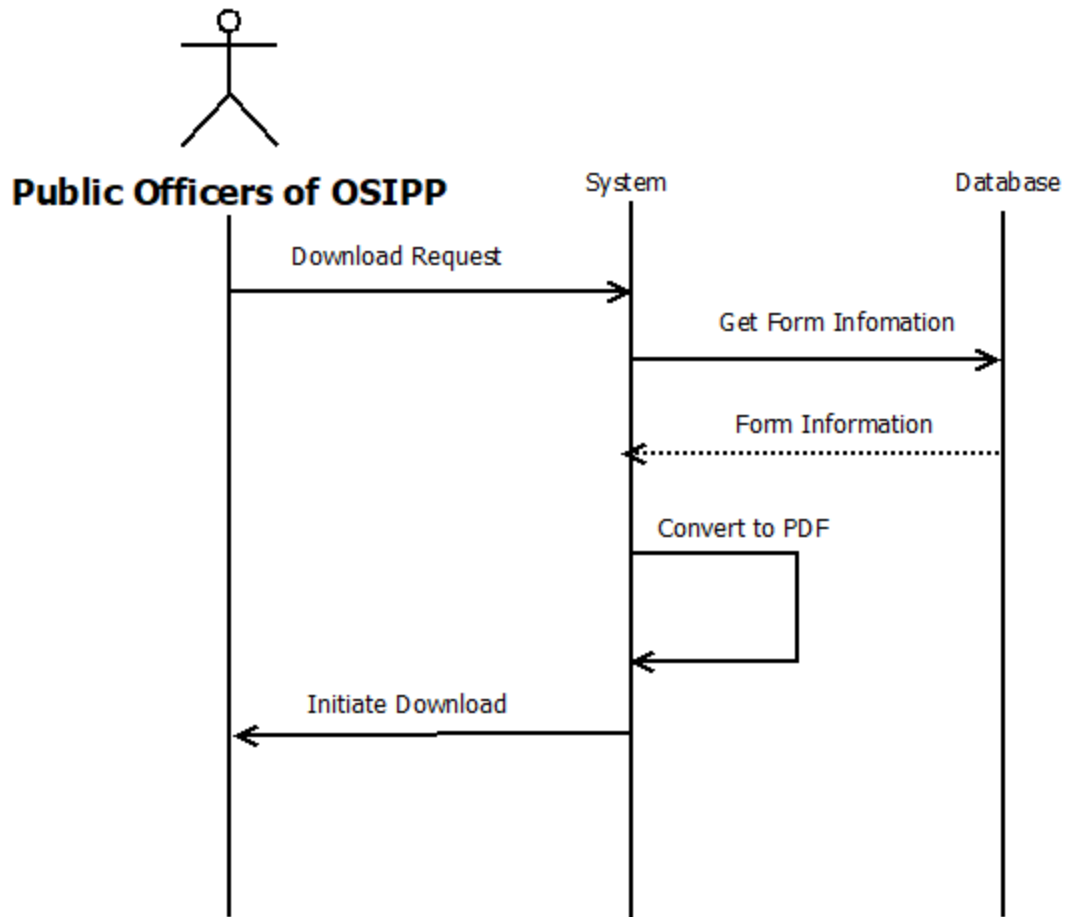


Figure 14. The sequence diagram for UC-10

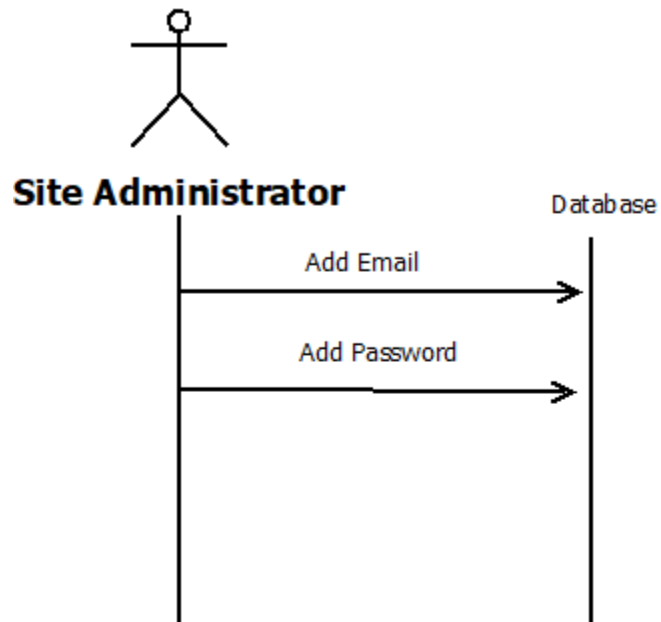


Figure 15. The sequence diagram for UC-11

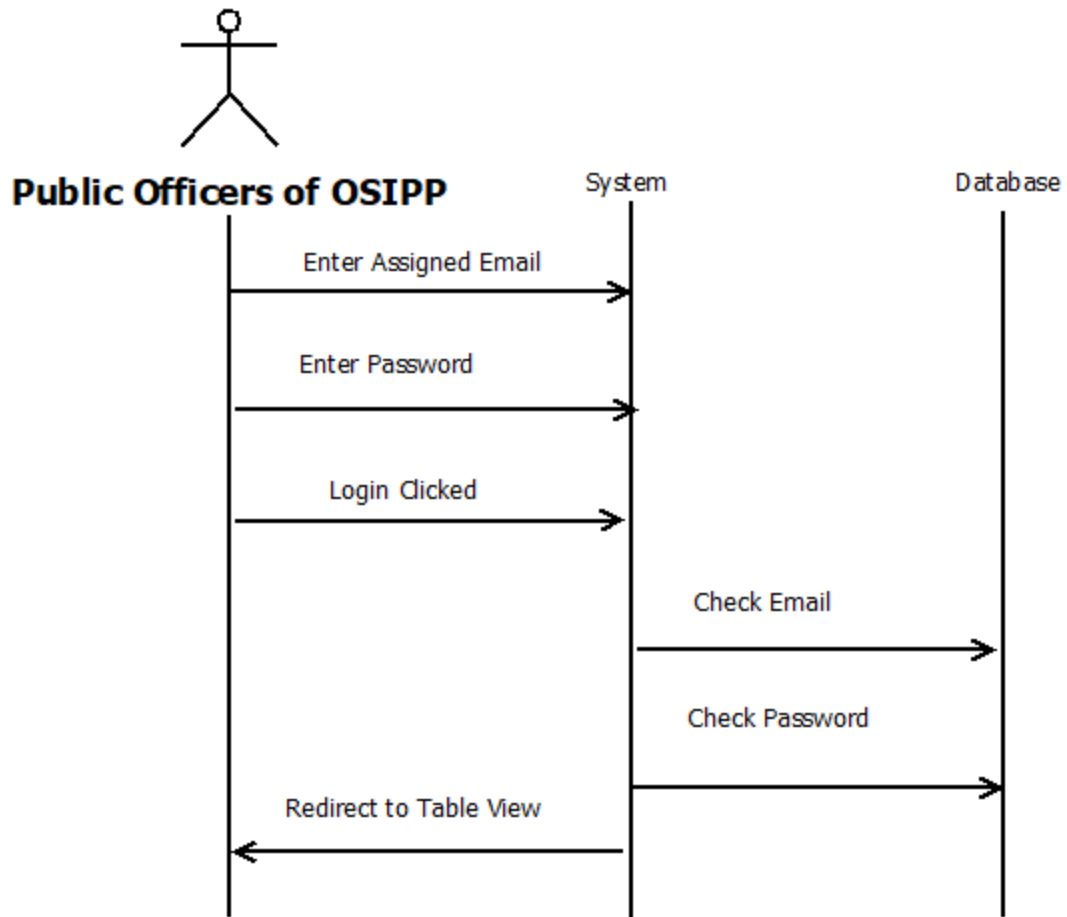


Figure 16. The sequence diagram for UC-12

INTERFACE SPECIFICATION

Preliminary Design:

- **UC1-FillingForm**

All fields in the form must be filled in order for the system to correctly capture all fields. The system will check all the fields, verifying none are missing before the 3rd party agent can submit. The picture below shows an illustration of the Use case.



OFFICE OF THE SUPERVISOR OF
INSURANCE & PRIVATE PENSION



User

Form ID: 2045783820

BIOGRAPHICAL AFFIDAVIT FORM

1. Affiant's Full Name:

2. Other names used at any time

3. Have you ever had your name changed? ☐ Yes ☒ No
If "Yes", (a) provide previous name(s) and (b) give the reason for the change.

4. Affiant's Identification No. applied to Government Record Systems. Two certified copies of picture ID must be submitted. The documents must be current and valid.

Document	Number of Document	Date of issuance	Country of issuance
Social Security	<input type="text" value="23262988"/>	<input type="text" value="2/15/21"/>	<input type="text" value="Belize"/>
Passport	<input type="text" value="P00035684"/>	<input type="text" value="10/28/20"/>	<input type="text" value="Belize"/>
National Health Insurance	<input type="text" value="N/A"/>	<input type="text" value="N/A"/>	<input type="text" value="N/A"/>
Other, specified	<input type="text"/>	<input type="text"/>	<input type="text"/>

5. Date of Birth: (DD/MM/YYYY)

6. Place of Birth: include District/State and Country

7. Nationality: Indicate how acquired.
☒ Birth
☐ Naturalization
☐ Marriage
☐ Other, specify

8. Spouse's Name:

9. Affiant's Address:

Physical Address	Telephone No.	Fax No.	Email Address
Residential	<input type="text" value="501-223-4567"/>	<input type="text" value="N/A"/>	<input type="text" value="johndoe@gmail.com"/>
Business	<input type="text" value="N/A"/>	<input type="text" value="N/A"/>	<input type="text" value="N/A"/>

Signature:

Date:

SUBMIT

Figure 10. Shows required fields when filling form

- **UC2-AcceptingForm**

The public office of OSIPP will review the form submitted by the 3rd party agent. The public officer will click on “Accept” if the form has the correct information. The system will then move the form from unverified(Pending) to verified database.

Form Review & Approval

Admin

Pending

Date Submitted	ID	Name	Status
2/03/23	2045783820	John Doe	Pending
2/13/23	2039305732	Jane Doe	Pending
1/25/23	2028912516	Sam Doe	Pending
12/14/23	2043829064	Doe Doe	Pending

Form Review & Approval

ID: 2045783820

Admin

Comment

- Affiant's Full Name: John Doe
- Other names used at any time:
- Have you ever had your name changed? ☐ Yes ☒ No
If "Yes", (a) provide previous name(s) and (b) give the reason for the change:
- Affiant's Identification No. applied to Government Record Systems. Two certified copies of picture ID must be submitted. The documents must be current and valid.

Document	Number of Document	Date of issuance	Country of issuance
Social Security	23042988	2/15/21	Switzerland
Passport	FP00035684	10/28/20	Switzerland
National Health Insurance	N/A	N/A	N/A
Other, specified			
- Date of Birth: 12/03/1995 (DDMMYYYY)
- Place of Birth: Zurich (include District/State and Country)
- Nationality: Swiss (Indicate how acquired)
 - ☒ Birth
 - ☐ Naturalization
 - ☐ Marriage
 - ☐ Other, specify:
- Spouse's Name:
- Affiant's Address: 104 River street

Physical Address	Telephone No.	Fax No.	Email Address
Residential	901.023.4307	N/A	john.doe@gmail.com
Business	N/A	N/A	N/A

Signature: [Signature] Date: 2/03/23

Accept Decline

Form Review & Approval

Admin

Pending

Verified

Date	ID	Name	Status
2/03/23	2045783820	John Doe	verified
2/13/23	2039305732	Jane Doe	Verified
1/25/23	2028912516	Sam Doe	Verified
12/14/23	2043829064	Doe Doe	Verified

Figure 11. Shows steps taken by the public officer of OSIPP to accept a form

- **UC5- StoringData**

The public officer will click on the “Approve” button that will be available to him if the form is correctly filled out. After approval the system will display a “success” message confirming the form has been stored successfully in the database.

Form Review & Approval
ID: 2045783820

1. Affiant's Full Name: John Doe

2. Other names used at any time:

3. Have you ever had your name changed? ☐ Yes ☒ No
If "Yes", (a) provide previous name(s) and (b) give the reason for the change:

4. Affiant's Identification No. applied to Government Record Systems. Two certified copies of picture ID must be submitted. The documents must be current and valid.

Document	Number of Document	Date of issuance	Country of issuance
Social Security	2302998	2/13/21	Belize
Passport	P00035684	10/28/20	Belize
National Health Insurance	N/A	N/A	N/A
Other, specified			

5. Date of Birth: 12/10/1985 (DDMMYYYY)

6. Place of Birth: Belize (Include District/State and Country)

7. Nationality: Belizean
☒ Birth
☐ Naturalization
☐ Marriage
☐ Other, specify:

8. Spouse's Name:

9. Affiant's Address: 704 River street

Physical Address	Telephone No.	Fax No.	Email Address
Residential	501-223-4567	N/A	john.doe@gmail.com
Business	N/A	N/A	N/A

Signature: Date: 2/03/23

Approve **Decline**

Form Review & Approval

	Date	ID	Name	Status
Pending				
Verified				
Approved >	2/03/23	2045783820	John Doe	Approved
Archived				
	2/13/23	2039305732	Jane Doe	Approved
	1/25/23	2028912516	Sam Doe	Approved
	12/14/23	2043829064	Doe Doe	Approved

Figure 12. Shows steps taken by the public officer of OSIPP to approve a form

- **UC11-Making Account**

Site Administrators will be creating an OSIPP officer's email address and password in order for them to successfully access the forms with admin privileges.



