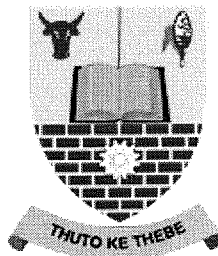


CSI 605



UNIVERSITY
of
BOTSWANA

24.5/40

PUMAS: A Web-based Publication Management System

REQUIREMENTS ANALYSIS DOCUMENT

OKAILE R MARUMO

201101426

GABRIEL RASENYAI

200503652

OFENTSE JABARI

201006088

KGALALELO MPIPI

200101032

COULSON KGATHI

200800533

TSHEPO MOILE

201005818

Table of Contents

Table of Figures.....	2
1. Introduction	3
1.1 Purpose	3
1.2 Scope	3
1.3 Objectives.....	3
1.4 Success Criteria of the Project.....	3
2. Current System	4
3. Proposed system	5
3.1 Overview	5
3.2 Functional requirements.....	5
3.3 Nonfunctional requirements	5
3.4 Constraints ("Pseudo requirements")	5
3.5 System models	6
→ 3.5.1 Use case model.....	6
3.5.2 Textual description of each use case	7
3.5.3 Sequence diagrams.....	10
3.5.3 Class diagram.....	14
3.5.4 State chart diagrams.....	15
3.5.5 Activity diagrams.....	15
3. 6 User interface	16

Table of Figures

FIGURE 1: PUMAS USECASE DIAGRAM	6
FIGURE 2: REGISTER USER SEQUENCE DIAGRAM.....	10
FIGURE 3: UPLOAD SEQUENCE DIAGRAM	11
FIGURE 4: APPROVE SEQUENCE DIAGRAM	12
FIGURE 5: VIEW DOCUMENTS SEQUENCE DIAGRAM.....	12
FIGURE 6: SEARCH DOCUMENTS SEQUENCE DIAGRAM.....	13
FIGURE 7: DOWNLOAD DOCUMENTS SEQUENCE DIAGRAM	13
FIGURE 8: PUMAS CLASS DIAGRAM	14
FIGURE 9: STATE CHART DIAGRAM.....	15
FIGURE 10: ACTIVITY DIAGRAM	15
FIGURE 11: LOGIN INTERFACE.....	16
FIGURE 12: HOME PAGE INTERFACE	17
FIGURE 13: UPLOAD DOCUMENT INTERFACE.....	18
FIGURE 14: VIEW DOCUMENT INTERFACE.....	19

1. Introduction

1.1 Purpose

The purpose of the PUMAS system is to manage publication of research output from University of Botswana by lecturers and students. It is a web-based platform for upload and viewing of published articles, journals and past projects in the University of Botswana by the University community and the world at large. It addresses (detects and prevents aka controls) publishing of plagiarized academic content.

1.2 Scope

The scope of the system is inclusive of general publication document management operations. The system's plagiarism control module shall not check plagiarism against external publications and will be restricted to checking for duplication of work within the PUMAS system repository.

1.3 Objectives

- a) Develop a data store for publications produced within the University of Botswana.
- b) Develop and integrate a plagiarism module to crosscheck new publications against existing and published documents for academic dishonesty before being published.
- c) Develop a workflow functionality to manage approval process for new publication.
- d) Develop content management functionality to manage and maintain the publication data store.
- e) Develop a web-accessible interface to view published content.

1.4 Success Criteria of the Project

The project shall be considered a success if it minimally has the following functionality:

- a) Upload and download of a publication.
- b) Ability to search the repository for a publication based on specified criteria.
- c) Manage the publishing process of submitted publications through a specified approval workflow.
- d) Ability to detect to similar work currently published when new requests for publication are submitted.

1/1

2. Current System

There is currently no central repository for publications by lecturers and students in the University of Botswana. There are however two operational platforms for storing research output by University lecturers, this being the UBRISA and the RMS systems.

There is a need for consolidation of the research output and also facilitate the capture/storing of research output by not only lecturers but students of the University as well as different levels of study.

Every year students produce theses, dissertations, and project documents that are not captured anywhere. Because a record of these is not kept, lecturers and students are not able to access them when needed.

Some unethical students take advantage of this lack of access, to duplicate the work of past students and pass it off as their own. Since there is no record to check against, lecturers and research supervisors find it difficult to monitor or prevent this dishonest behavior.

1/2
→ Discuss the shortcomings of these existing systems in relation to the objective set (including their functionalities)

3. Proposed system

3.1 Overview

The purpose of the proposed system is to provide a central repository for management of publications from the lecturers and students of the University of Botswana. The system aims to make these publications accessible to both the UB community and members of the public. The system will have a plagiarism detection mechanism that will aid in ensuring that research papers produced by students are authentic.

3.2 Functional requirements

- a) The system shall allow for upload of research documents (or artifacts)
- b) The system shall allow for download of research documents.
- c) The system shall allow for searching of research documents by different criteria. *3/4 Can you specify*
- d) The system shall display research artifacts.
- e) The system shall detect local plagiarism in uploaded/submitted research artifacts.
- f) The system shall provide user access control and authenticate users with different privileges.

