Content Based Video Retrieval System

By

Mehwish Siddique

Supervised By

Dr. Umer Rashid

Department of Computer Science,

Quaid-i-Azam University, Islamabad, Pakistan.

Session (2015 – 2019)

ACKNOWLEDMENT

All the praises, thanks and acknowledgments are for the creator Allah Almighty, the most Beneficent, the most Merciful, who gave me strength and enabled me to undertake and execute this task. I am thankful to my parents for their prayers and support during my educational career.

I would like to express my gratitude to my Project Supervisor, **Dr. Umer Rashid** for his advice, guidance and patience throughout the development of the project. I am sincerely grateful to him for sharing his truthful and illuminating views on a number of issues related to the project.

Mehwish Siddique

ABSTRACT

Content based video retrieval is an approach for facilitating the searching and browsing of video collections based on content i.e. text, image. It includes methods such as Indexing, Searching and Retrieval, and Ranking of video results.

Traditional video retrieval methods fail to meet technical challenges due to large and rapid growth of multimedia data, demanding effective retrieval systems. Therefore, a more efficient method for video retrieval within large video archives is needed. The existing systems relay mainly on textual search. Among them, content-based video retrieval is a challenging and important problem of practical value. It can help users to retrieve desired video from video dataset efficiently based on the video contents.

Content Based Video Retrieval System is a desktop application that is developed to retrieve videos using multimodal query i.e. user can search the video dataset using textual or visual query. Extracted Features from the uploaded query are matched with stored content in video index (divided in textual, visual) and then relevant videos are retrieved. Retrieved Videos for monomodal search are ranked whereas for multimodal search they are merged and then ranked. Ranked Results are displayed to the user.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

CBVR	Content Based Video Retrieval
CBVRS	Content Based Video Retrieval System
UC	Use Case
UCD	Use Case description
SSD	System Sequence Diagram
SPMP	System Project Management Plan
SRS	Software Requirements Specifications
SDD	Software Design Description
STD	Software Test Documentation
TC	Test cases
WWW	World Wide Web
TCP/IP	Transmission Control Protocol/Internet Protocol
HTTP	Hyper Text Transfer Protocol
RS	Result Set

CHAPTER #1
SOFTWARE PROJECT MANAGEMENT PLAN

1.1 INTRODUCTION

This chapter describes roles, responsibilities, processes and schedules for managing the software development process, in addition to outlining the tools, methods, and procedures to be used.

1.1.1 Project Overview

Content Based Video Retrieval System is a desktop application that will be developed to retrieve videos using multimodal query i.e. user can search the video dataset using textual or visual query. Extracted Features from the uploaded query are matched with stored content in video index (divided in textual, visual) and then relevant videos are retrieved. Retrieved videos for monomodal search are ranked whereas for multimodal search they are merged and then ranked. Ranked Results are displayed to the user.

1.1.2 Project Deliverables

Deliverable is a tangible output of human effort provided by a developer to a customer. These deliverables are delivered to the project supervisor. The deliverables for the project "Content Based Video Retrieval system" are as follows:

Document Deliverables	**Description**	**DELIVERABLE**
Software Project Management Plan (SPMP)	Description of the software approach and associated milestones	1st Deliverable
Software Requirements specification (SRS)	Description of expected software features, constraints, interfaces and other attributes	1st Deliverable
Software Design Description (SDD)	Description of how the software will meet the requirements. Also describes the rationale of design decisions taken.	2nd Deliverable

Software Test Documentation (STD)	Description of the plan and specifications to verify and validate the software and the results.	2nd Deliverable
Implementation	Implementation of the project.	3rd Deliverable

Table 1.1 Project Deliverables

1.1.3 Problem Statement

Traditional video retrieval methods fail to meet technical challenges due to large and rapid growth of multimedia data, demanding effective retrieval systems. Therefore, a more efficient method for video retrieval within large video archives is needed. The existing systems relay mainly on textual search. Among them, content-based video retrieval is a challenging and important problem of practical value. It can help users to retrieve desired video from video dataset efficiently based on the video contents.

