

BITBYBIT

COS301: SOFTWARE ENGINEERING

TWITTER SUMMARISER: COS301

By submitting this assignment I confirm that I have read and am aware of the University of Pretoria's policy on academic dishonesty and plagiarism and I declare that the work submitted in this assignment is my own as delimited by the mentioned policies. I explicitly declare that no parts of this assignment have been copied from current or previous students' work or any other sources (including the internet), whether copyrighted or not. I understand that I will be subjected to disciplinary actions should it be found that the work I submit here does not comply with the said policies.

Contents

1	Introduction	2
1.1	Purpose	2
1.2	Scope	2
1.3	Definitions, Acronyms, and Abbreviations	2
1.4	References	2
1.5	Overview	2
2	Class Diagram	3
3	User Characteristics	3
4	Functional Requirements	4
4.1	Use Cases	4
4.2	Requirements	8
4.3	Subsystems	10
5	Quality Requirements	12
5.1	Performance	12
5.2	Modifiability	12
5.3	Security	12
5.4	Availability	12
5.5	Usability	12
5.6	Integrability	12
6	Trace-ability Matrix	13

List of Figures

1	Twitter-Summarise Overall Class Diagram	3
2	Use case diagram for Register/Login	4
3	Use case diagram for searching for reports	5
4	Use case diagram for Generating Reports	5
5	Use case diagram for share generated report	6
6	Use case diagram for edit shared report	6
7	Use case diagram for clone shared report	7
8	Use case diagram for manage reports	7
9	Use case diagram for schedule reports	8
10	Use case diagram for publish generated reports	8
11	Use case diagram for explore published reports	9
12	Overall Use Case Diagrams, Grouped into SubSystems	11

1 Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of the Twitter Summariser. It will explain the purposes and feature of the system and what the system will do. This document is intended for the project owners, COS 301 lecturers and the developers of the system.

1.2 Scope

This software system will be a Twitter Summariser for a user who would like summaries of most talked-about topics on Twitter. The system will be designed to provide meaningful summaries from tweets of the user's choice in an article-like format to users by providing a tool to curate tweets and to generate such articles. By providing such a system, this system's users will get highlights of discussions on Twitter of their choice.

More specifically, this system is designed to allow users to search for topics using keywords to extract tweets to generate a report. The system will facilitate communication between other users.

1.3 Definitions, Acronyms, and Abbreviations

Creator Person who can generate, edit, share and publish reports.

Database Collection of information monitored by the system.

Twitter Online news and social networking site.

Reader Person who can read published reports only.

1.4 References

Paul Gil. (2019). What Exactly Is Twitter? And What Is "Tweeting"? Lifewire.
<https://www.lifewire.com/what-exactly-is-twitter-2483331>

1.5 Overview

In section 2 of this document, a class diagram of the this system will be provided. This diagram will illustrate the software solution by illustrating different entities of the system and the relationship between the entities of the system.

The third section of this document details the user characteristics of the system users.

The fourth section of this document is written primarily for the developers and describes in detail the system requirements of the product.

The fifth section of this documents specifies the quantity of each quality requirements for this system.

In section 6, a requirements vs subsystems matrix is provided to illustrate the allocation of each system requirement to a subsystem.

2 Class Diagram

Visual Paradigm Standard (©holo (University of Pretoria))