3.3 Nonfunctional requirements

- a) The system shall be accessible everywhere in the world
- b) The system shall implement security measures to protect published document from abuse. *? ambiguous what measures?*
- c) The system must support concurrent access and querying of the repository from multiple users without any performance degradation. *1/2*

3.4 Constraints ("Pseudo requirements")

- a) The system's look and feel should be uniform across all major web browsers.
- b) The system shall be delivered by the end of the 2016/2017 first semester period, i.e. within 3 months. *1/1*

* The need for Supervisor approval
* The need for not allowing downloading of source code by Students
* etc were not captured !

3.5 System models

3.5.1 Use case model

3/5

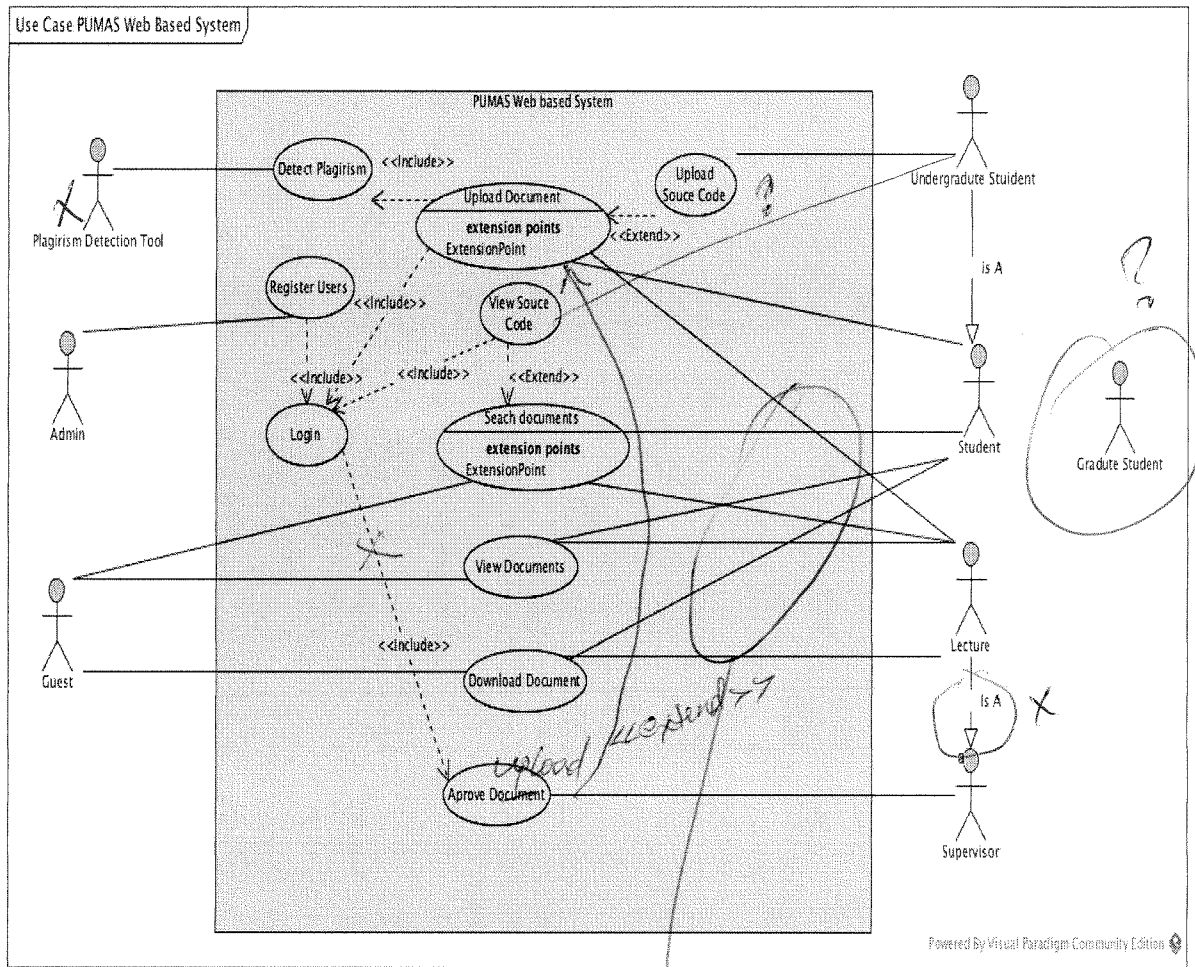


Figure 1: pumas usecase diagram

✓ In such cases, you could breakdown the use case diagram

3.5.2 Textual description of each use case

3/5

a) Use case name:

View Document

Actors:

Lecturers, Guests and Students

use the names of actors in the use case diagram!

Entry condition:

The user selects a specific document and then selects a 'View' option

Flow of events:

1. The user selects a document within the system
2. The system displays a description of the document and options to view or download the document.
3. The user selects the 'View' option related to the document they have selected

Exit condition:

The document is displayed within the users web-browser (depends on the user's web-browser's capabilities in some cases)

exception? special requirements?

b) Use case name:

Download Document

Actors:

Lecturers, Guests and Students

Entry condition:

The user selects a specific document and then selects a 'Download' option

Flow of events:

1. The user selects a document to download
2. The system displays a description of the document and offers to view or download the document selected.
3. The user clicks the 'Download' button.
4. The system displays a download window to select the path to download the file to on the local machine.

Exit condition:

The document is downloaded to the user's device

exception, special requirements?

c) Use case name:

Download Source-code

Actors:

Lecturers

Entry condition:

The user is logged in as either a lecturer

The user selects a specific source-code project and then selects a 'Download' option

Flow of events:

1. The user selects a project within the system
2. The system displays a description of the project and options to download the project source files.
3. The user selects the 'Download' option related to the project they have selected

doesn't exist in the use case diagram!