Create Account
Username

Email

Password

User Effort Estimation:

- Scenario: Filling Form
Navigation: 26 clicks
 - Click on affidavit's fullname field
 - Click on radio button on field question
 - Click on number of document field
 - Click on date of insurance field
 - Click on country of insurance field
 - Click on date of birth field
 - Click on place of birth field
 - Click on Nationality field
 - Click on radio button based on nationality
 - Click on affiant's address field
 - Click on telephone number field
 - Click on fax no. field
 - Click on email address field
 - Click on signature field
 - Click on date field
-
- Data entry: 26 fields (keystrokes depend on the filling form, so it is difficult to estimate)
 - Add affidavit's full name
 - Choose option field question
 - Add number of document
 - Add date of insurance
 - Add country of insurance
 - Add data of birth field
 - Add place of birth field
 - Add nationality field
 - Choose option based on nationality
 - Add affiant's address
 - Add email address
 - Add signature
 - Add date

- Scenario : AcceptingForm

Navigation: 1 click

Click on Accept button

- Data entry: No fields
No data entry
-

- Scenario: StoringData

Navigation: 1 click

Click on Submit button

- Data entry: No fields
No data entry
-

- Scenario: SubmittingForm

Navigation: 1 click

Click on Submit button

- Data entry: No fields
No data entry
-

- Scenario: Making Account

Navigation: 5 clicks

Click on username field

Click on email address field

Click on password field

Click on password confirmation field

Click on register button

-
- Data Entry: 4 fields

Enter username

Enter email address

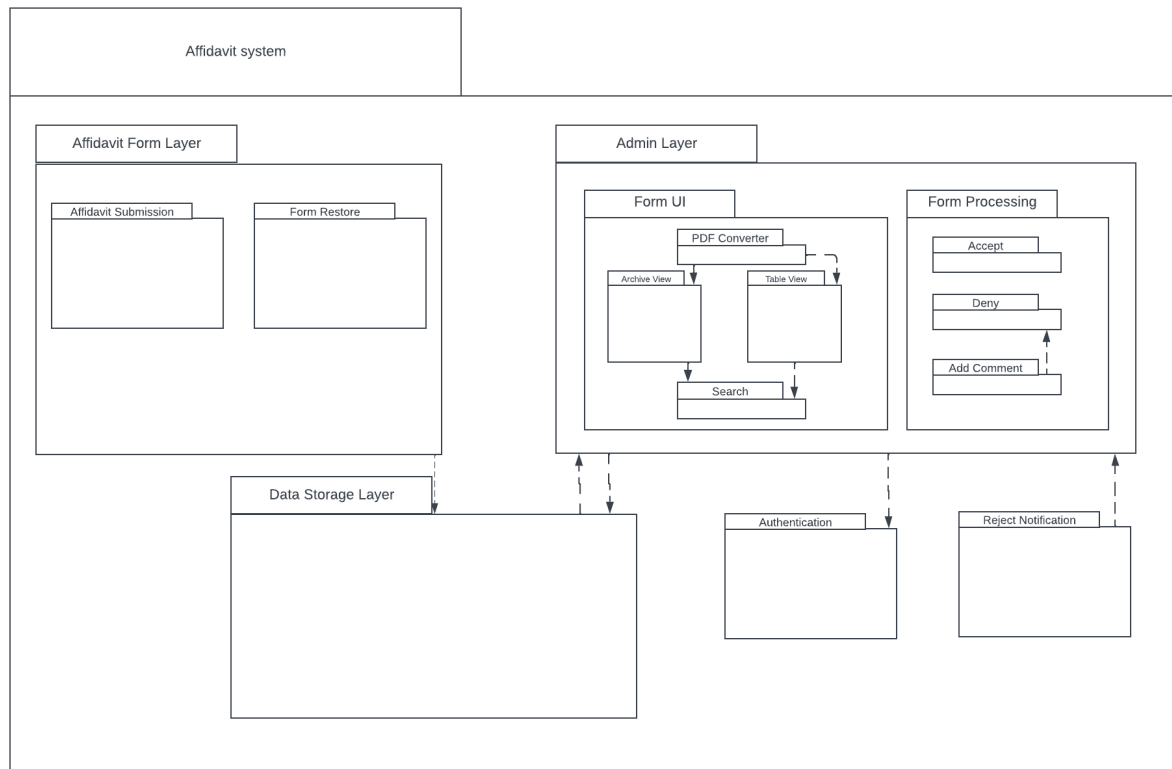
Enter password

Enter confirmation password

SYSTEM ARCHITECTURE

Identifying Subsystems & Architecture Styles:

UML Package Diagram



+

The UML Package diagram above is a high-level package view of the Affidavit system. Our system is made up of 5 Logical packages. Our first package is the Affidavit Form layer which has two subsystems. One is for Affidavit submission and one takes care of the Form Restoration. This package has one dependency, the Data storage layer because it needs to access data in order to restore the form based on its ID or to send the submitted form.

The Admin layer has two sub-packages named Form UI and Form Processing. Form UI houses our PDF converter, Archives view, Table view, and Search packages. Archives and tables both have a dependency on search. PDFconverter has dependencies on both the Archives view and Tables view. Form Processing takes care of Accepting, denying, and adding comments to the form. Adding comments has a dependency on Deny. The Admin layer has dependencies with Data storage and with Authentication.

Data storage has a dependency on the Admin layer. Lastly, our Rejection notification package has a dependency on the Admin UI since it cannot be executed without the form being denied.

Architecture Styles:

The software architectural style employed in the design is a three-tier (Client-Server) architecture: presentation, application, and data. The presentation tier includes Admin, Market, and Vendor UI, which displays information about services like login, form data entry, and verification. The application tier includes the ability to generate reports or search 3rd party agent details, as well as view which forms are incomplete, completed, or canceled. Finally, the data tier would include local server storage of third-party agent details. REST is an abbreviation for Representational State Transfer, and API is an abbreviation for Application Program Interface. REST is a software architectural style that specifies the rules for developing web services. RESTful web services are web services that adhere to the REST architectural style. It enables requesting systems to access and manipulate web resources through the use of a consistent and predefined set of rules. REST-based systems communicate using the Internet's Hypertext Transfer Protocol (HTTP).

Mapping Subsystems to Hardware:

The system will be housed on a server at the OSIPP office, filling out the form, processing the form and accessing the information will all be done via a client. All the the different clients will be serviced by the server using an API to communicate the necessary functionality. We will be using Linux based Cent OS, Apache, PHP, and MySQL.

Connectors and Network Protocols:

Connectors → Physical connection CAT5 wire for data transfer

Network Protocols → include HTTP(HyperText Transfer Protocol) used to display the appropriate page and FTP(FileTransferProtocol) which allows data to be inserted into the database. Also,

Global Control Flow:

Execution Order

Our biographical affidavit form web application is, for the most part event-driven, meaning that in order for any action to be taken by the system, an event must be detected to initiate the action. The events are mainly triggered by the users of the system; for instance, filling out a form depends entirely on the 3rd party agent's interaction with the system to submit. Most object states within the system depend entirely on event-driven updates. Before a submitted form can be reviewed, the OSIPP public office will first need an account; only then can submissions be accepted, approved, archived, and unachieved.

Time Dependency

The system we are creating is one that will have multiple people interacting with it, inputting and verifying sensitive information. Due to this we will not be having any timer on our system. 3rd party agents will be able to take as much time to fill out a form in order for mistakes not to occur, OSIPP public officers will also have plenty of time when reviewing forms to make sure all required information is available. As far as the system goes it will not have any concerns with real time other than the form submission and approval date.

Hardware Requirements:

Recommended → Hosting a web server to fill out forms requires minimal power so we have decided to run on a 2 core 2GB RAM 40GB HDD bandwidth 1Mbps

Minimum Requirements → 1 Core 1GB RAM, 20GB HDD Bandwidth 56 kbps

PLAN OF WORK

This project, besides being a part of a course at the University of Belize is also an actual Commission from The Supervisor of Insurance at OSIPP. Therefore it's essential that our system has its core functionality on deployment. The process of the system's development is as follows:

1. Finalizing the interface which the system will use along with the initial versions that'll be used during its development
2. Creating a functional web server that will be able to facilitate all functionality needed by the system.
3. Designing the database that'll be used from deployment till end of life; along with the CRUD(Create, Read, Update, Delete) functions necessary.
4. Ensuring that the system fulfills all requirements and aesthetics of OSIPP before deployment
5. Maintenance of the system past deployment if needed

All members of the group will contribute to the code base of the system. However, following the interests and skills of the group members. Parts of the system will be the primary focus for each group member respectively. The reports, just like the system, will have participation from all group members. Ensuring that all members have a full understanding of the system, it's functionality, and the problem domain it's being applied to.

Gantt Chart for the development of the System:

Task	Timeline									
	March						April			
	1	8	10	17	22	29	12	19	26	28
Finalizing the design of the Frontend	1 Day									
Creation of Codebase, Github Repository, and assignment of roles and targets for group members		7 Days								
Webserver creation, Methods and Member functions, Starting GUI development			2 Days							
Implementation of CRUD functions, connecting Frontend and Backend				6 Days						
Demonstration preparation and debugging					5 Days					
1st Demonstration of system						7 Days				
Revision of the code base, re-assignment of roles and targets							14 Days			
Finalizing all functionality and GUI interactions with backend								7 Days		
Final debugging and testing of system									7 Days	
Finalizing of system for final demonstration										2 Days

REFERENCES

Fielding, R. T. (2023, February 20). *Wikipedia*. Retrieved from Representational state transfer: https://en.wikipedia.org/wiki/Representational_state_transfer

Government of Belize. (2023, February 22). *Office of the Supervisor of Insurance and Private Pension*. Retrieved from Office of the Supervisor of Insurance and Private Pension: <https://osipp.gov.bz>

Geeks for Geeks. (2023, January 5). *REST API Architectural Constraints*. Retrieved from geeks for geeks: <https://www.geeksforgeeks.org/rest-api-architectural-constraints/>

Ministry of Finance. (2023, February 24). *Ministry of Finance*. Retrieved from Government of Belize: <https://www.mof.gov.bz>



Biographical Affidavit

Report #2

Group 1:

Hipolito Bautista

Justin Chuc

Michael Gomez

Rene Allen

Dennis Arnold

Jahmur Lopez

CMPS4131

Manual Medina

March 31,2023

TEAM SIGNATURES

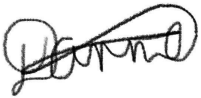



SIGNATURE BLOCK			
Statement	I did my share of the work, and I have a general understanding of the contents of the assignment		
Team Member	Contribution	Signature	Date
Dennis Arnold			26/2/2023
Jahmur Lopez			26/2/2023
Justin Chuc	•	<i>J. Chuc</i>	26/2/2023
Hipolito Bautista	•		26/2/2023
Michael Gomez	•		26/2/2023
Rene Allen	Conceptual Model Concept Definitions Class Diagram and Interface Specifications	<i>Rene Allen</i>	26/2/2023

Table of Contents

TEAM SIGNATURES	2
Table of Contents	3
INDIVIDUAL CONTRIBUTION BREAKDOWN	6
ANALYSIS AND DOMAIN MODELING	7
Conceptual Model:	7
Attributes Definitions	14
Traceability Matrix	15
System Operation Contracts	15
Data Models & Persistence storage	16
For this project we do need persistent data storage. The data we need stored will be the form inputs given to us by the user. We will do this by storing it in a table within our database named "Form". We also need Admin and User data which have their own tables. We will be tracking the history of the admins making comments for future use with it's own table in the database. Lastly we will need storage for all archived forms (older forms) for long term storage. There is a specific "archive" table which will store all old forms.	16
Data Dictionary	17
Interaction Diagrams	18
Class Diagram and Interface Specifications	31
Traceability Matrix	33
Algorithms	34
The system will primarily consist of a sorting algorithm in order to group forms that external users have already filled that are linked with its form ID and user ID. The algorithm will be responsible to group forms based on its status if its brand new, pending for verification, has been verified or returned for error correction. In addition, the API will be responsible for verifying what type of users will be logging in to the system.	34
The system will be in charge of granting certain users access whether it be an admin, 3rd-party users or OSIPP officers depending on what data is passed to the API. When the system retrieves the API response, the system will present a welcome message along with the option to either create a new form, view status or make corrections.	34
Algorithm:	
3rd-party user logs into Biographical Affidavit:	
If "user credentials" is true	
Display message "Welcome %sFirstName"	
If else "user credentials" is false	
Notify user that either username or password incorrect	
While "user credentials" is true	
Display message "Welcome %sFirstName"	34
Set form status to "new"	
User fills out fields in the form and its submitted	
Set form status to "pending"	
If "Affidavit form" is incorrect	
Return form back to user	

Set form status to “returned”
User verifies and re-submit form
Set form status as “pending”
If “Affidavit form” passes verification
Set form status as “verified”
Transfer form to database

Concurrency

The system will allow middleware to enable communications with the application and database. Since there will be many users utilizing the software at once, event handlers will be placed to assist with the workload of interactions with the application and the users. This case applies to different users filling out the form and other users who are verifying many forms for different 3rd-party users. 35