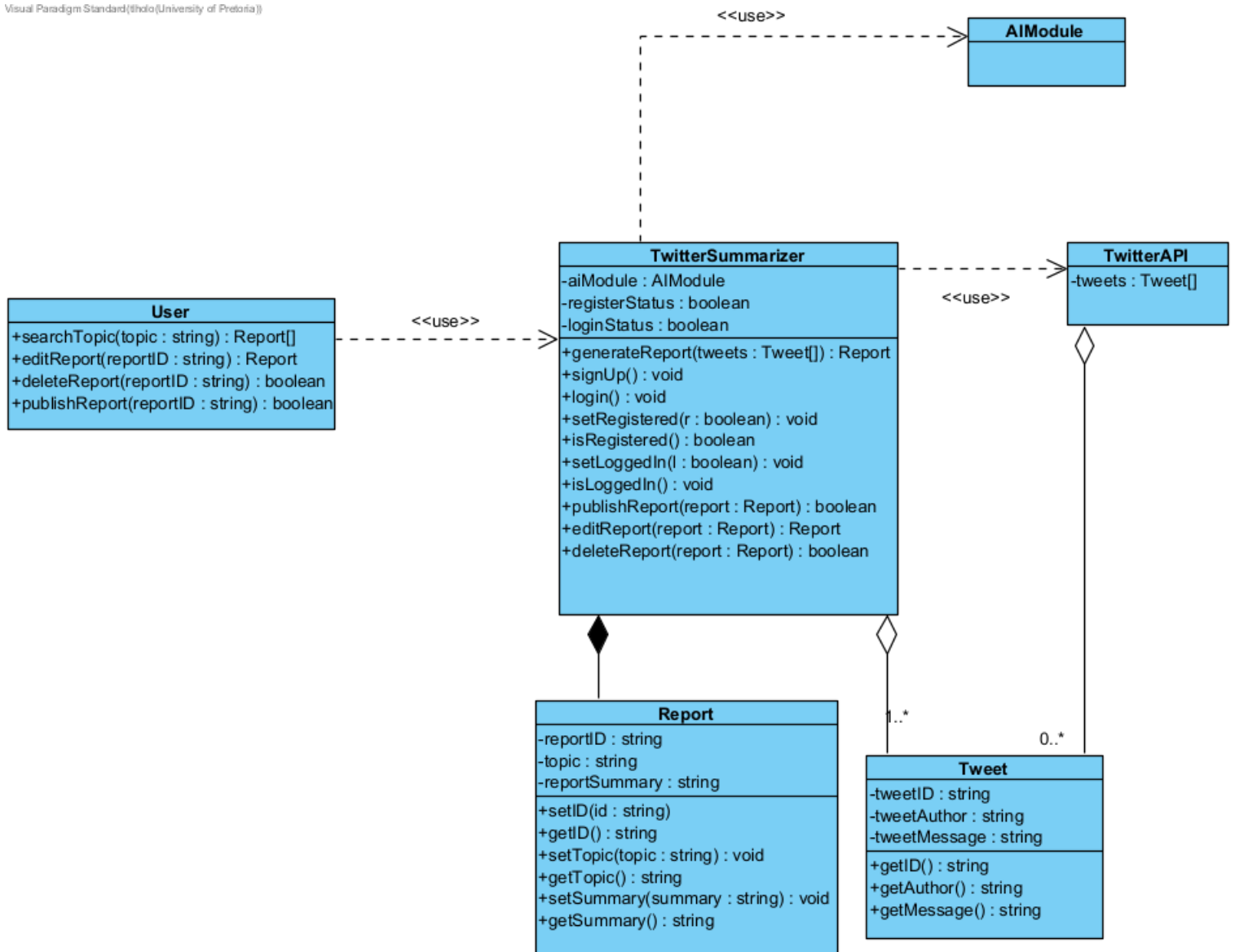


Figure 1: Twitter-Summarise Overall Class Diagram

3 User Characteristics

The Creator is expected to be computer and Internet literate and be able to use a search engine, text editor and button. The main screen of the Twitter Summariser Website will have a search function.

4 Functional Requirements

4.1 Use Cases

UC1: Register/Login The Creator enters authentication details to enter web app.

Before this use case can be initiated, the creator has already accessed the Twitter Summariser Website.

1. If a user does not have a Twitter Summariser account, then the user is prompt to create one.
 - (a) The system will register the new user as a creator.
 - (b) The system presents the user with the login option.
 - (c) The new creator is prompted to enter credentials.
 - (d) The system provides the creator access to use the system.
2. If a creator has a Twitter Summariser account, then the creator is prompted to enter credentials.
 - (a) The system provides the creator access to use the system.