4. The system displays a download window to select the path to download the file to on the local machine.

Exit condition:

The project source code is downloaded to the user's local machine/device

d) Use case name:

Approve Document

Actors:

Lecturers (Supervisor)

Entry condition:

The user is logged in as a Supervisor lecturer.

The user selects a specific source-code project or document which they are the supervisor of and is pending their approval after being uploaded

Flow of events:

1. The user selects a document which is pending for approval within the system
2. The system displays details of the document with options to view, download and approve the document
3. The user selects the 'Approve' option related to the document after they have evaluated the document

Exit condition:

- ~ The system changes the documents status from pending-approval to approved.
- ~ The document will now be included in search results available to be viewed and downloaded by other users including guests and students

e) Use case name:

Search Document

Actors:

Student, Guest, Lecture, Administrator

Entry condition:

The user enters search criteria to look for a document.

Flow of events:

1. The user enters a search criteria, and clicks search
2. Search results are displayed in the screen
3. The user picks a relevant search result/document.

Exit condition:

1. Search results if any, are then displayed matching the specified criteria.

exception?

f) Use case name:

Upload Document

Actors:

Student, Lecture

Entry condition:

The user logged in as student of lecture loads an upload page on the system.

Flow of events:

1. The user clicks upload button.

2. An upload window appears to select the location of the document to upload.
3. The user clicks submit to upload the document.
4. The plagiarism detection tool inspects the document and gives response to the supervisor if the work is copied or not.

Exit condition:

1. The system will send a notification for a successful document upload.

Exception:

Document failed to upload; this may be due to network problem or any other reasons.

e) Use case name:

Detect Plagiarism

Actors:

Plagiarism Detection Tool

Entry condition:

The tool detects an uploaded document.

Flow of events:

1. The tool inspects the document for plagiarism
2. The tool then sends the inspection results to a supervisor.

Exit condition:

3. The tool sends plagiarism detection results.

f) Use case name:

Login

Actors:

Student, Lecture, Admin

Entry condition:

The user enters password and username to login.

Flow of events:

1. Enter login details and click submit
2. The system takes you to the home page.

Exit condition:

1. The system displays a successful login message

Exception:

1. Failed login, wrong login details.
2. User does not exist.

g) Use case name:

Register

Actors:

Admin

Entry condition:

1. The administrator loads data from external system.

Flow of events:

1. Load account details from an external system
2. The system creates account based on details from data from external systems.

Exit condition:

1. The system displays a successful accounts creation

Exception:

1. Failed account creation

upload source code?

3.5.3 Sequence diagrams

what types of objects?

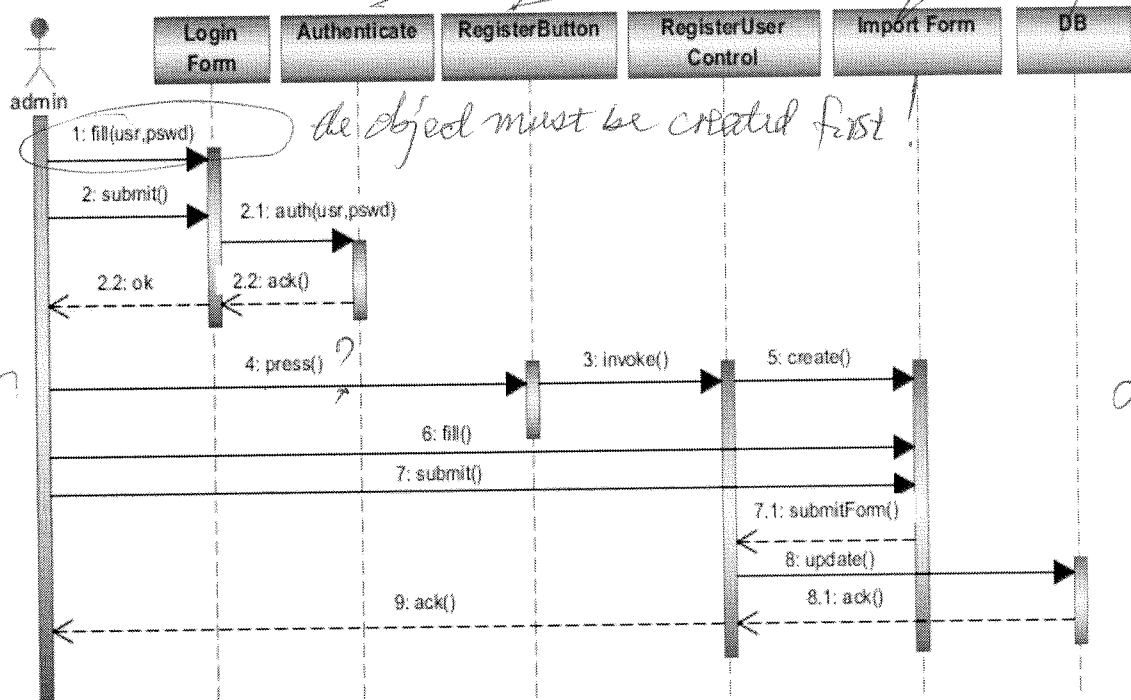


Figure 2: Register User Sequence Diagram

Description: The administrator is responsible for registering system users i.e. students and lecturers. For this to happen, the administrator has to login to the system through the login form, *boundary object*. Once the system has given the admin authorization, they can now request the data import form through the register button.

indicate the types & objects?