1.1.4 Purpose of Project

Most of the Retrieval Systems allow the user for textual search. There is a need for effective retrieval of videos based on multimodal search through its contents. So CBVRS will facilitate user for multi-search using text, visuals.

1.1.5 Objective

The main objectives of this project are:

- Development of an interactive system which provides users with a simple and convenient interface for searching in video dataset using various methods.
- It will provide multimodal query search (textual, visual) for more effective video retrieval.
- It will help users to retrieve desired video from a video index efficiently based on the video contents
- It will merge the results in case of multi-query search i.e. user searches using different search methods
- It provides ranked results.

1.2 PROJECT ORGANIZATION

This section details the architecture of the project, including: the process model, organizational structure, interfaces, and project responsibilities.

1.2.1 Software Process Model

For Content Based Video Retrieval System, Incremental process model will be used because

- This model can be used when the requirements of the complete system are clearly defined and understood.
- Increments will help getting an early review of the product.
- It is easier to test and debug during a smaller iteration.
- When software engineering team are not very well skilled or trained i.e. some risk features involved.

1.2.2 Roles and Responsibilities

I am responsible for developing Project Plan, status reporting, developing the software product, and making sure the project is delivered on schedule and within scope.

The Project Supervisor assists with major issues, problems, and policy conflicts and removes obstacles.

1.2.3 Tools

Following are the tools and techniques used for this project.

Sr.	**Tools**	
1	MS Word 2016	Used for documentation purposes.
2	Project Libre 1.6.2	Used to make project plan.
3	Eclipse	Used for implementation

Table 1.2 Tools

1.3 PROJECT MANAGEMENT PLAN

This section describes how the project will be managed, what are its tasks, deliverables, milestones etc.

1.3.1 Tasks

Following is the list of tasks for the project:

1.3.1.1 Problem Understanding

- **Description:**

  First problem definition is must.

- **Resources Needed:**

  The following are the resources required to complete the task:

  *People***:** Student, Supervisor

- **Dependencies and Constraints:**

  None

- **Risks and Contingencies**

  None

1.3.1.2 Software Project Management Planning

- **Description:**

  Secondly software approach and milestones are identified.

- **Deliverables and Milestones:** SPMP

- **Resources Needed:**

  The following are the resources required to complete the task:

  People: Student, Supervisor Software:

  MS Word, Project Libre

- **Dependencies and Constraints:**

  Problem Understanding

- **Risks and Contingencies:**

  Problem Understanding not clear.

1.3.1.3 Software Requirement Specification

- **Description:**

  Thirdly analysis on how the requirements will meet is included.

- **Deliverables and Milestones:**

  SPMP and SRS Document.

- **Resources Needed:**

  The following are the resources required to complete the task:

  People: Student, Supervisor

  Software: MS Word, Argouml

- **Dependencies and Constraints:**

  SPMP and Problem

  Understanding

- **Risks and Contingencies:**

  Problem Understanding and SPMP not clear

1.3.1.4 Software Design Description

- **Description:**

  Fourthly detailed design and interface design will be included.

- **Deliverables and Milestones:**

  SDD

- **Resources Needed:**

  The following are the resources required to complete the task:

  People: Student, Supervisor

  Software: MS Word, Argouml

- **Dependencies and Constraints:**

  Analysis and Requirement.

1.3.1.5 Software Test Documentation

- **Description:**

In this part test plans and test case to verify and validate the system and their expected results will be included

- **Deliverables and Milestones:** Software Test Documentation
- **Resources Needed:** People: Student, Supervisor

  Software: MS Word
- **Dependencies and Constraints:** SDD
- **Risks and Contingencies:**

  Requirement and Analysis not clear.