User Interface Design and Implementation 36

Initially, the affidavit form and Review and Approval interface was designed using a UI designer. However, the current design has been implemented using HTML/CSS. As part of the implementation process, We made significant improvements to the form's usability by adding placeholders to the insertion boxes, making it less likely for users to make mistakes when filling out the form. Furthermore, we made slight changes to the Review and Approval interface by incorporating a search bar to enhance the user experience and enable quicker access to specific forms. 36

Fill form 37

- The Diagram below displays the Biographical Affidavit form 37

View Unverified 38

- The Diagram below displays the Unverified interface. 38
- The Diagram below displays the verified interface. 38

View Archive 39

- The Diagram below displays the Archive interface. 39
- The Diagram below displays the login screen for OSSIP Officers to access system 39
- The Diagram below displays a form that is being reviewed by the OSIPP officer. 42

Design of Tests 43

Test Case 1 43

Test Case 2 44

Test Case 3 44

Test Case 4 45

Test Case 5 47

Test Case 6 47

Test Case 7 48

Test Case 8 49

Test Case 9 50

Test Case 10 51

Test Case 11 51

Test Case 12 52

Test Case 13 53

Test Case 14

54

PROJECT MANAGEMENT AND PLAN OF WORK

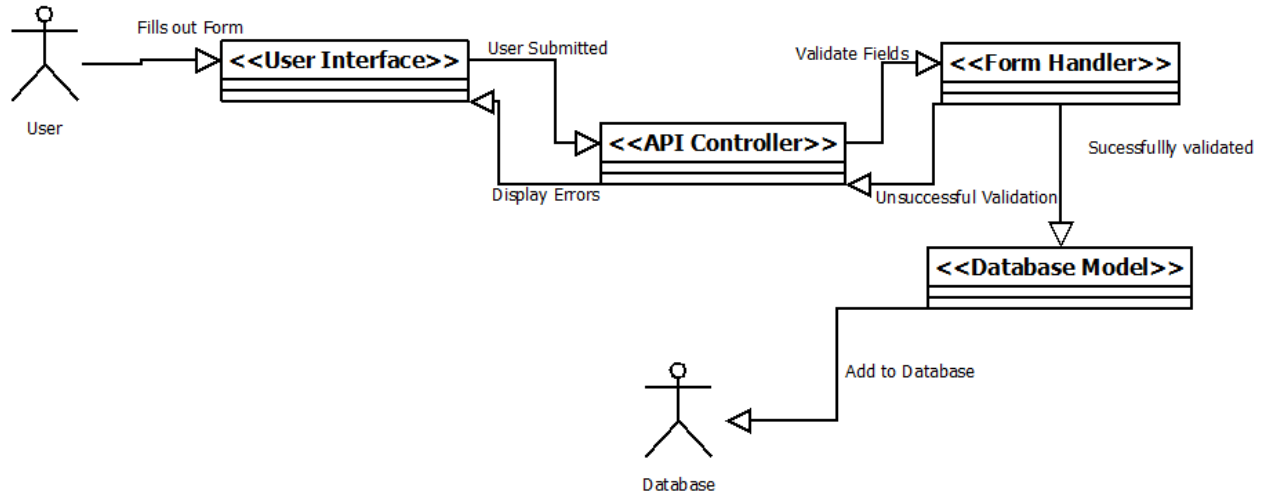
56

INDIVIDUAL CONTRIBUTION BREAKDOWN

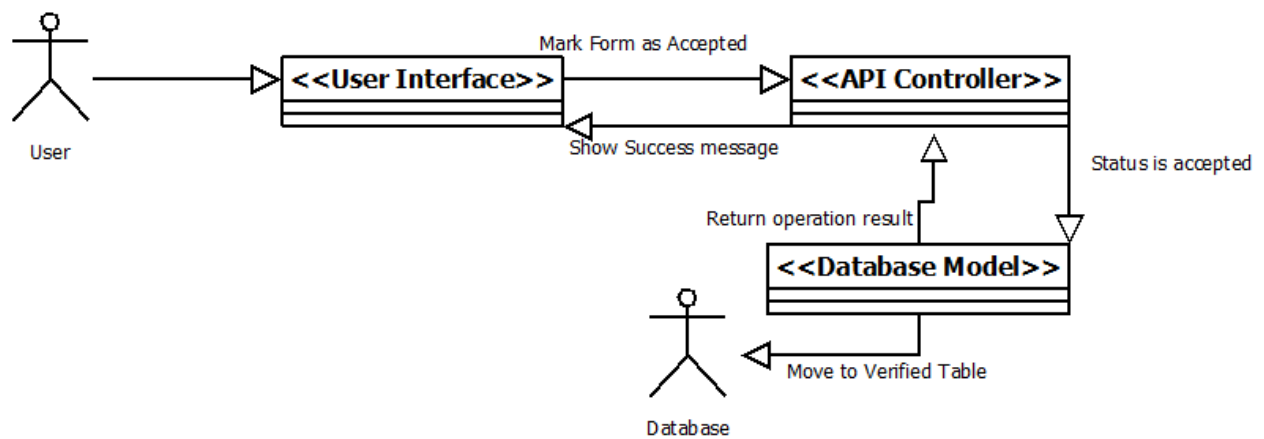
Responsibility Level	Team Member Names					
	Dennis Arnold	Jahmur Lopez	Justin Chuc	Hipolito Bautista	Michael Gomez	Rene Allen
Project Management (10 points)						
Sec. 1 Customer Statement of Requirements (9 points)						
Sec. 2 System Requirements (6 points)						
Sec. 3 Functional Requirements Specification (30 points)						
Sec. 4 User Interface Specs (15 points)						
Sec. 5 Domain Analysis (25 points)						
Sec. 6 Plan of work (6 points)						

ANALYSIS AND DOMAIN MODELING

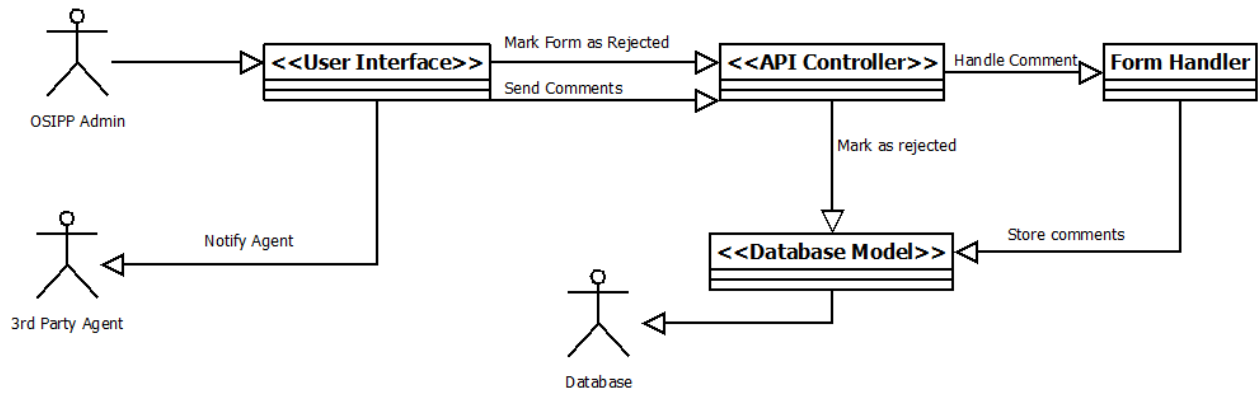
Conceptual Model:



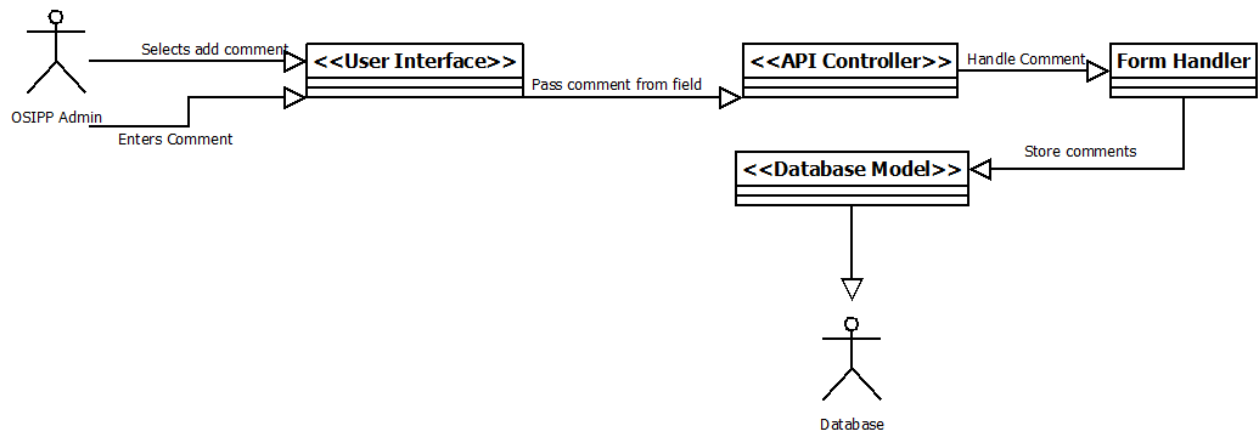
UC 1 - FillingForm



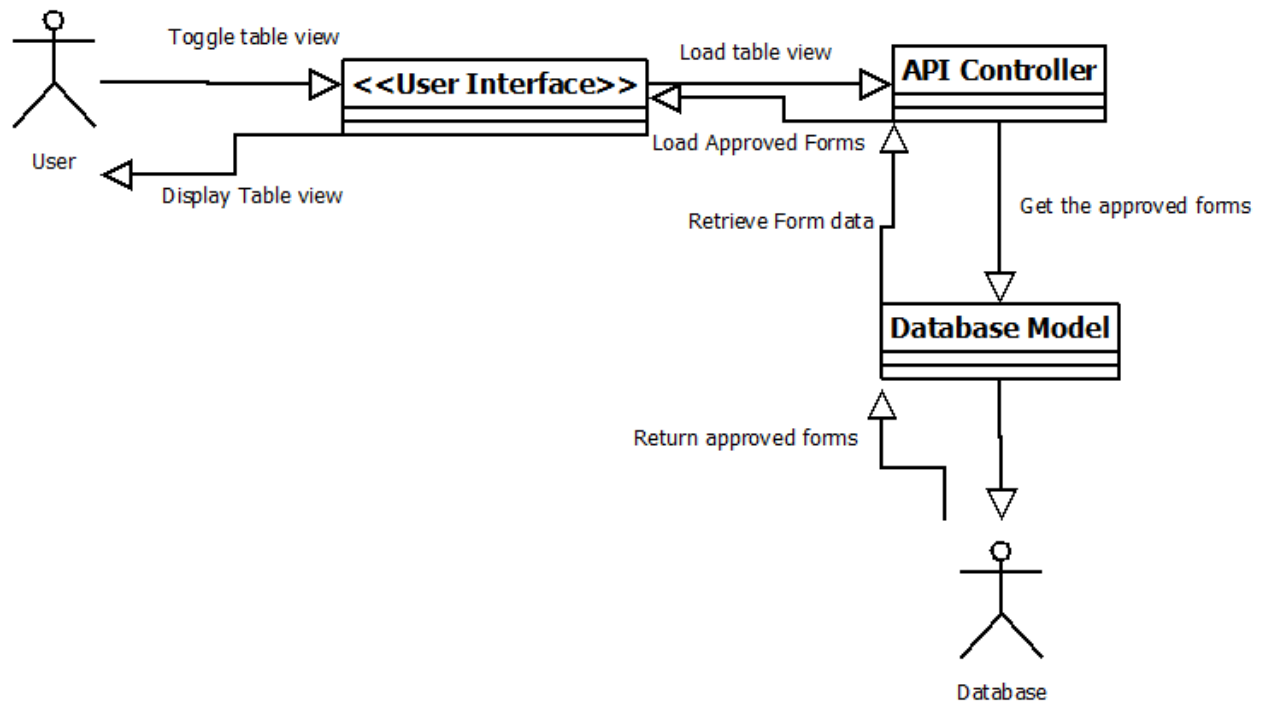
UC 2 - AcceptingForm



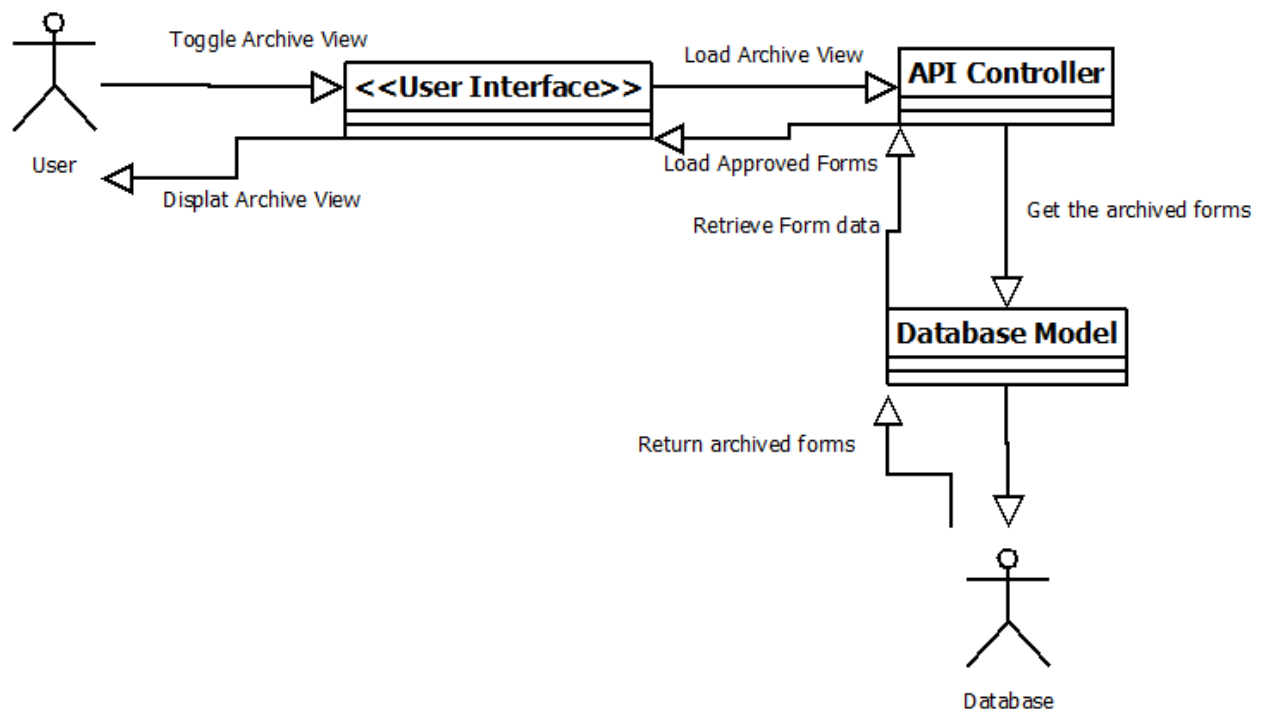
UC 3 - DenyingForm



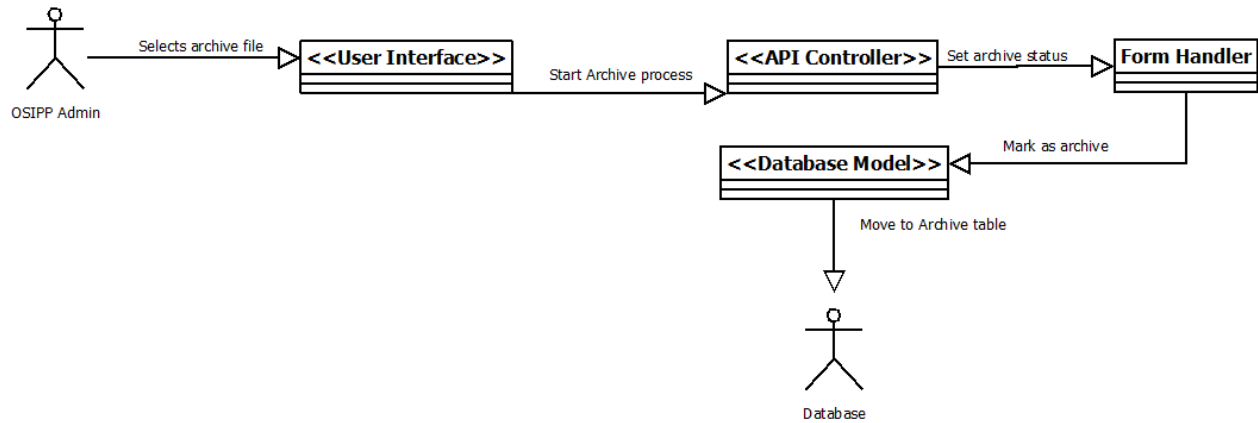
UC 4 - AddingComments



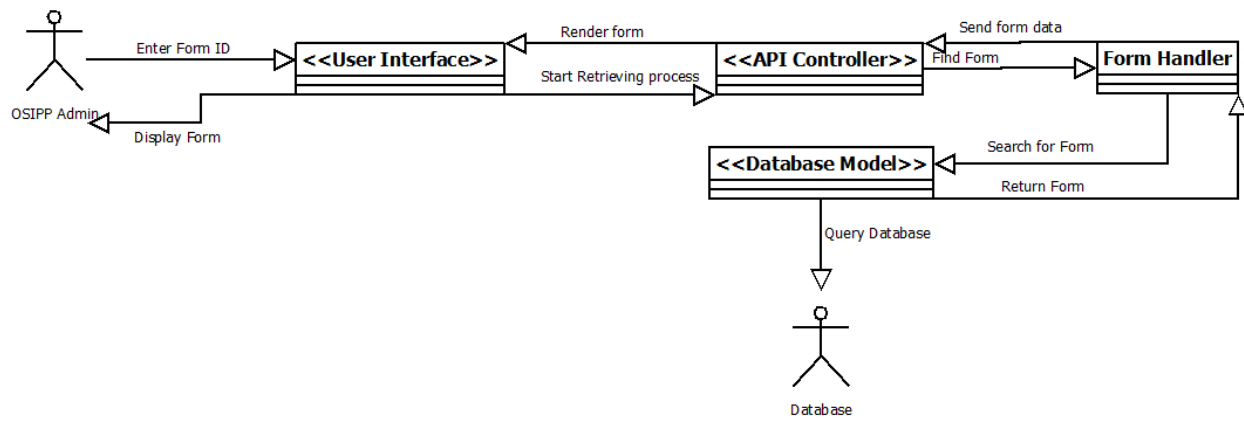
UC 5 - ToggleTableView



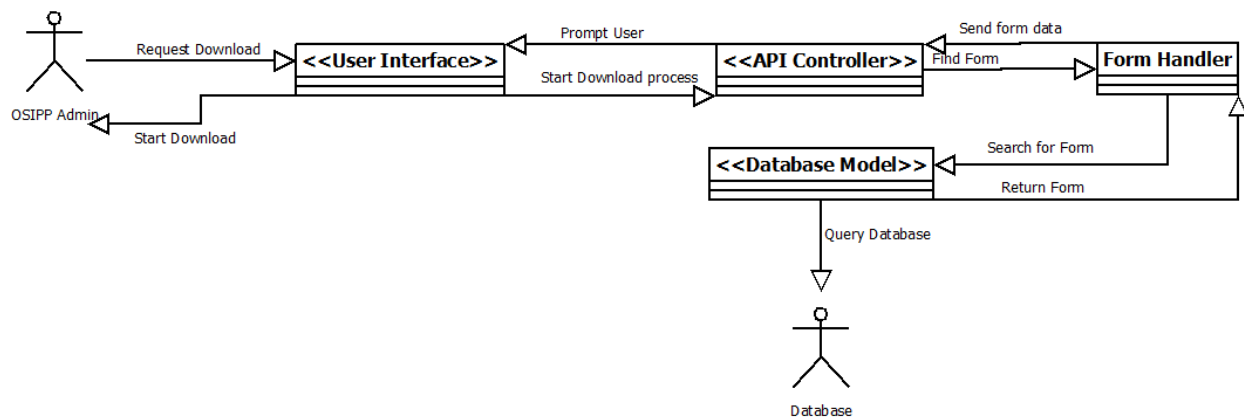
UC 6 - Archive View



UC 7 - Archiving file



UC 8 - Restoring Form



UC 9 - Downloading Form

Concept Definitions

Responsibility Description	Concept Name
Provide the necessary fields to fill out the form from the user.	User Interface
Will delegate the task to another handler. Will generate error messages to the user	API Controller
Responsible for validating the user input. Forward the reason for unsuccessful validation	Form Handler
Add the form to the pending table.	Database Model

Concept Definition for UC-1

Responsibility Description	Concept Name
Provide the graphical element for the user to mark the form as accepted Display the success message to the user	User Interface
Contacting the database to mark as accepted Return the success message to the UI	API Controller
Move the form to the verified table	Database Model

Concept Definition for UC-2

Responsibility Description	Concept Name
Provides the field to input comments	User Interface
Receive the comments Delegate to the form handler set up to mark form as rejected	API Controller
Process comments	Form Handler
Mark the form as rejected Store the comments for the form	Database Model

Concept Definition for UC-3

Responsibility Description	Concept Name
Provide a method for users to add and enter comments	User Interface
Delegate to the form handler	API Controller
Communicate with the database to store comments	Form Handler
Save the comments in the database	Database Model

Concept Definition for UC-4

Responsibility Description	Concept Name
Allow the user to toggle table view. The UI will display the table view	User Interface
Communicate with the database to retrieve all approved forms. Send approved forms to the UI	API Controller
Send approved form data to the API Controller	Database Model

Concept Definition for UC-5

Responsibility Description	Concept Name
Allow the user to archived view. The UI will display the archived view	User Interface
Communicate with the database to retrieve all archived forms. Send archived forms to the UI	API Controller
Send archived form data to the API Controller	Database Model

Concept Definition for UC-6

Responsibility Description	Concept Name
Provide the UI t the user to archive the form	User Interface
Delgate to the form handler	API Controller
Communicate with the database model to mark as archive	Form Handler
Move form to archived table	Database Model

Concept Definition for UC-7

Responsibility Description	Concept Name
Provide the interface to enter form ID Displays the result of the form ID	User Interface
Start the retrieving process Sending form data to be processed by the UI	API Controller
Requesting the form data from the database Forward the response to the controller	Form Handler
Search database for form Return form to the handler	Database Model

Concept Definition for UC-8

Responsibility Description	Concept Name
Allow the user to start the download process Start the download	User Interface
Start the retrieving process Sending form data to be processed by the UI	API Controller
Requesting the form data from the database Forward the response to the controller	Form Handler
Search database for form Return form to the handler	Database Model

Concept Definition for UC-9

Association Definitions

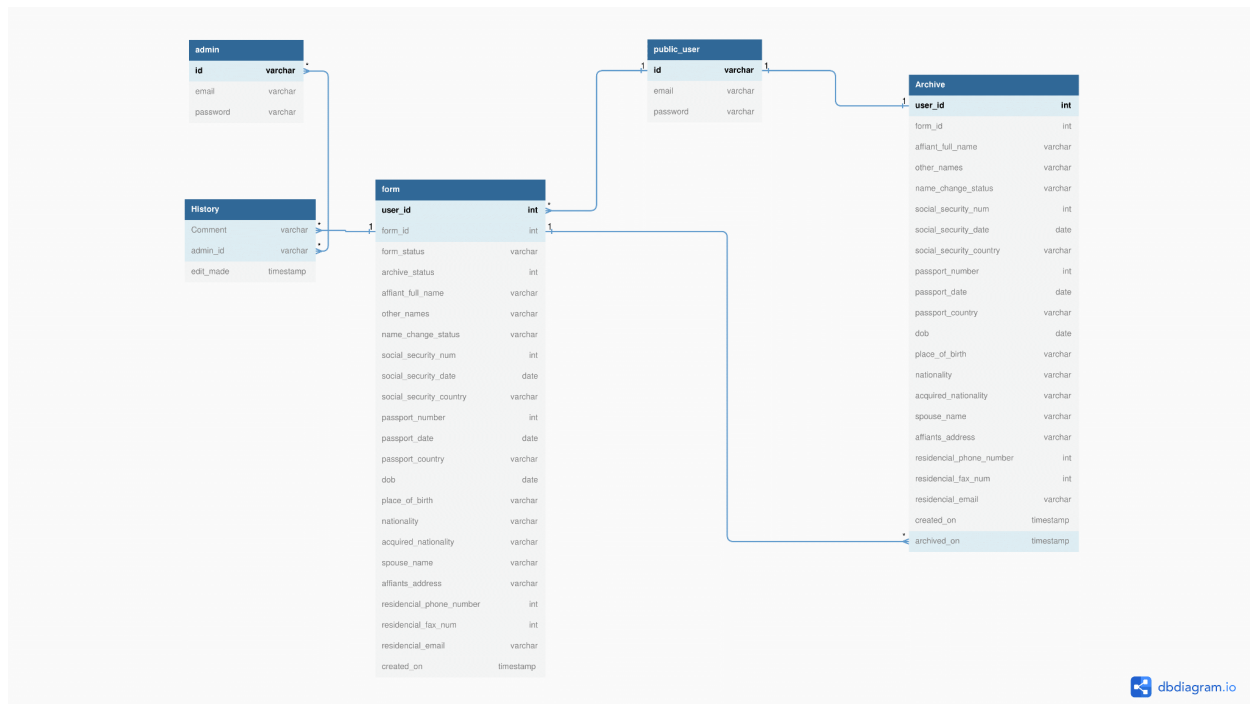
Attributes Definitions

Traceability Matrix

	UC 1 - FillingForm	UC 2 - AcceptingForm	UC 3 - DenyingForm	UC - 4 AddingComments	UC 5 - ToggleTableView	UC 6 - View Archive	UC 7 - ArchivingFile	UC 8 - RestoringForm	UC 9 - DownloadForm
User Interface	X	X	X	X	X	X	X	X	X
Controller	X	X	X	X	X	X	X	X	X
Form Handler	X		X	X		X		X	X
Database Model	X	X	X	X	X	X	X	X	X