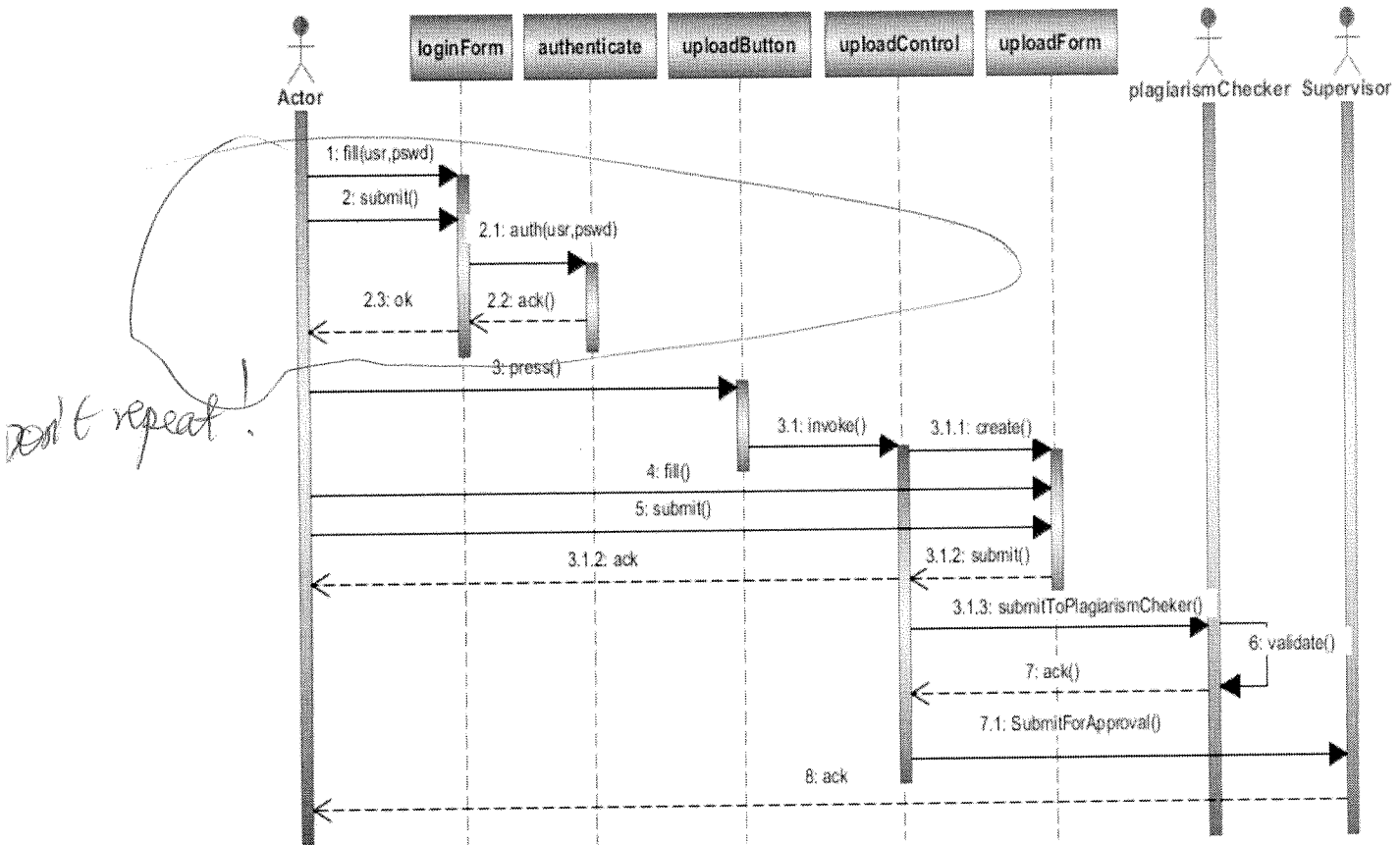


Figure 3: upload sequence diagram

Description: For users to upload their documents or projects, they have to be authenticated to control access to other system modules. Once the upload form has been filled and submitted, the plagiarism checker module scan the documents for plagiarism and pass control to the supervisor for final approval.