1.3.1.6 Software Implementation

- **Description:**

  How the system will be implemented.
- **Deliverables and Milestones:**

  Complete Implementation including source code.
- **Resources Needed:**

  People: Student

  Software: Matlab, Eclipse
- **Dependencies and Constraints:**

  Requirements, SDD and STD
- **Risks and Contingencies:**

  System not clear or no full understanding.

1.3.2 Timetable and Gantt Chart

Following is time table of the project which is divided into task and subtask

	Name	Duration	Start	Finish	Predecessors	Resource Names
	Content Based Video Retrieval System	162 days	**12/3/18 8:00 AM**	**7/16/19 5:00 PM**		
	Problem understanding	1 day	12/3/18 8:00 AM	12/3/18 5:00 PM		
	Make Software Project Management Plan	5 days	**12/4/18 8:00 AM**	**12/10/18 5:00 PM**	**2**	**Mehwish Siddique;PC;MS Word**
	Write Introduction	1 day	12/4/18 8:00 AM	12/4/18 5:00 PM		
	Define Project Organization	2 days	12/5/18 8:00 AM	12/6/18 5:00 PM		
	Define Project Management Plan	2 days	12/7/18 8:00 AM	12/10/18 5:00 PM	5	Project Libre
	Make Requirements document	56 days	**12/11/18 8:00 AM**	**2/26/19 5:00 PM**	**6**	**Mehwish Siddique;PC;MS Word**
	Make Software Requirement Specification Document	24 days	**12/11/18 8:00 AM**	**1/11/19 5:00 PM**		
	Give Introduction and Overview	1 day	12/11/18 8:00 AM	12/11/18 5:00 PM		
	Define Scope	1 day	12/12/18 8:00 AM	12/12/18 5:00 PM		
	Define Purpose and objective	1 day	12/13/18 8:00 AM	12/13/18 5:00 PM		
	Identify Specific Requirements	1 day	12/14/18 8:00 AM	12/14/18 5:00 PM		
	Explain External Interfaces	1 day	12/17/18 8:00 AM	12/17/18 5:00 PM	12	
	Identify Use Cases	3 days	12/18/18 8:00 AM	12/20/18 5:00 PM	13	
	Make UseCase Diagram	1 day	12/21/18 8:00 AM	12/21/18 5:00 PM	14	Argo Uml
	Define UseCase descriptions	2 days	12/24/18 8:00 AM	12/25/18 5:00 PM	15	
	Define System Attributes	2 days	12/26/18 8:00 AM	12/27/18 5:00 PM	16	
	Make System Sequence Diagrams	2 days	1/3/19 8:00 AM	1/4/19 5:00 PM		Argo Uml
	Review and refine SSD	1 day	1/7/19 8:00 AM	1/7/19 5:00 PM	18	Dr Umer Rashid
	Make Domain Model	2 days	1/8/19 8:00 AM	1/9/19 5:00 PM	18	Argo Uml
	Review and Refine SRS	2 days	1/10/19 8:00 AM	1/11/19 5:00 PM	20	Dr Umer Rashid
	Make Software Design Description Document	16 days	**1/14/19 8:00 AM**	**2/4/19 5:00 PM**	**21**	**Mehwish Siddique;PC;MS Word**
	Give Introduction and Overview	1 day	1/14/19 8:00 AM	1/14/19 5:00 PM		
	Make Activity Diagrams	4 days	1/14/19 8:00 AM	1/17/19 5:00 PM		Argo Uml
	Review and Refine Activity Diagram	2 days	1/18/19 8:00 AM	1/21/19 5:00 PM	24	Dr Umer Rashid
	Make System Architectural Design	2 days	1/21/19 8:00 AM	1/22/19 5:00 PM		Argo Uml
	Review and Refine Architecture Diagram	1 day	1/23/19 8:00 AM	1/23/19 5:00 PM	26	Dr Umer Rashid
	Make Sequence Diagrams	2 days	1/24/19 8:00 AM	1/25/19 5:00 PM	26;27	Argo Uml
	Review and Refine SD	1 day	1/28/19 8:00 AM	1/28/19 5:00 PM	28	Dr Umer Rashid
	Identify Classes	1 day	1/29/19 8:00 AM	1/29/19 5:00 PM	29	
	Make Class Diagram	2 days	1/30/19 8:00 AM	1/31/19 5:00 PM	30	Argo Uml