System Operation Contracts

Data Models & Persistence storage

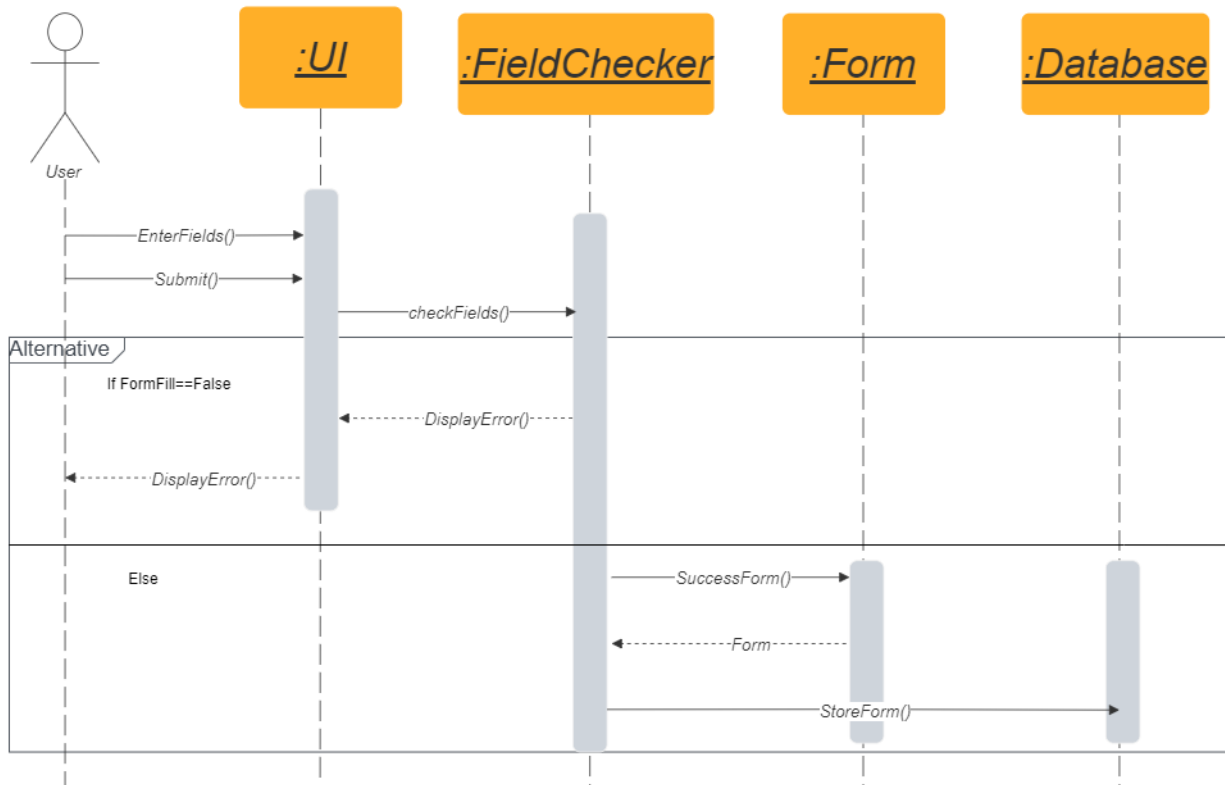


For this project we do need persistent data storage. The data we need stored will be the form inputs given to us by the user. We will do this by storing it in a table within our database named “Form”. We also need Admin and User data which have their own tables. We will be tracking the history of the admins making comments for future use with it’s own table in the database. Lastly we will need storage for all archived forms (older forms) for long term storage. There is a specific “archive” table which will store all old forms.