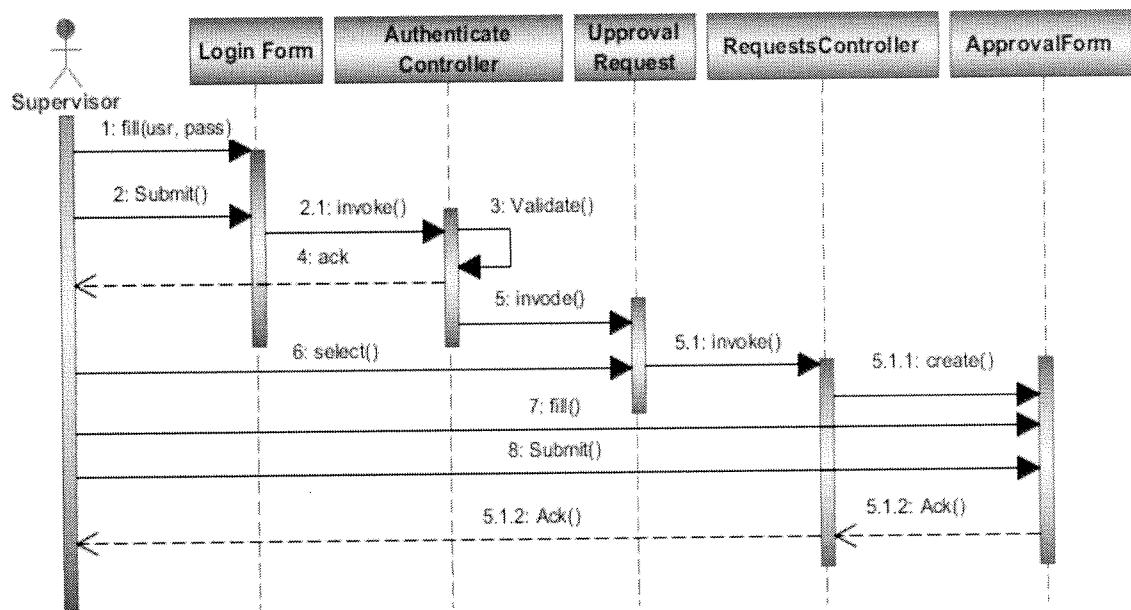


Figure 4: Approve Sequence Diagram

Description: Once the supervisor has been authenticated, they can select from a pool of request for review. The supervisor then interact with the approval form (**boundary object** to comment, approve or disapprove) and send feedback to the system to either reject or publish (make publicly available) the project or document.

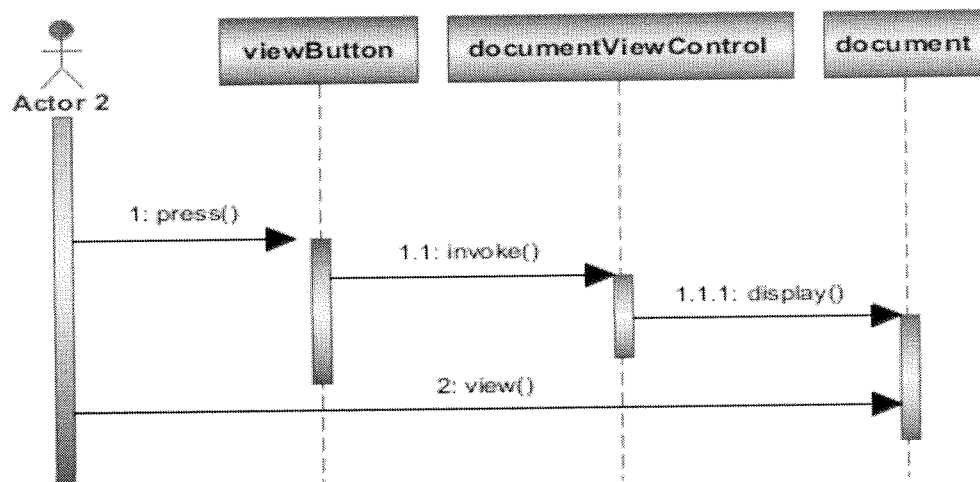


Figure 5: View Documents Sequence Diagram

Description: After document searching, the view button, *boundary object* allows the user to display a project/ document abstract. Detailed information about the document(s) can only be viewed from the abstract panel.

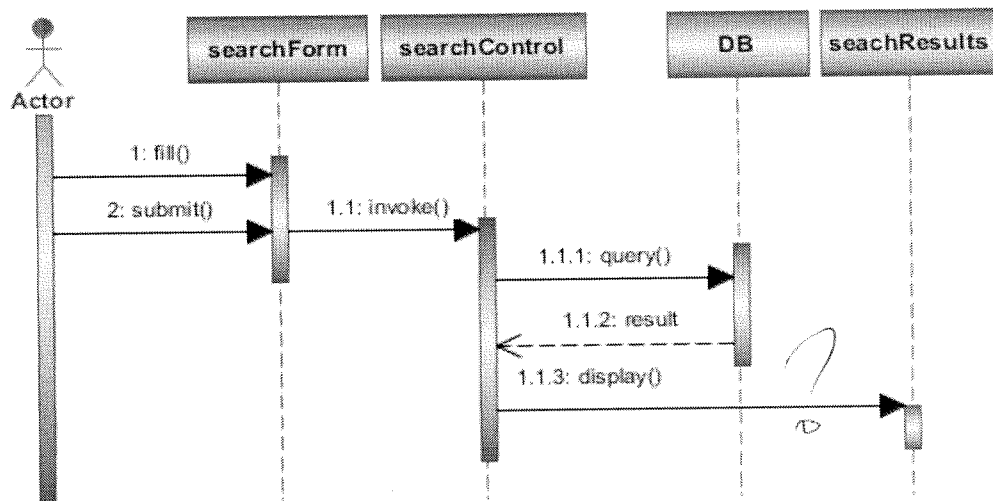


Figure 6: Search Documents Sequence diagram

Description: The search form allows (boundary object) user the lookup for a desired document from the system which involves database (entity object) querying.