Figure 1.1 Project Timetable(a)

		Name	Duration	Start	Finish	Predecessors	Resource Names
25		Review and Refine SRS	2 days	1/10/19 8:00 AM	1/11/19 5:00 PM	24	Dr Umer Rashid
26		**Make Software Design Description Document**	16 days	**1/14/19 8:00 AM**	**2/4/19 5:00 PM**	**25**	**Mehwish Siddique;PC;MS Word**
27		Give Introduction and Overview	1 day	1/14/19 8:00 AM	1/14/19 5:00 PM		
28		Make Activity Diagrams	4 days	1/14/19 8:00 AM	1/17/19 5:00 PM		Argo Uml
29		Review and Refine Activity Diagram	2 days	1/18/19 8:00 AM	1/21/19 5:00 PM	28	Dr Umer Rashid
30		Make System Architectural Design	2 days	1/21/19 8:00 AM	1/22/19 5:00 PM		Argo Uml
31		Review and Refine Architecture Diagram	1 day	1/23/19 8:00 AM	1/23/19 5:00 PM	30	Dr Umer Rashid
32		Make Sequence Diagrams	2 days	1/24/19 8:00 AM	1/25/19 5:00 PM	30;31	Argo Uml
33		Review and Refine SD	1 day	1/28/19 8:00 AM	1/28/19 5:00 PM	32	Dr Umer Rashid
34		Identify Classes	1 day	1/29/19 8:00 AM	1/29/19 5:00 PM	33	
35		Make Class Diagram	2 days	1/30/19 8:00 AM	1/31/19 5:00 PM	34	Argo Uml
36		Review and Refine Class Diagram	1 day	2/1/19 8:00 AM	2/1/19 5:00 PM	35	Dr Umer Rashid
37		Review and Refine Software Design Description	1 day	2/4/19 8:00 AM	2/4/19 5:00 PM	36	Dr Umer Rashid
38		**Make User Manual**	8 days	**2/5/19 8:00 AM**	**2/14/19 5:00 PM**	**37**	**Mehwish Siddique;PC;MS Word**
39		Select Tools and Technologies	1 day	2/5/19 8:00 AM	2/5/19 5:00 PM		
40		Make User Interfaces	3 days	2/6/19 8:00 AM	2/8/19 5:00 PM	39	
41		Give Description of UI	2 days	2/11/19 8:00 AM	2/12/19 5:00 PM	39	
42		Review and Refine UI	2 days	2/13/19 8:00 AM	2/14/19 5:00 PM	41	Dr Umer Rashid
43		**Make Software Test Document**	4 days	**2/15/19 8:00 AM**	**2/20/19 5:00 PM**	**42**	
44		Make Test Cases	3 days	2/15/19 8:00 AM	2/19/19 5:00 PM		MS Word
45		Review and Refine Test Document	1 day	2/20/19 8:00 AM	2/20/19 5:00 PM	44	Dr Umer Rashid
46		Review Analysis and Design Document	3 days	2/21/19 8:00 AM	2/25/19 5:00 PM	42;45	Dr Umer Rashid
47		Provide 1st Deliverable	1 day	2/26/19 8:00 AM	2/26/19 5:00 PM	46	
48		Project Implementation	100 days	2/27/19 8:00 AM	7/16/19 5:00 PM	47	Matlab 2018;Mehwish Siddique;Dr U...

Figure 1.2 Project Timetable(b)

Figure 1.3 Gantt Chart

CHAPTER #2

SOFTWARE REQUIREMENT SPECIFICATION

2.1 INTRODUCTION

It outlines the major functions and objectives of the system and the major constraints associated with system. In short, we can say that this chapter provides an overview of the proposed system.