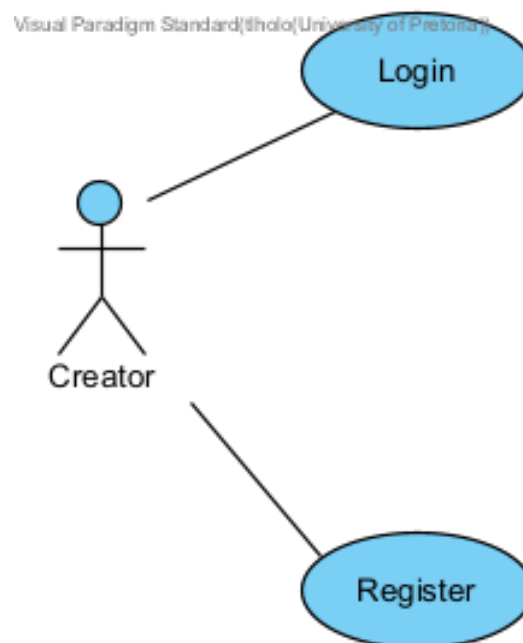


Figure 2: Use case diagram for Register/Login

UC2: Search for topic The Creators enters keyword to generate report.

Before this use case can be initiated, the creator has already connected to the Twitter Summariser Website.

1. The creator enters keyword or topic to search.
2. The system presents a choice for entering keywords or topics.
3. The creator can filter the search by time range, minimum number of likes and number of tweets to return.

4. The system presents a choice of filters(time range, minimum number of likes and number of tweets to return).
5. The creator can sort the search results by most replies and most retweets.
6. The system presents a choice of sorting by most replies and most retweets.

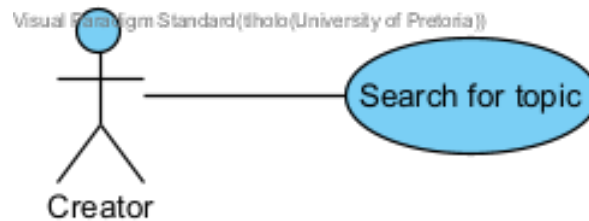


Figure 3: Use case diagram for searching for reports

UC3: Generate report The Creator pushes button to generate report.

Before this use case can be initiated, the creator has already connected to the Twitter Summariser Website.

1. The Creator selects to *Generate Report*.
2. The system returns a report in the form of a hyperlink.

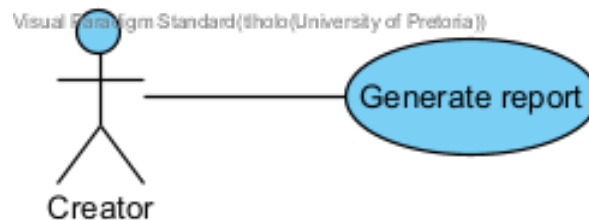


Figure 4: Use case diagram for Generating Reports

UC4: Share generated report

The Creator shares report with other creators.

Before this use case can be initiated, the creator has already connected to the Twitter Summariser Website.

1. The creator selects a saved report.
2. The system presents the selected report.
3. The creator chooses to share report.
4. The system presents an input to enter username of creator to share to.
5. The system notifies the other creator about shared report.
6. The system sends shared report link to other creator.

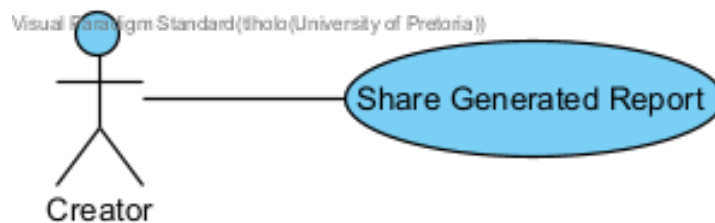


Figure 5: Use case diagram for share generated report

UC5: Edit shared report The other creators edits shared reports.

Before this use case can be initiated, the creator has already connected to the Twitter Summariser Website.

1. The creator selects *Shared reports*
2. The system provides the creator with shared reports.
3. The creator selects a shared report.
4. The system presents the creator with the selected report.
5. The system presents the creator a choice to edit shared report.
6. The creator is given the ability to edit the shared report.