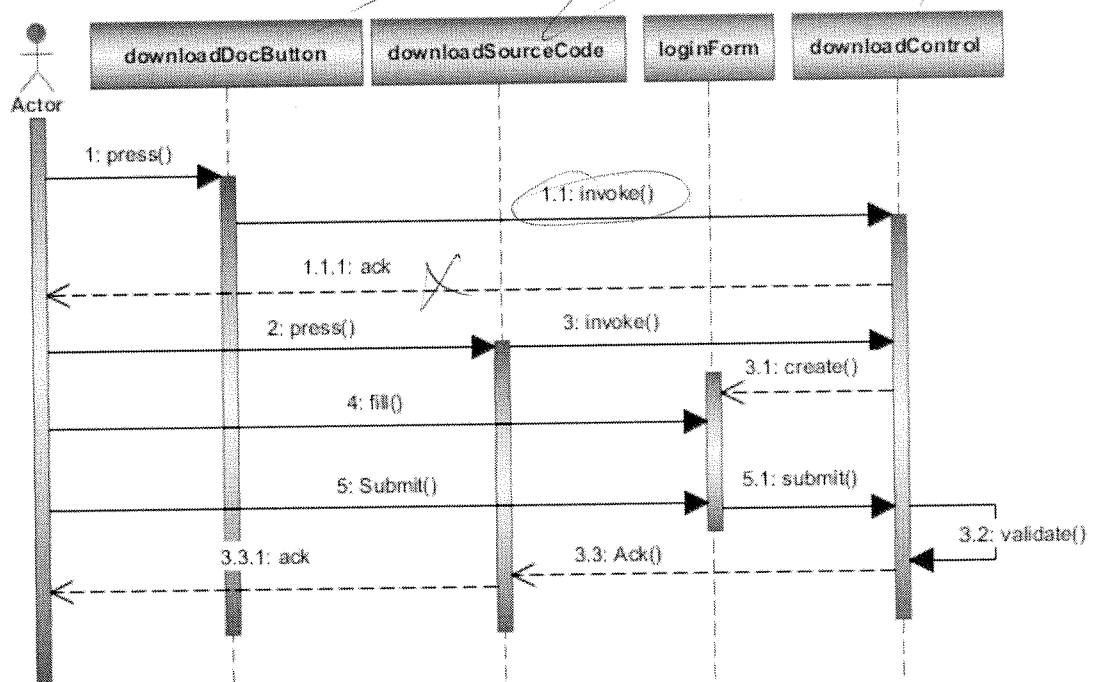


Figure 7: Download Documents Sequence Diagram

Description: Document download is publicly available for every user with no restriction as opposed to source code download. User are authenticated to download project(s) source code.

what about for the other 4 use cases?