2.1.1 Purpose

The purpose of this SRS document is to provide a detailed overview of software product, its parameters and goals. It will illustrate the purpose and complete declaration for the development of system. This document describes the project's intended audience, all the functional as well as non-functional requirements. It also defines how the client, developer and audience see the product and its functionality.

2.1.2 Product Overview

CBVRS is implemented in Eclipse. This product will retrieve relevant videos based on the contents of the video for Multimodal search.

2.1.3 Scope

Content Based Video Retrieval System will allow users to get results for multimodal search that are most relevant based on its content.

- It allows the user for multimodal query search (textual, visual).
- Features are extracted from uploaded example and matched with content of videos in index (textual, visual)
- Relevant Results are retrieved from index in two separate sets i.e. textual and visual.
- Results are merged for multi-query search. i.e. user searches using more than one query (textual, visual).
- Results are then ranked.
- Ranked Results are then displayed to the user.
- User can view these results.

Inputs:

✦ Video dataset

✦ Textual query, Visual query for search

Outputs:

- ✦ Results displayed to user

Functionalities:

- ✦ Searching the relevant videos for given query
- ✦ Retrieving results matched with query
- ✦ Merging results in case of multimodal query
- ✦ Ranking Result of content-based video retrieval

2.1.4 Document Organization

In the coming chapters following aspects are described:

- Overall description of software.
- Specific requirements.
- Use case model.
- User Characteristics
- Non-Functional Requirements.

2.2 OVERALL DESCRIPTION

This section will give an overview of the whole system. The system will be explained in its context to show how the system interacts with other systems and introduce the basic functionality of it. It will also describe what type of stakeholders that will use the system and what functionality is available for each type. At last, the constraints and assumptions for the system will be presented.

2.2.1 Product Perspective

CBVRS is a desktop application that will index dataset using open source software. Retrieval, Merging and ranking is done by system based on multimodal query.