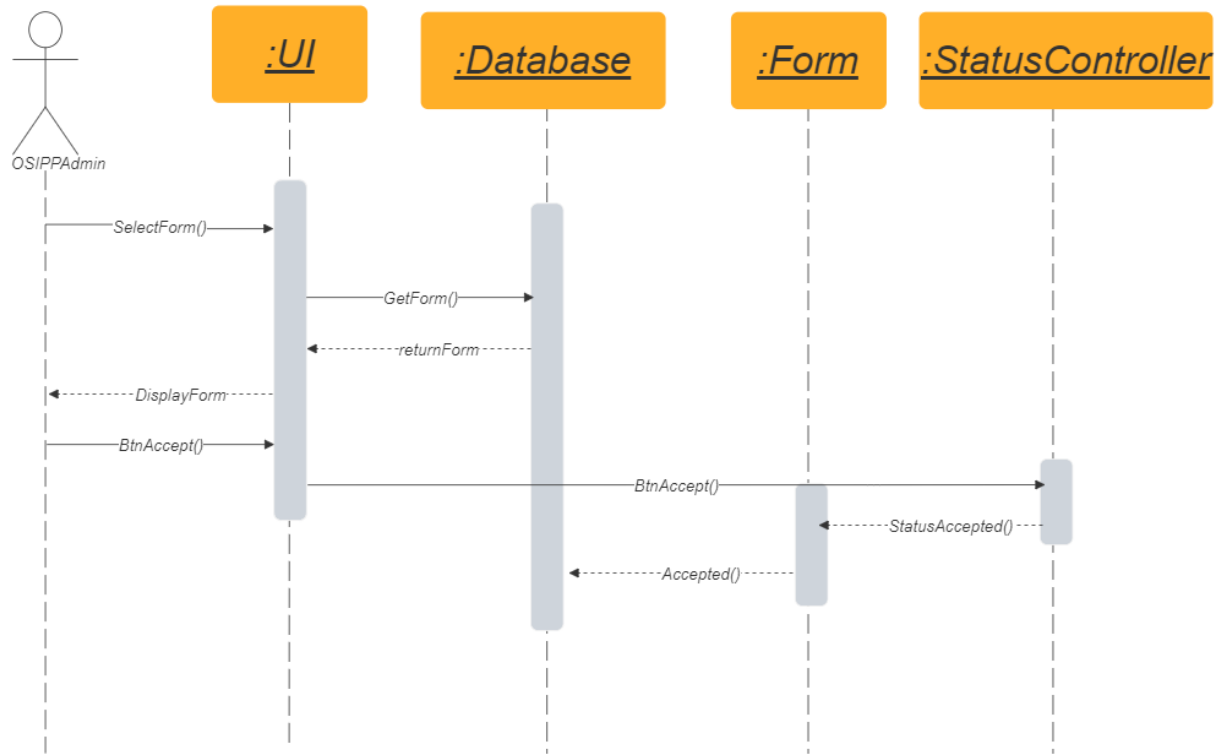
Data Dictionary

Interaction Diagrams

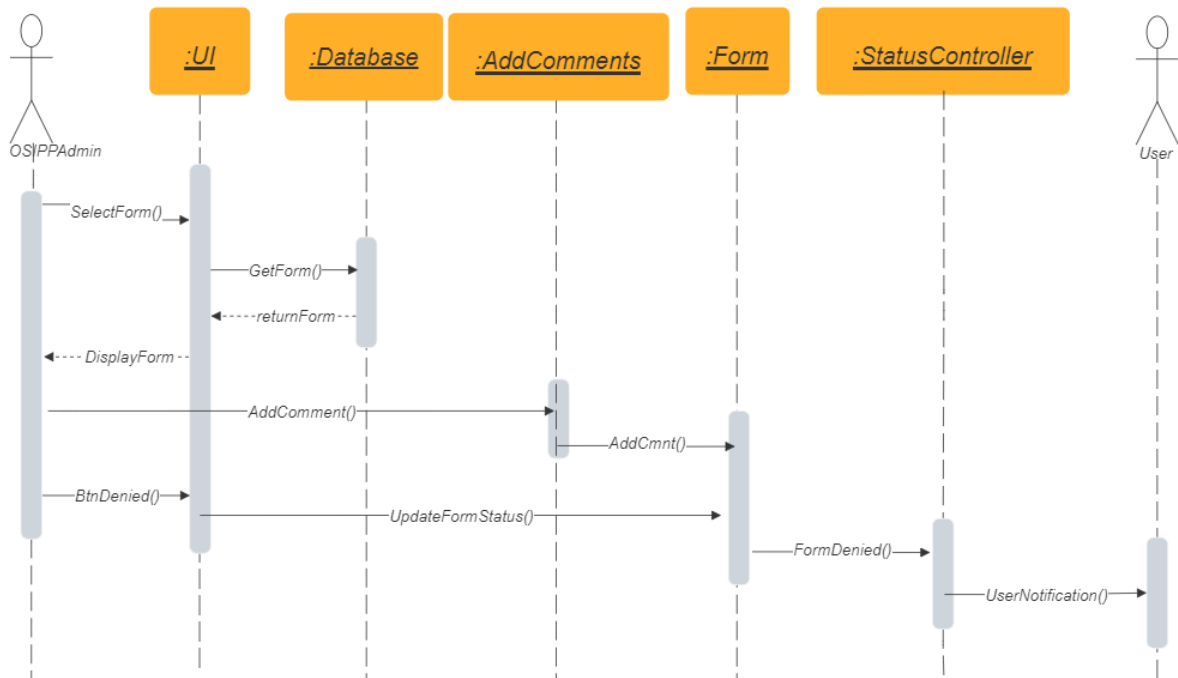
UC1: FillingForm



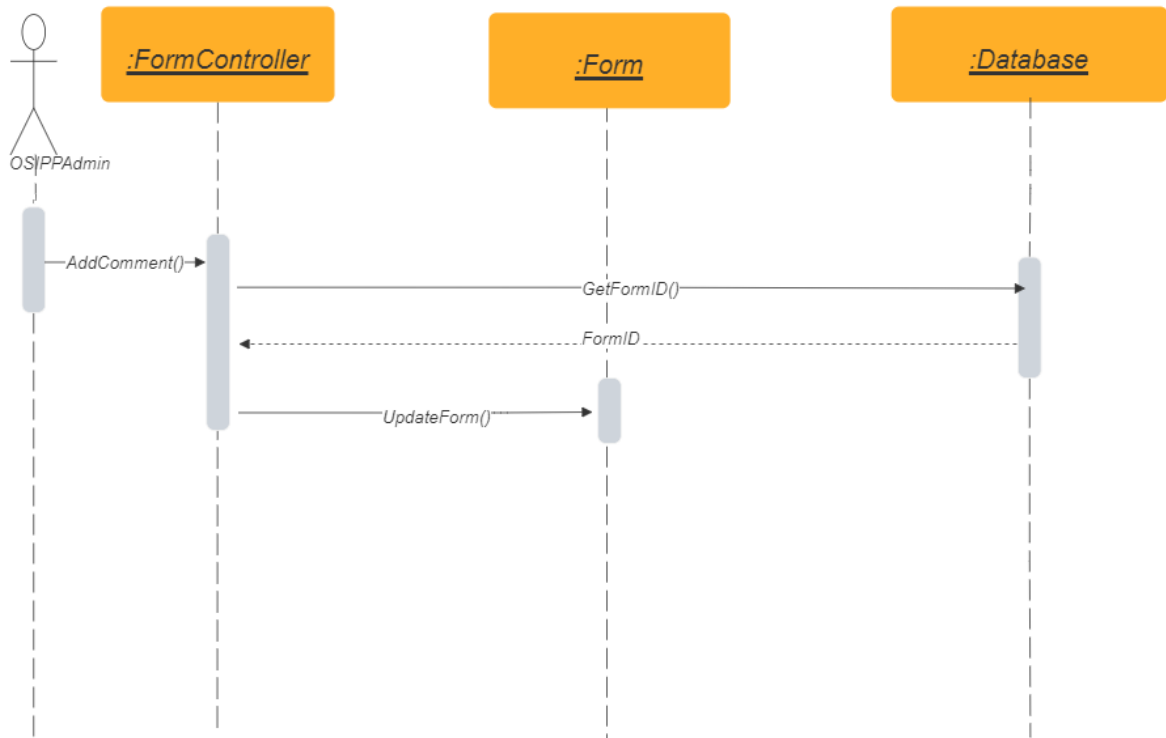
UC2: Accept Form



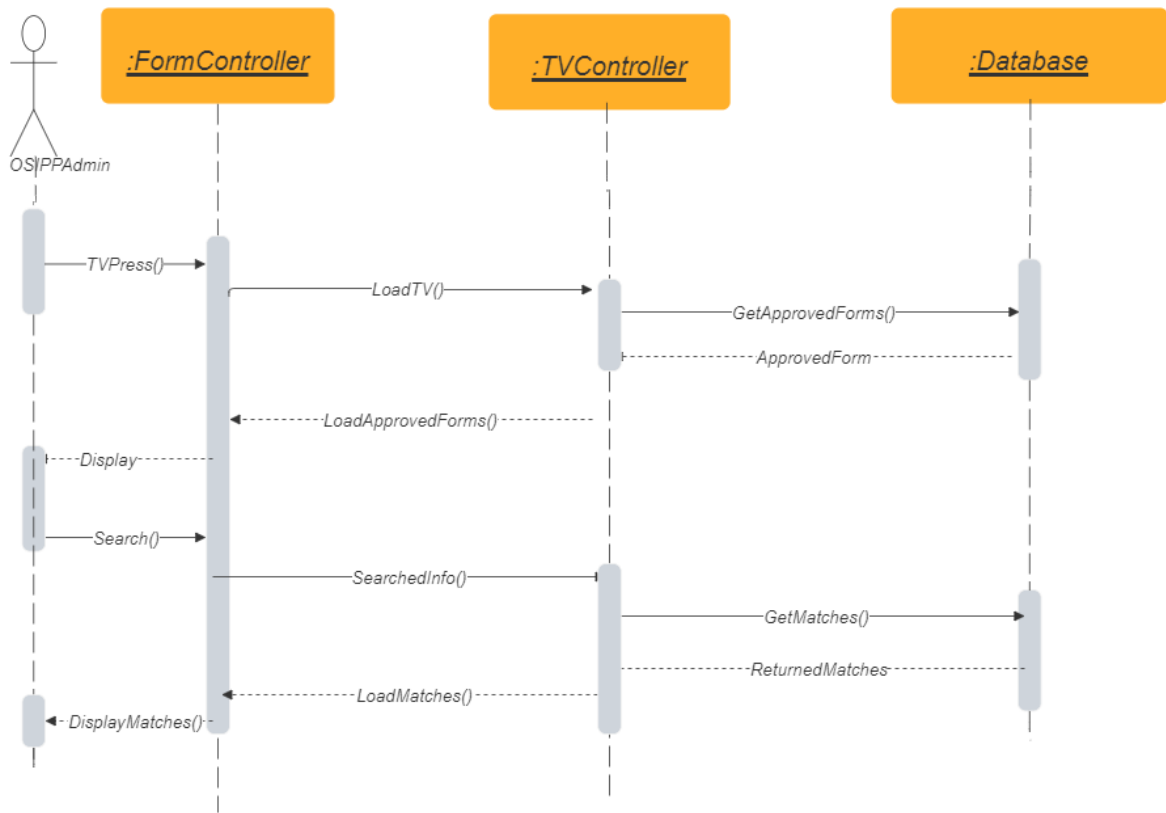
UC3: Deny Form



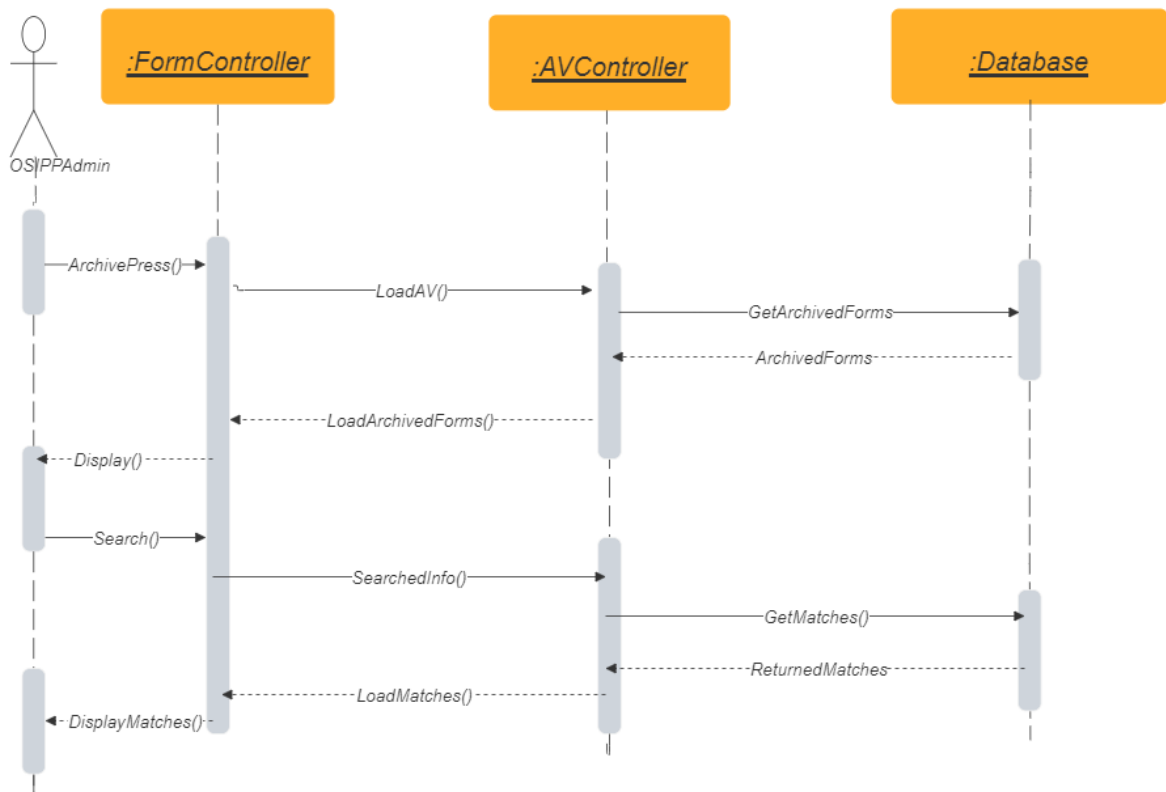
UC4: Add Comments



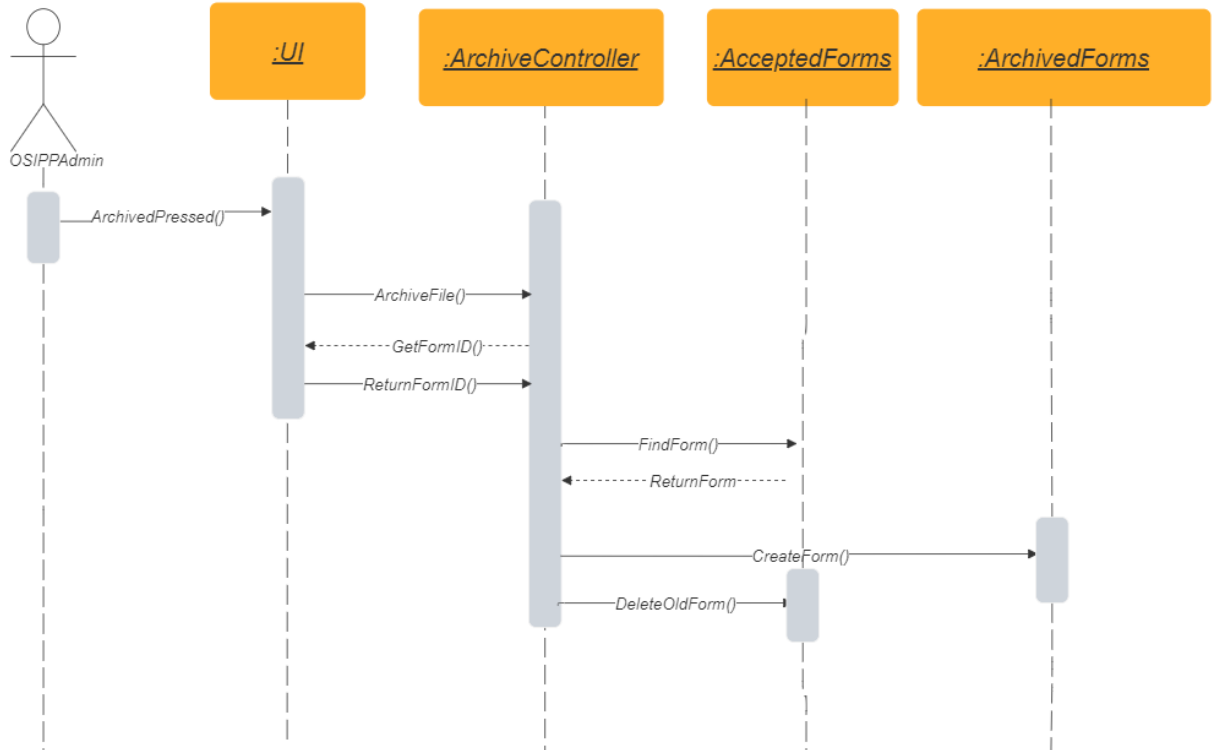
UC5: TogglingTableView



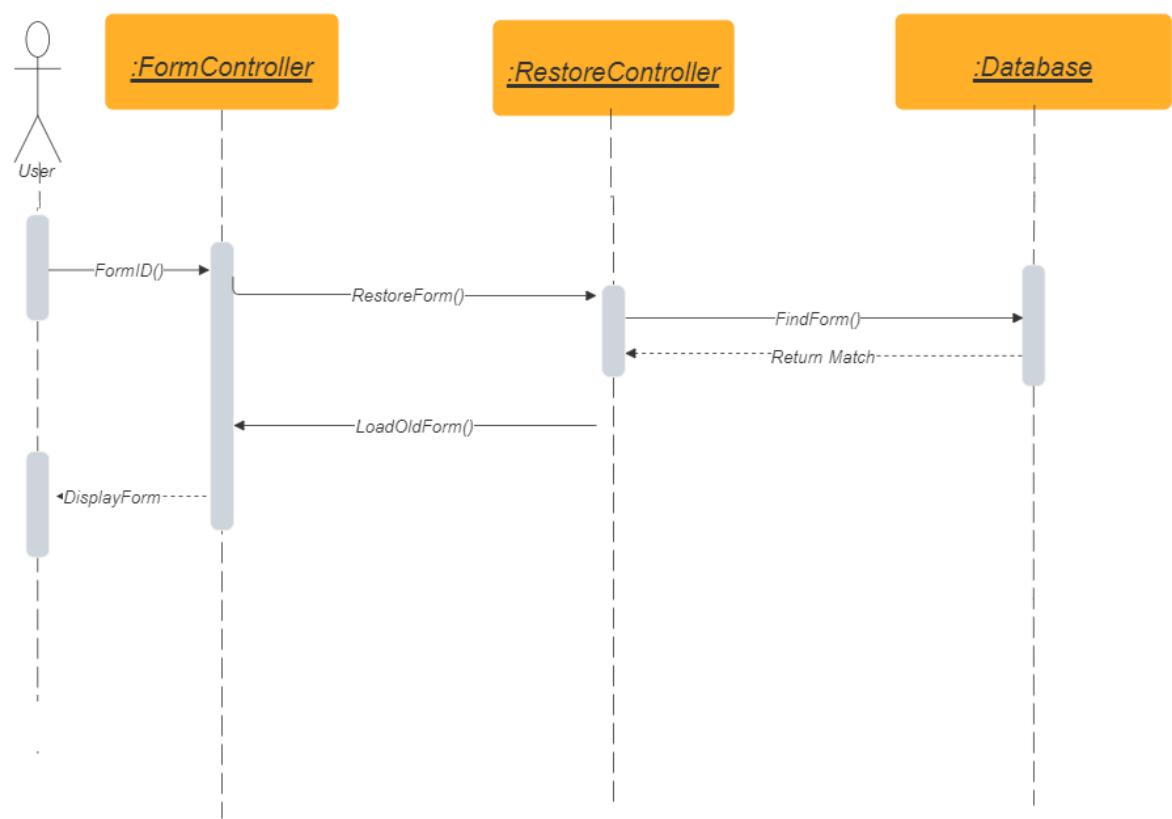
UC6: Archive View



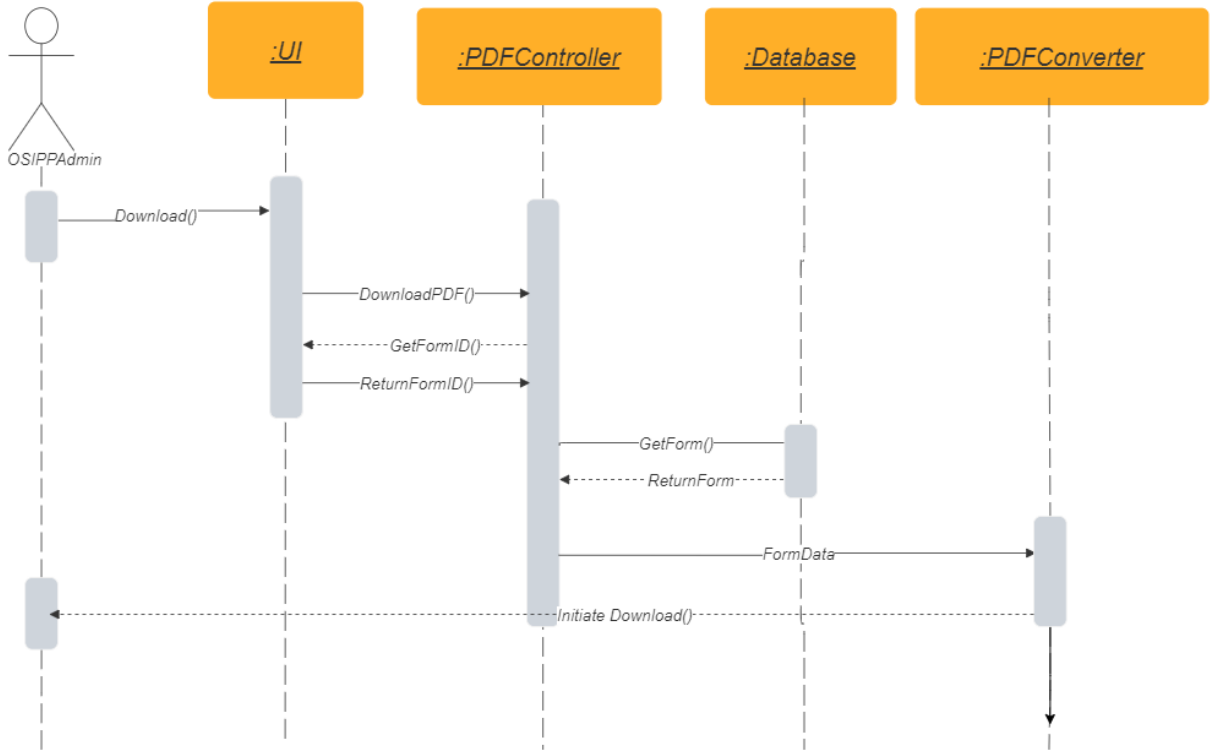
UC7: Archiving File



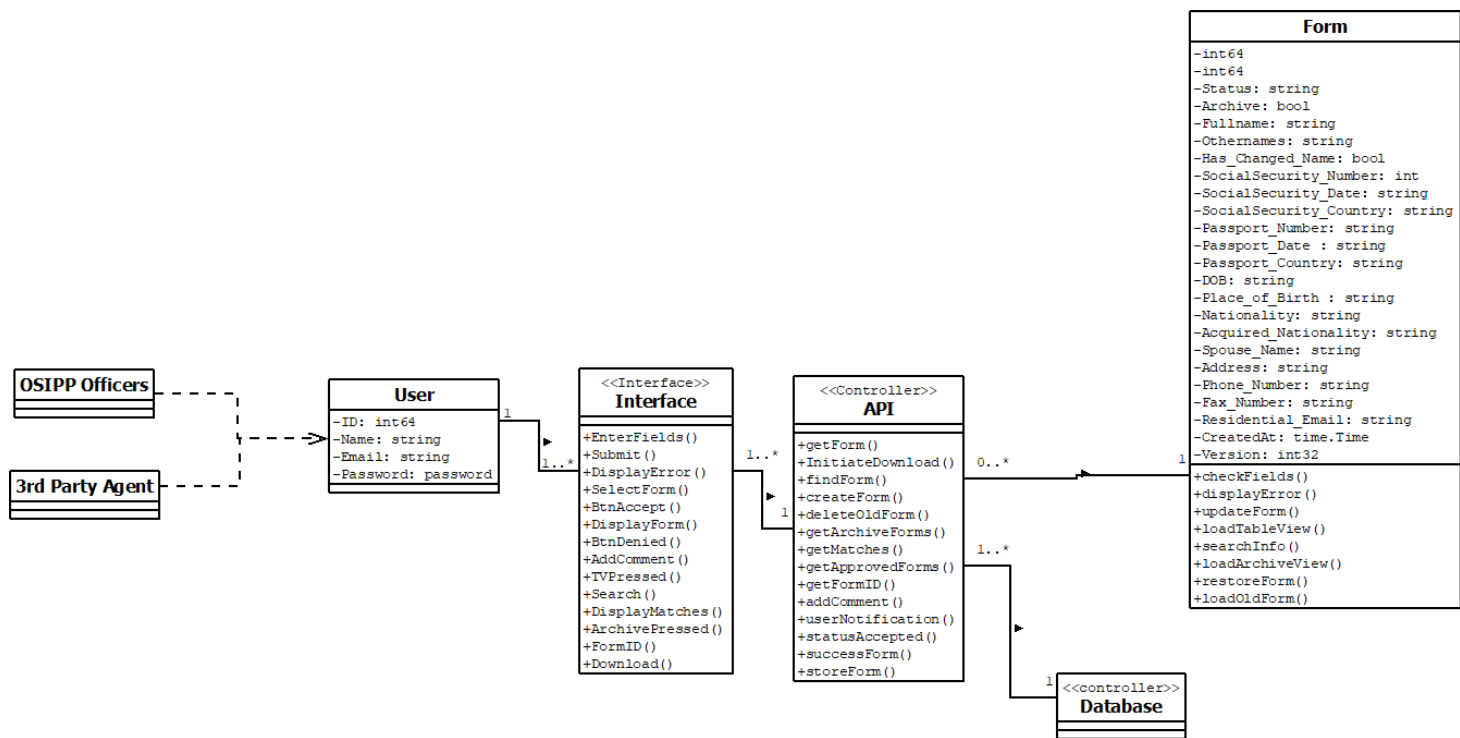
UC8: Restoring Form



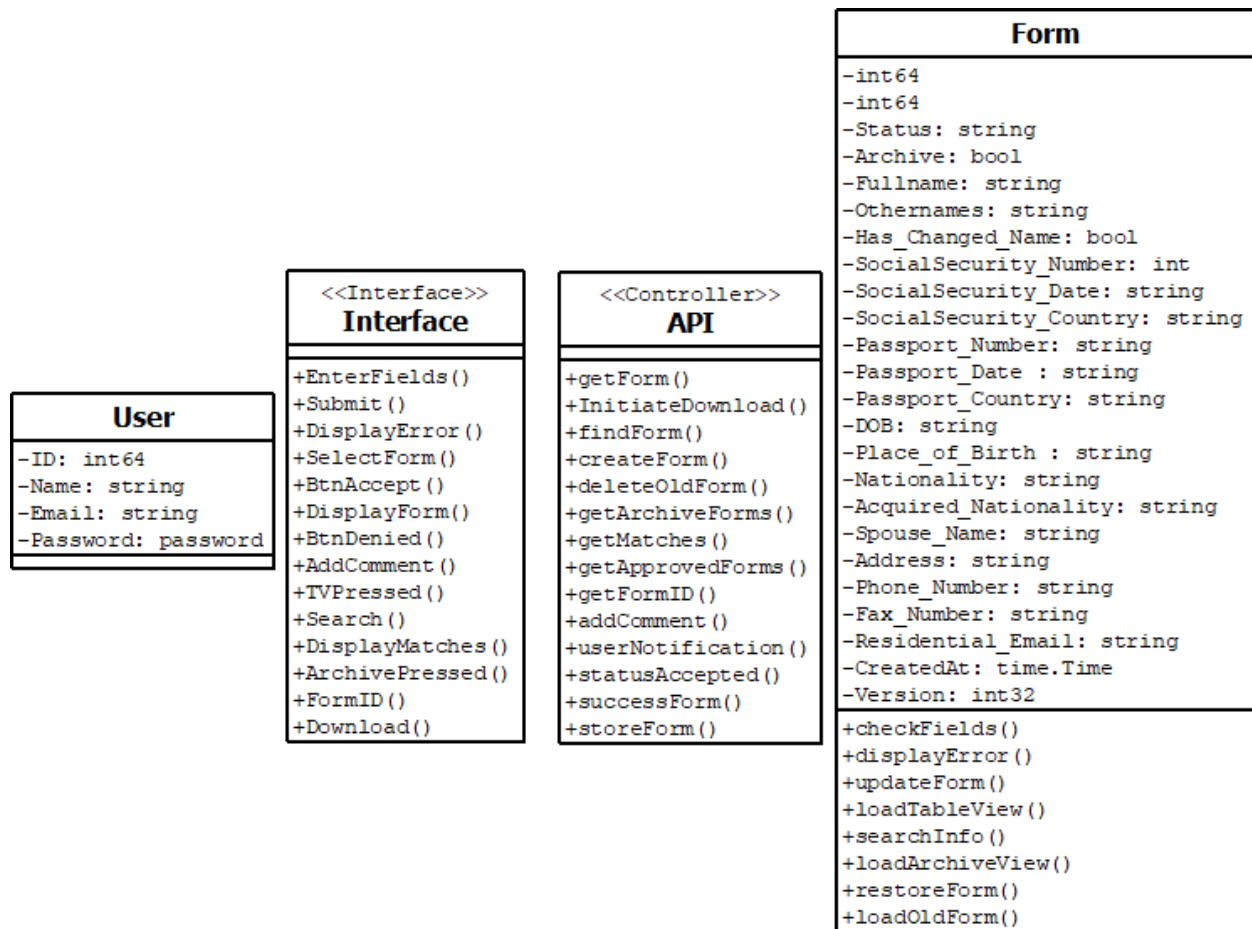
UC9: Downloading Form



Class Diagram and Interface Specifications



Class Diagram



Datatypes and Operation Signature

Traceability Matrix

	User	Interface	API	OSIPP Officers	3rd party agents	Form
User Interface	X	X		X	X	X
Controller		X	X			X
Form Handler						X
Database Model			X			X

Algorithms

The system will primarily consist of a sorting algorithm in order to group forms that external users have already filled that are linked with its form ID and user ID. The algorithm will be responsible to group forms based on its status if its brand new, pending for verification, has been verified or returned for error correction. In addition, the API will be responsible for verifying what type of users will be logging in to the system.

The system will be in charge of granting certain users access whether it be an admin, 3rd-party users or OSIPP officers depending on what data is passed to the API. When the system retrieves the API response, the system will present a welcome message along with the option to either create a new form, view status or make corrections.

Algorithm:

3rd-party user logs into Biographical Affidavit:

If “user credentials” is true

Display message “Welcome %sFirstName”

If else “user credentials” is false

Notify user that either username or password incorrect

While “user credentials” is true

Display message “Welcome %sFirstName”

Set form status to “new”

User fills out fields in the form and its submitted

Set form status to “pending”

If “Affidavit form” is incorrect

Return form back to user

Set form status to “returned”

User verifies and re-submit form

Set form status as “pending”

If “Affidavit form” passes verification

Set form status as “verified”

Transfer form to database

Concurrency

The system will allow middleware to enable communications with the application and database. Since there will be many users utilizing the software at once, event handlers will be placed to assist with the workload of interactions with the application and the users. This case applies to different users filling out the form and other users who are verifying many forms for different 3rd-party users.

User Interface Design and Implementation

Initially, the affidavit form and Review and Approval interface was designed using a UI designer. However, the current design has been implemented using HTML/CSS. As part of the implementation process, We made significant improvements to the form's usability by adding placeholders to the insertion boxes, making it less likely for users to make mistakes when filling out the form. Furthermore, we made slight changes to the Review and Approval interface by incorporating a search bar to enhance the user experience and enable quicker access to specific forms.

Fill form

- The Diagram below displays the Biographical Affidavit form

Biographical Affidavit

1. Affiant's Full Name:	<input type="text" value="Full name"/>																				
2. Other names used at any time:	<input type="text" value="Other name"/>																				
3. Have you ever had your name changed?	<input type="radio"/> Yes <input type="radio"/> No																				