3.5.3 Class diagram

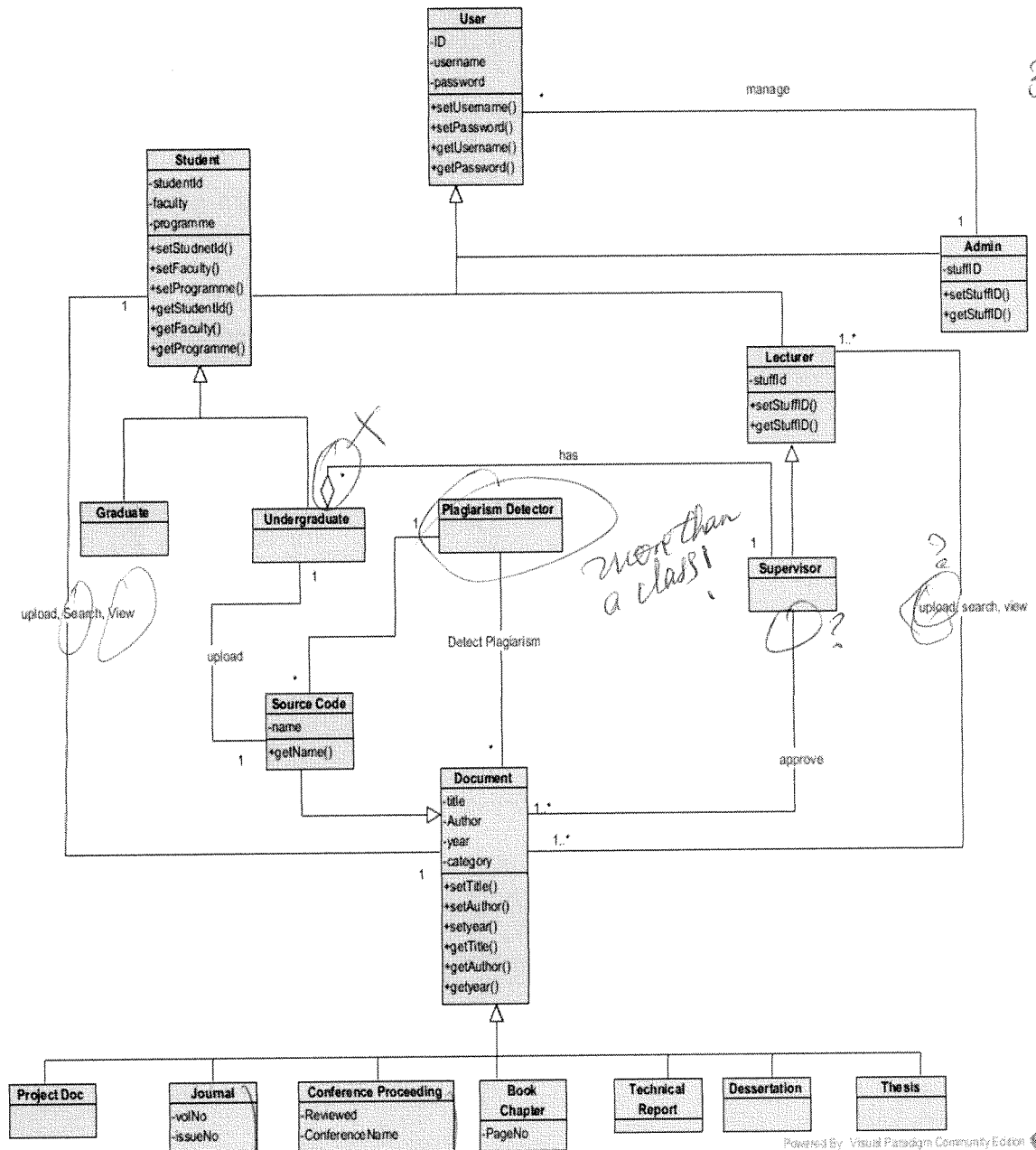


Figure 8: Pumas Class Diagram

3.5.4 State chart diagrams

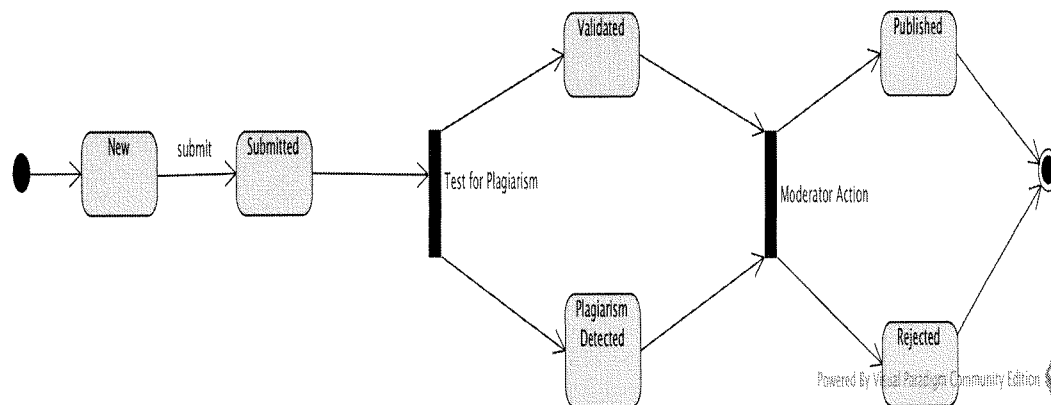


Figure 9: State Chart Diagram

3.5.5 Activity diagrams

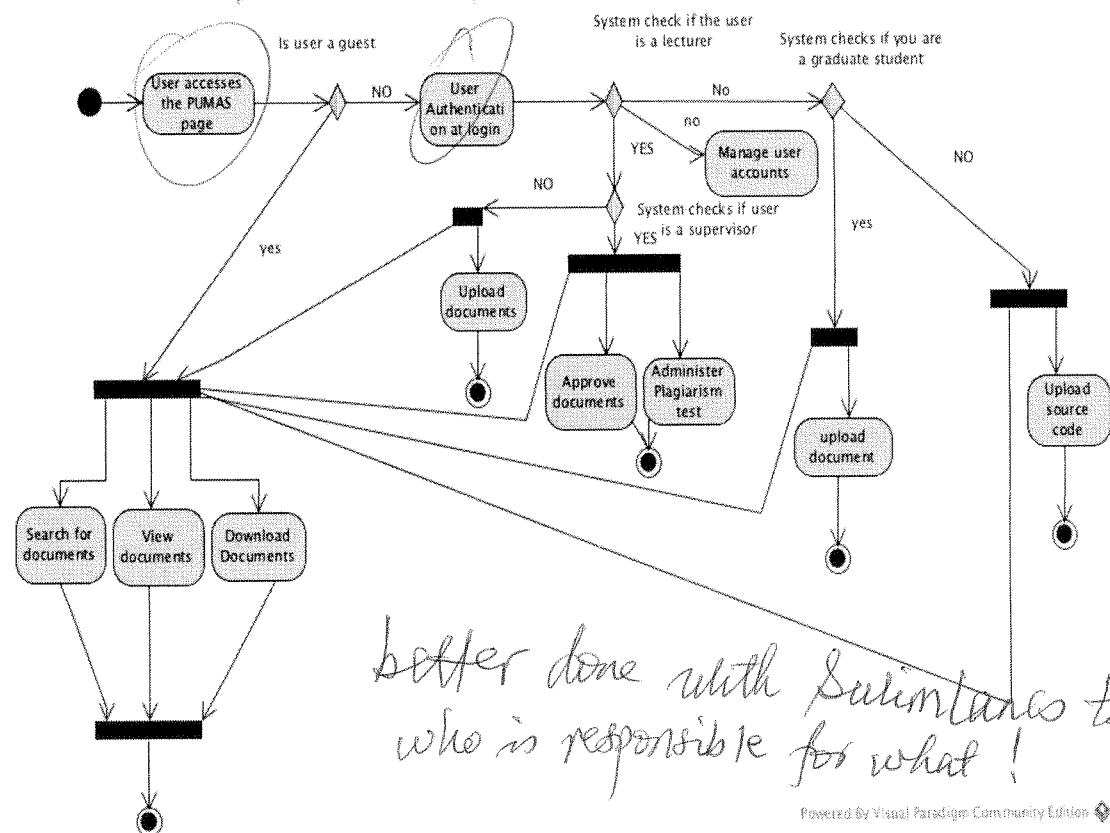


Figure 10: Activity Diagram

3. 6 User interface

Interfaces

Login Interface

4/5

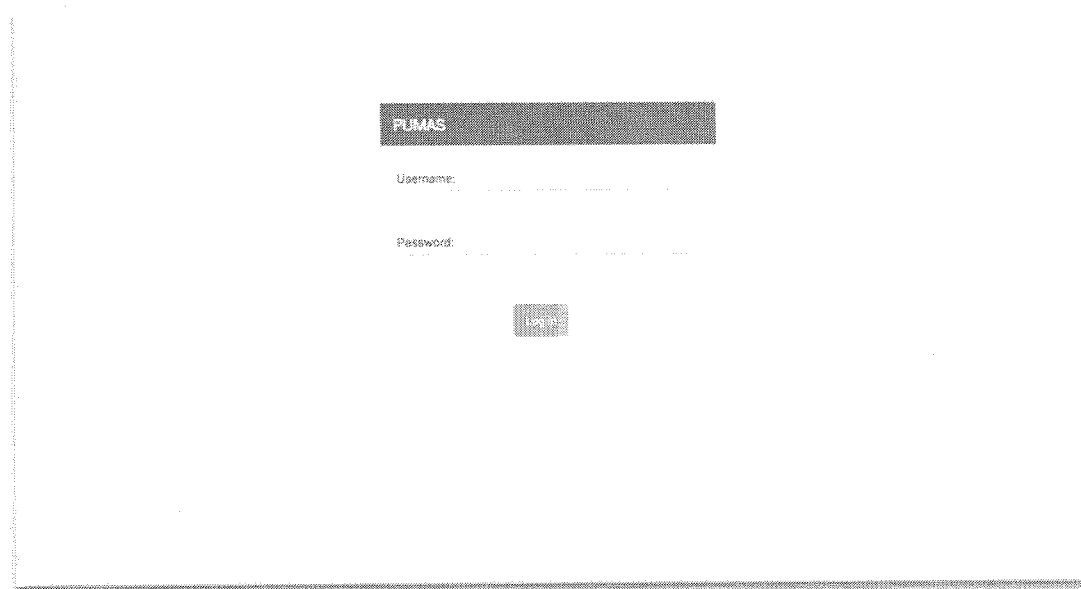


Figure 11: Login Interface

Home Page

PUMAS University of Botswana			
		Search	
Content			