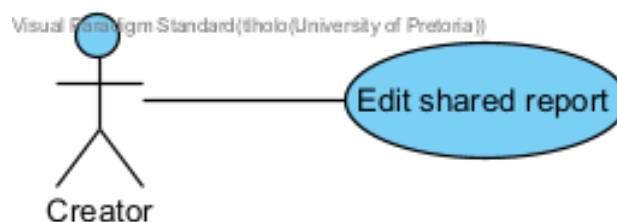


Figure 6: Use case diagram for edit shared report

UC6: Clone shared report The other creators can clone shared reports.

Before this use case can be initiated, the creator has already connected to the Twitter Summariser Website.

1. The creator selects *Shared reports*
2. The system provides the creator with shared reports.
3. The creator selects a shared report.
4. The system presents the creator with the selected report.
5. The system presents the creator a choice to clone shared report.
6. The system provides the creator with a copy of shared report.

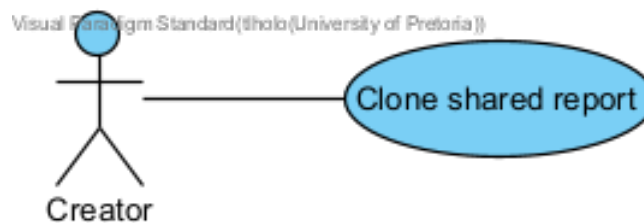


Figure 7: Use case diagram for clone shared report

UC7: Manage reports The Creator can edit and delete previously generated reports.

Before this use case can be initiated, the creator has already connected to the Twitter Summariser Website.

1. The creator selects *Reports*
2. The system presents the creator with generated reports.
3. The system presents the creator a choice of deleting or editing a report.
4. The creator chooses to delete or edit.
5. The system links to the Twitter Summariser database.
6. If the creator is deleting a report, the system removes report from Twitter Summariser database; else if the creator is editing a report, the system returns the report for editing.

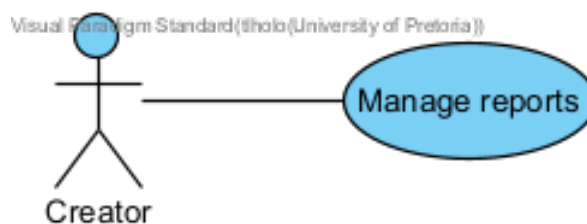


Figure 8: Use case diagram for manage reports

UC8: Schedule reports The Creator can schedule automatic generated reports for specific keywords.

Before this use case can be initiated, the creator has already connected to the Twitter Summariser Website.

1. The creator enters the Twitter Summariser Website.
2. The system presents the creator with interactive reports generated periodically based on specific keywords.
3. The system presents the creator a choice to publish report.

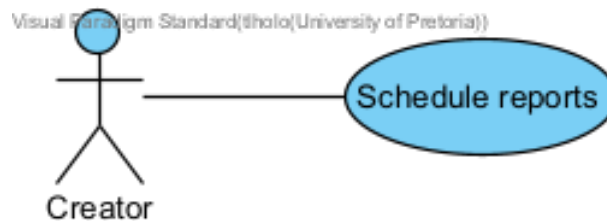


Figure 9: Use case diagram for schedule reports

UC9: Publish generated reports The Creator can publish generated reports.

Before this use case can be initiated, the creator has already connected to the Twitter Summariser Website.

1. The creator selects *Reports*
2. The system presents the creator with generated reports.
3. The creator selects a report
4. The system presents the creator with the selected report.
5. The system presents a choice of publishing a report.
6. The creator chooses to publish report.
7. The system makes the report public to the Twitter Summariser community.

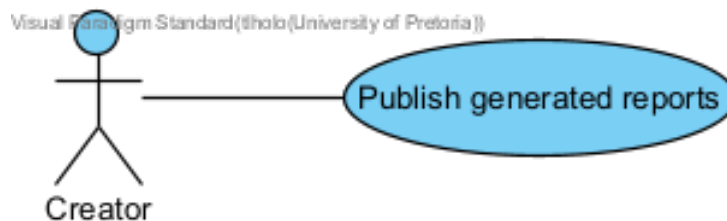


Figure 10: Use case diagram for publish generated reports

UC10: Explore published reports The Creator can explore published reports.