If "Yes", please provide the following information:																					
Previous name(s):	<input type="text"/>																				
Reason for the change:	<input type="text"/>																				
4. Affiant's Identification No. applied to Government Record Systems. Two certified copies of picture ID must be submitted. The documents must be current and valid.																					
<table><thead><tr><th>Document</th><th>Number of Document</th><th>Date of issuance</th><th>Country of issuance</th></tr></thead><tbody><tr><td>Social Security</td><td>23262988</td><td>2/15/21</td><td>Belize</td></tr><tr><td>Passport</td><td>P00035684</td><td>10/28/20</td><td>Belize</td></tr><tr><td>National Health Insurance</td><td>N/A</td><td>N/A</td><td>N/A</td></tr><tr><td>Other ,specify</td><td></td><td></td><td></td></tr></tbody></table>	Document	Number of Document	Date of issuance	Country of issuance	Social Security	23262988	2/15/21	Belize	Passport	P00035684	10/28/20	Belize	National Health Insurance	N/A	N/A	N/A	Other ,specify				
Document	Number of Document	Date of issuance	Country of issuance																		
Social Security	23262988	2/15/21	Belize																		
Passport	P00035684	10/28/20	Belize																		
National Health Insurance	N/A	N/A	N/A																		
Other ,specify																					
5. Date of Birth:	<input type="text" value="DD/MM/YYYY"/>																				
6. Place of Birth:	<input type="text" value="District/State and Country"/>																				
7. Nationality: (Indicate how aquired.)	<input type="text" value="Birth"/> <input type="button" value="v"/>																				
8. Spouse's Name:	<input type="text" value="Name"/>																				
9. Affiant's Address:																					
<table><thead><tr><th>Physical Address</th><th>Telephone Number</th><th>Fax Number</th><th>Email Address</th></tr></thead><tbody><tr><td></td><td></td><td></td><td></td></tr></tbody></table>	Physical Address	Telephone Number	Fax Number	Email Address																	
Physical Address	Telephone Number	Fax Number	Email Address																		

View Unverified

- The Diagram below displays the Unverified interface.

Unverified	<input type="text" value="Search"/>	Unverified		
Verified				
Archive				
ID	FNAME	LNAME	DATE	STATUS
1	John	Doe	2022-01-01	Pending
2	Jane	Smith	2022-02-01	Pending
3	Bob	Johnson	2022-03-01	Pending
4	Alice	Williams	2022-04-01	Pending

View Verified

- The Diagram below displays the verified interface.

Unverified	<input type="text" value="Search"/>	Verified		
Verified				
Archive				
ID	FNAME	LNAME	DATE	STATUS
1	John	Doe	2022-01-01	Verified
2	Jane	Smith	2022-02-01	Verified
3	Bob	Johnson	2022-03-01	Verified
4	Alice	Williams	2022-04-01	Verified

View Archive

- The Diagram below displays the Archive interface.

Unverified

Verified

Archive

Search

Archive

ID	FNAME	LNAME	DATE	STATUS
1	John	Doe	2022-01-01	Verified

Login

- The Diagram below displays the login screen for OSSIP Officers to access system

Welcome

Username

Enter your username

Password

Enter your password

Log In

Cancel

Forgot your password?

Restore Form

- The Diagram below displays a form being restored after selecting an Affiant name.

Unverified

Verified

Archive

Unverified

ID	FNAME	LNAME	DATE	STATUS
1	John	Doe	2022-01-01	Pending
2	Jane	Smith	2022-02-01	Pending
3	Bob	Johnson	2022-03-01	Pending
4	Alice	Williams	2022-04-01	Pending

1. Affiant's Full Name:

2. Other names used at any time:

3. Have you ever had your name changed? ☐ Yes ☒ No

If "Yes", please provide the following information:

Previous name(s):

Reason for the change:

4. Affiant's Identification No. applied to Government Record Systems. Two certified copies of picture ID must be submitted. The documents must be current and valid.