About Us	Contact Us		

where is the link for
download, upload, etc ?

~~login~~ ?

login come

Figure 12: Home Page Interface

Upload Document

PUMAS University of Botswana	Logged in as username	
	<input type="text"/> Search	
<div><div>Document details</div><div><div>File location</div><div>Upload button</div></div></div>		
About Us	Contact Us	

Figure 13: Upload Document Interface

View Document

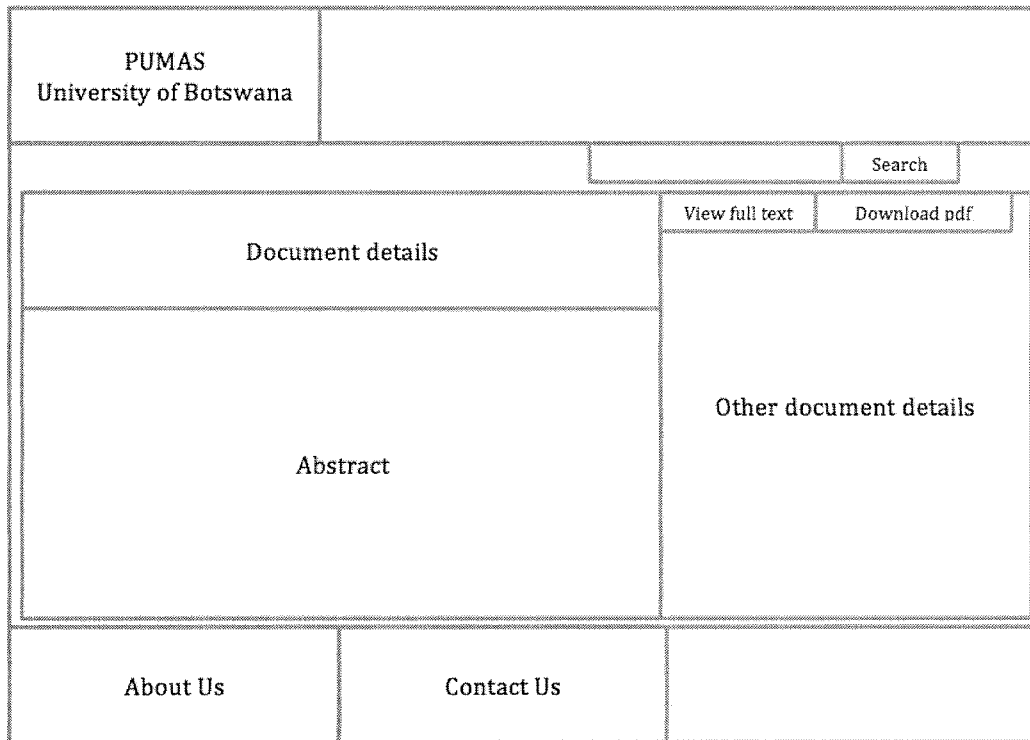


Figure 14: View Document Interface

Others such as download,