Before this use case can be initiated, the creator has already connected to the Twitter Summariser Website.

1. The creator selects *Explore*
2. The system returns published reports

4.2 Requirements

FR1: Twitter Summariser must allow a creator to search for topic using a keyword or term input to generate report

- The creator must be able to filter search results by time-range, non-replies, etc.
- The creator must be able to sort search results by most replies, most retweets, etc.

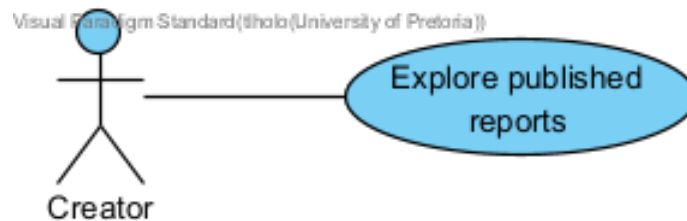


Figure 11: Use case diagram for explore published reports

FR2: Twitter Summariser must allow a creator to save generated report draft

FR3: Twitter Summariser must allow a creator to edit generated reports

- The creator must be able to add custom text to generated report.
- The creator must be able to choose custom styles for the generated report.

FR4: Twitter Summariser must allow a creator to publish generated reports

FR5: Twitter Summariser must allow a creator to share generated reports

- The creator must be able to share reports with other creators.
- The creator must be able to clone shared reports
- The creator must be able to edit shared reports

FR6: Twitter Summariser must allow a creator to clone shared reports

FR7: Twitter Summariser must allow a creator to edit shared reports

FR8: Twitter Summariser must allow a creator to schedule periodic reports for specific keywords

- The creator must be able to specify topic to generate periodic reports for.
- The creator must be able to choose when to generate periodic reports.
- The creator must be able to publish scheduled reports.

FR9: Twitter Summariser must allow a creator to manage previously created reports.

- The creator must be able to delete previously generated reports.
- The creator must be able to edit previously generated reports.
 - The creator must be able to add custom text to previously created report.
 - The creator must be able to add/change custom style to previously created report.

FR10: Twitter Summariser must allow a new creator to create an account

FR11: Twitter Summariser must allow an existing creator to log into their account

FR12: Twitter Summariser must allow a creator to explore published reports

FR13: Twitter Summariser must allow a reader to explore published reports

4.3 Subsystems

	Subsystem	Functional Requirement(s)
SS1	Collaboration	FR5; FR6; FR7
SS2	Management	FR2; FR3; FR8; FR9
SS3	Generation	FR1
SS4	Publication	FR4; FR12; FR13
SS5	Authentication	FR10; FR11

ual Paradigm Standard(thob(University of Pretoria))