Document	Number of Document	Date of issuance	Country of issuance
Social Security	23262988	2/15/21	Belize
Passport	P00035684	10/28/20	Belize
National Health Insurance	N/A	N/A	N/A
Other ,specify			

5. Date of Birth:

6. Place of Birth:

7. Nationality: (Indicate how aquired.)

8. Spouse's Name:

9. Affiant's Address:

Physical Address	Telephone Number	Fax Number	Email Address
Residential	501-202-4532	N/A	johnndoe@gmail.com
Business	N/A	N/A	N/A

Add comment

- The Diagram below displays a comment being added to the form because the affiant missed some fields.

1. Affiant's Full Name:

2. Other names used at any time:

3. Have you ever had your name changed? ☐ Yes ☒ No
If "Yes", (a) provide previous name(s) and (b) give the reason for the change.

4. Affiant's Identification No. applied to Government Record Systems. Two certified copies of picture ID must be submitted. The documents must be current and valid.

Document	Number of Document	Date of issuance	Country of issuance
Social Security	23262988	2/15/21	Belize
Passport	P00035684	10/28/20	Belize
National Health Insurance	N/A	N/A	N/A
Other, specified			

5. Date of Birth: (DD/MM/YYYY)

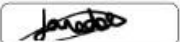
6. Place of Birth: include Distict/State and Country)

7. Nationality: Indicate how acquired.
☒ Birth
☐ Naturalization
☐ Marriage
☐ Other, specify:

8. Spouse's Name:

9. Affiant's Address:

Physical Address	Telephone No.	Fax No.	Email Address
Residential	501-223-4567	N/A	johndoe@gmail.com
Business	N/A	N/A	N/A

Signature: 

Date:

Comment

Form Has missing fields. Please add informationa dn resubmit

Deny and Accept From

- The Diagram below displays a form that is being reviewed by the OSIPP officer.

1. Affiant's Full Name:

2. Other names used at any time:

3. Have you ever had your name changed? ☐ Yes ☒ No
If "Yes", (a) provide previous name(s) and (b) give the reason for the change:

4. Affiant's Identification No. applied to Government Record Systems. Two certified copies of picture ID must be submitted. The documents must be current and valid.

Document	Number of Document	Date of issuance	Country of issuance
Social Security	23262988	2/15/21	Belize
Passport	P00035684	10/28/20	Belize
National Health Insurance	N/A	N/A	N/A
Other, specified			

5. Date of Birth: (DD/MM/YYYY)

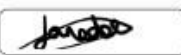
6. Place of Birth: include District/State and Country)

7. Nationality: Indicate how acquired:
☒ Birth
☐ Naturalization
☐ Marriage
☐ Other, specify:

8. Spouse's Name:

9. Affiant's Address:

Physical Address	Telephone No.	Fax No.	Email Address
Residential	501-223-4567	N/A	johndoe@gmail.com
Business	N/A	N/A	N/A

Signature: 

Date:

Ease of use

To ensure ease of use, the affidavit system will feature a user-friendly interface accessible to all users. The platform will be adaptable to different interfaces, ensuring excellent usability for everyone. In designing the OSIPP officer dashboard UI, we will prioritize simplicity and clarity to facilitate quick and easy learning for users. By prioritizing clear and intuitive design, we aim to create a system that is efficient, effective, and user-friendly for all.

Design of Tests

This design test stage will employ fourteen use cases. The tested functions will provide critical feedback on how each function will respond when the user interacts with the system. The test case will refer to one or more use cases. Other cases will be used as part of the testing procedures in fourteen cases. Each test case will pass or fail based on a fail or success scenario. As a result, the design test intends to provide the scope of the system's implementation.

Test Case 1

Test Case	TC-1
Use Case in Test	UC-1: FillingForm
Criteria for success/fail	Test is successful once the 3rd party agent fills all fields of the form
Input Data:	User Info
Test Procedure:	Expected Output
Step 1: Leaving "Affidavit's Fullname" field blank	Display error message "Fullname field can't be blank"
Step 2: Leaving half of the form filled out	Display error message "kindly verify that all information is entered"
Step 3: Filling all fields of the form to be submitted	Display message "Form was submitted"
This test case tests the functionality of verifying each field of	

the form that was filled(UC-1).

Test Case 2

Test Case	TC-2
Use Case in Test	UC-2: AcceptingForm
Criteria for success/fail	Test is successful once the 3rd party agent has all the necessary fields filled in the form
Input Data:	Accept Button
Test Procedure	Expected Output
Step 1: Accepting a blank form	Display error message “unable to accept a blank form”
Step 2: Accepting a form that all necessary fields are filled	Display Message “Form Accepted”
Step 3: Accepting a form that was incomplete	Display error message “unable to accept an incomplete form”
This test case tests the functionality of the accept button once the user pressed it(UC-2).	

Test Case 3

Test Case	TC-3
-----------	------

Use Case in Test	UC-3: DenyingForm
Criteria for success/fail	Test is successful once the form can denied after once information entered was incorrect
Input Data:	Return Button
Test Procedure	Expected Output
Step 1: Denying the form if the data is inaccurate	Display “Form has been returned”
This test case tests the web application ability to deny the form properly that is returned back to the 3rd party agent(UC-3).	

Test Case 4

Test Case	TC-4
Use Case in Test	UC-4: AddingComments
Criteria for success/fail	Test is successful once the public officer can add comments to each field of the

	form during the verification phase
Input Data:	Insert Comment Option
Test Procedure	Expected Output
Step 1: Adding comments to notify the user that “Date of Birth” is incorrect	Display comment beside “date of birth” field
Step 2: Adding a blank comment	No comment gets added
Step 3: Edit comments	Popup Menu displayed “edit comment”
This test case tests the web application’s ability to add comments(UC-4) to any field of the form.	

Test Case 5

Test Case	TC-5
Use Case in Test	UC-5: StoringData
Criteria for success/fail	Test that the application can store the data to the database after it passed the verification phase
Input Data:	None
Test Procedure	Expected Output
Step 1: Attempt to store data if the database is offline	Display message “unable to reach data server”
Step 2: Attempts to store data if the database is active	Display message to user “form has been stored”
This test case tests the web application’s ability to store the information into the database(UC-5).	

Test Case 6

Test Case 6	TC-6
Use Case in Test	UC-6: TogglingTableView
Criteria for success/fail	Test that the web application can display all forms that were approved by the public officer

Input Data	ToggleFormView Button
Test Procedure	Expected Output
Step 1: Selecting the option to display all approved forms	Another window will open to display all approved forms
Step 2: Attempting to view approved forms if the server is down	Display error message “Unable to view approved forms”
This test case tests the web application ability to view all approved forms in a readable manner(UC-6).	

Test Case 7

Test Case 7	TC-7
Use Case in Test	UC-7: CancelingForm
Criteria for success/fail	Test that the function can not only cancel the form but to also clear all fields.
Input Data	Click “Cancel Form”
Test Procedure	Expected Output
Step 1: Pressing Cancel when the form is completely filled	Display message “Form has been cancel” all fields are cleared
Step 2: Pressing Cancel when the form is already empty	Display message “No field(s) have been filled” void

	cancelation
This test case tests the cancelForm function that not only checks if the form is already filled but also ensures that all fields are cleared(UC-7).	

Test Case 8

Test Case 8	TC-8
Use Case in Test	UC-8: Submitting form
Criteria for success/fail	Test that the function can submit the form once the 3rd party agent fills it out properly
Input Data	Press Submit button
Test Procedure	Expected Output
Step 1: Attempts to submit a form that is incomplete	Display error message “Unable to submit form, kindly ensure all fields are completed” function SubmitForm did not execute
Step 2: Attempts to submit a form that has been filled out properly	Display message “Form has been submitted” function SubmitForm executes
Step 3: Attempts to submit a form that is blank	Display error message “Unable to submit a blank

	form” function SubmitForm did not execute
This test case tests the function’s ability to verify that the user has filled out the form properly before it can be submitted(UC-8)	

Test Case 9

Test Case 9	TC-9
Use Case in Test	UC-9: ViewArchive
Criteria for success/fail	Test that the function can allow the user to view all archived forms in the system
Input Data	ViewArchive button
Test Procedure	Expected Output
Step 1: Attempts to view archived forms while the server is offline	Display error message “unable to view archived forms” function ViewArchive didn’t execute
Step 2: Attempts to view archived forms while the server is online	A new window pops up to view all archived forms
This test case tests the functionality of viewing all archived forms that is stored in the web application(UC-9).	

Test Case 10

Test Case 10	TC-10
Use Case in Test	UC-10: ArchivingFile
Criteria for success/fail	Test that the web application is able to Archive a form during its verification phase
Input Data	ArchiveForm button
Test Procedure	Expected Output
Step 1: Archiving a form while the server is offline	Display error message “unable to archive form” function ArchiveForm did not execute
Step 2: Archive a form while the server is online	Display message “Form archived successfully” function
This test case tests the functionality of Archiving a form while its being verified(UC-10)	

Test Case 11

Test Case 11	TC-11
Use Case in Test	UC-11: RestoringForm
Criteria for success/fail	Test that the function is able to restore back the form once the 3rd party user receives a

	unique restoration code via email
Input Data	RestoreForm button
Test Procedure	Expected Output
Step 1: User attempts to receive restoration code during power outage	Display message “unable to generate restoration code” function RestoreForm didn’t execute
Step 2: User attempts to receive restoration code	Web application sends restoration code to user’s email, function RestoreForm executes
This test case tests the functionality of sending 3rd party agents a unique restoration code to retrieve back a form(UC-11).	

Test Case 12

Test Case 12	TC-12
Use Case in Test	UC-12: DownloadForm
Criteria for success/fail	Test is successful once the form can be downloaded into a PDF format for the public officer
Input Data	DownloadAffidavit button
Test Procedure	Expected Output

<p>Step 1: OSIPP public officer attempts to download the form that was filled</p> <p>Step 2: User attempts to download the form during a black out</p>	<p>Web application downloads the pdf version of the form will all user's info filled out, function DownloadAffidavit executed</p> <p>Display error message “unable to download form, please try again later”, function DownloadAffidavit did not execute</p>
<p>This test case tests the web application's functionality of converting the automated form into a downloadable pdf file for the user(UC-12).</p>	

Test Case 13

Test Case 13	TC-13
Use Case in Test	UC-13: MakingAccount
Criteria for success/fail	Test is successful once the OSIPP public officers has already gotten a working government email address
Input Data	OSIPP government email address
Test Procedure	Expected Output
Step 1: Admin attempts to	Display error message

create OSIPP user account with a false email address	“gobmail doesn’t exist”
Step 2: Admin attempts to create OSIPP user account with an actual government email address	Display message “User account as been created” function CreateAccount was executed
This test case tests the functionality of created accounts for OSIPP public officers to use the web application(UC-13).	

Test Case 14

Test Case 14	TC-14
Use Case in Test	UC-14: LoggingIn
Criteria for success/fail	Test is successful once both OSIPP officers and 3rd party agents are able to login to the web application
Input Data	Username ,Password
Test Procedure	Expected Output
Step 1: OSIPP officer enters invalid username and correct password	Display message at login screen “Username or password is incorrect”
Step 2: 3rd party agent enters valid username and incorrect password	Display message at login screen “Username or password is incorrect”
Step 3: OSIPP officer enters valid username and password	Web application allows access to OSIPP officer and to be greeted with a welcome message
Step 4: 3rd party agent enters valid username and password	Web application allows access to 3rd party agent and to be greeted with a welcome message
This test case tests the loggingIn function that ensures different users whether it be OSIPP	

officers or the public are able to login to the system and has certain rights to their respective accounts.

PROJECT MANAGEMENT AND PLAN OF WORK

Mergin the Contributions from Individual Team Members

Granted that is project is one of many semester long projects for senior students, time management has been the biggest hurdle to overcome. Though a long period of time has been allotted to work on this project. A lot of time was stolen to work on other assignments outside this project's respective course. Leaving only windows to actively working on the project.

The work was divided in the middle of this duration

Project Coordination and Progress Report

At the time of this report being submitted only Use Case -5 has been implemented on the system, currently only the backend of the system is being developed; while most of the use cases will involve the frontend of the system. The remaining uses cases are currently being worked on: UC-4, UC-7, UC-8, UC-9, UC-11, UC-12.

Plan of work

They are no fixed dates set as yet, however the group on a whole will continue to chip away at the system's incompleteness in their freetime till a solid schedule can be determined.

Breakdown of Responsibilities

Team Member system coding responsibilities

- Hipolito - was incharge of managing the relationships of the database on their object models, also supervising the implementation of the use cases.
- Jahmur - was incharge of designing the test cases for the system
- Michael - was incharge of managing the code base and revising all of the code before they'd be implemented onto the system

Coordination of integration

As the lead coder Michael will tackle the task of coordinating the integration of all the functionality of the system.

Coordination of integration testing

With the aid of Jahmur's designs, Rene's revisions of Hipolito's use cases; Michael will implement the integration testing.

HISTORY OF WORK, CURRENT STATUS, FUTURE WORK

REFERENCES