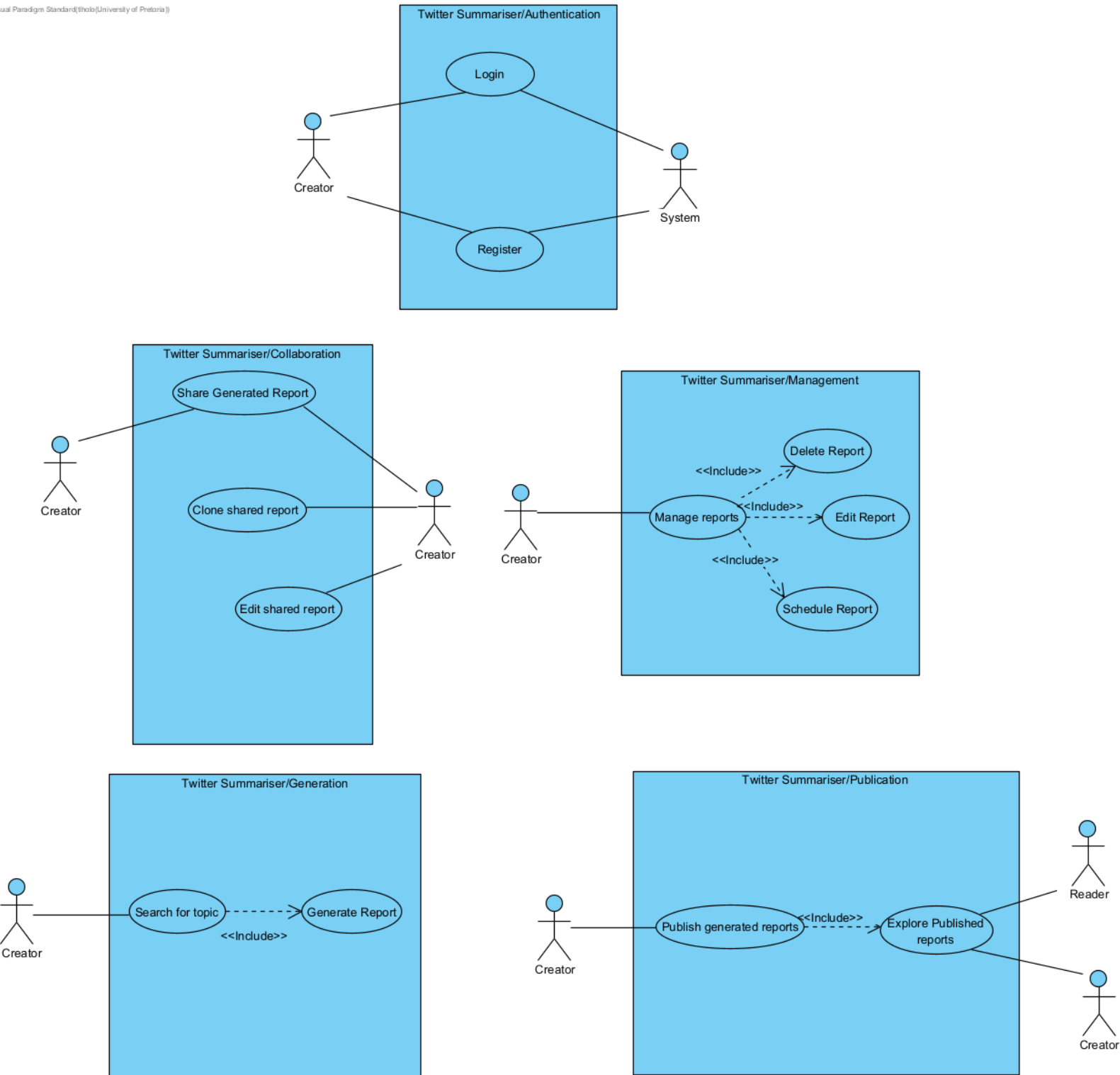


Figure 12: Overall Use Case Diagrams, Grouped into SubSystems

5 Quality Requirements

5.1 Performance

The responsiveness of the user interface must be almost immediate. An excellent response time for the service layer to deliver the report is expected.

5.2 Modifiability

The system must be developed in a way that it can be expandable to add more features to the system.

5.3 Security

The system must protect the sensitive data of a system's users. The system must maintain the integrity of the system's users data. All the system's users data must be made available for legitimate use.

5.4 Availability

A requirement of the system will and must be functional and available for service at all times. The system must be up for about 99% of the time.

5.5 Usability

The user interface of the software must be easy to use. It will be usable with no need of tutorial, but users will be guided through tool tips.

5.6 Integrability

Since the system consists of different parts, it has to be assured that the different parts of the system can work together.

6 Trace-ability Matrix

		Subsystem				
		Collaboration	Management	Generation	Publication	Authentication
Requirement	Priority					
FR01	1			X		
FR02	3		X			
FR03	4		X			
FR04	3				X	
FR05	3	X				
FR06	3	X				
FR07	4	X				
FR08	5		X			
FR09	3		X			
FR10	2					X
FR11	2					X
FR12	3				X	
FR13	3				X	
QR5.1	-		X	X		
QR5.2	-	X		X		
QR5.3	-		X			X
QR5.4	-		X		X	
QR5.5	-	X	X		X	
QR5.6	-	X				