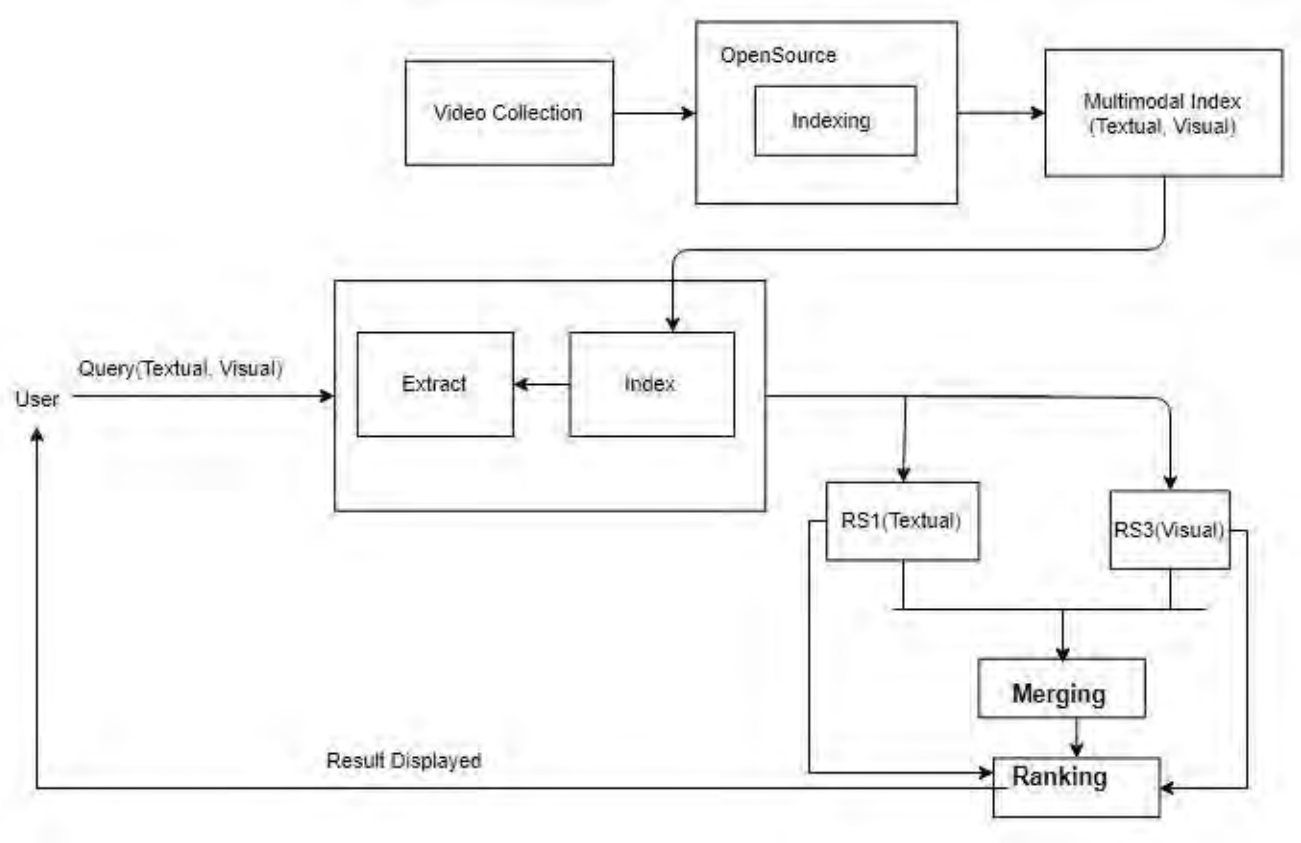


Figure 2.1 Block Diagram

2.2.2 User Characteristics

This tool is for all the users. User

is expected to be

- Educated
- Familiar with computer using
- Able to use a search engine.

2.2.3 General Constraints

The constraints are mentioned below:

1. Application must be installed on System.
2. Completion of this project by the end of spring semester 2019.

2.2.4 Assumption and Dependencies

1. The users should know the English language, as the interface will be provided in English.
2. User should know how to perform search.
3. User should know how to upload file or folders in system.

2.3 EXTERNAL INTERFACE REQUIRMENTS

It covers all the interfaces of the product.

2.3.1 User Interfaces

CBVRS should be designed for ease of use, providing help instructions. The background will be light in color with dark colored font to enhance the contrast and visibility.

Proper labels will be defined for user input fields. User of system is provided an interface to perform search

2.3.2 Hardware Interfaces

Keyboard and mouse is used to input, and monitor is to display output. While minimum hardware requirements are:

- Minimum processor Core i3

2.3.3 Software Interfaces

- Operating System: Windows 10
- Matlab 2018
- Eclipse

2.3.4 Communications Interfaces

As the system is a Desktop based Application so requires no Communication interfaces.

2.4. Software Product Features

Summary of major functions that the system should perform are listed below.

Query Input: This field will allow user to perform multimodal query search. User can search as a text or visual query.

Retrieve Videos: CBVRS will retrieve videos from multimodal index after matching extracted features from uploaded and existing videos.

Merging: Merging retrieved videos of multiple query search.

Ranking Results: Retrieved videos are then merged (only in case of multi search) and ranked.

Displaying Results: After ranking the search results are displayed on screen to user.

2.5 Use Cases

The use case diagram, list of use case and their description is as follows:

2.5.1 Use Cases List

- Upload Dataset
- Index video collection
  - Create Textual Index
  - Create Visual index
- Search using Monomodal
  - Search by text
  - Search by Visual
- Search using Multimodal
  - Search by Text and Visuals
- View Results
  - View Textual Search Results
  - View Visual Search Results
  - View Text and Visual Search Results

2.5.2 Usecases Diagram

Following is the use case diagram of the system:

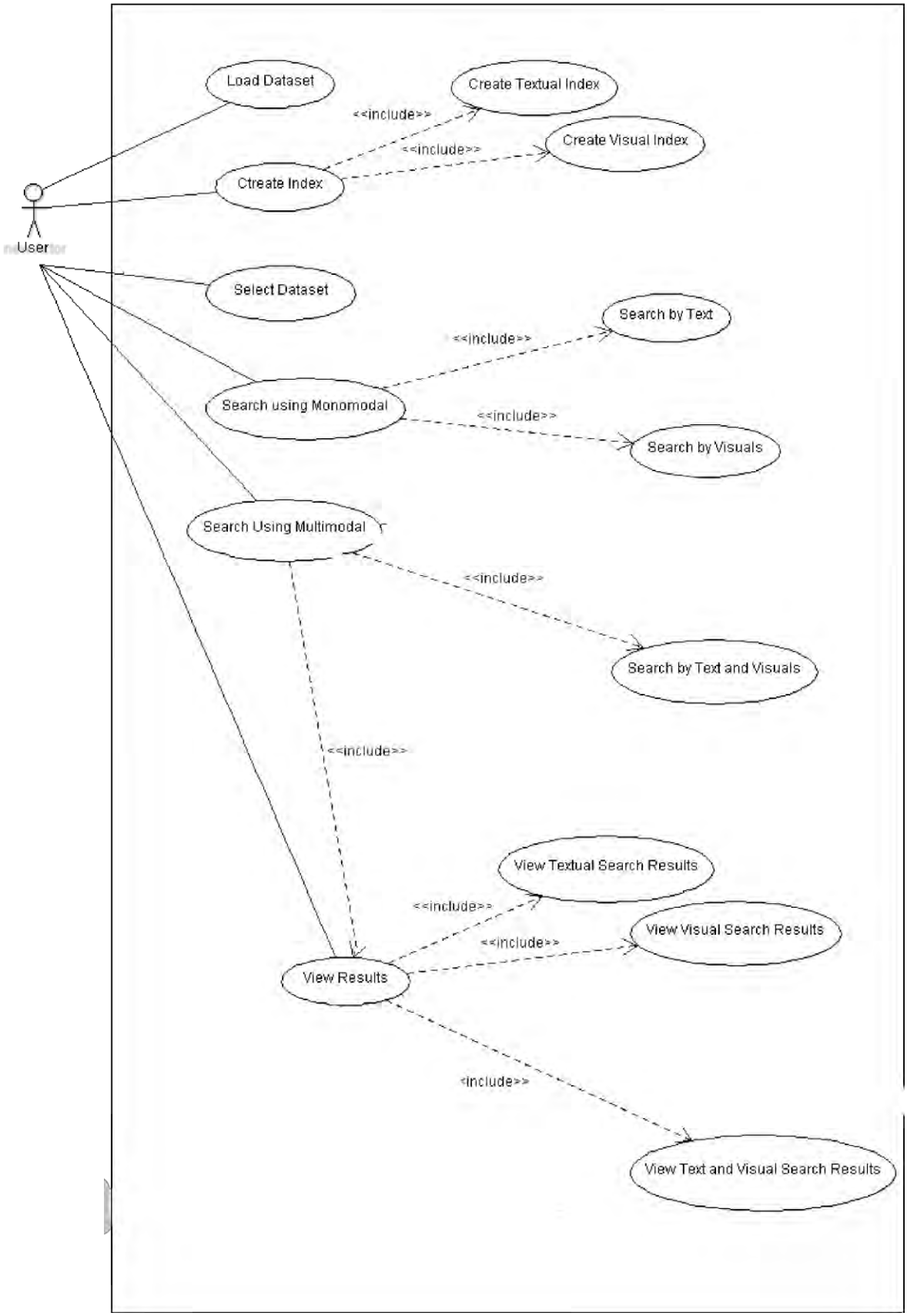


Figure 2.2 Usecase Diagram

2.5.3 Usecases Details

The use case details include primary actor, pre and post conditions, main and alternative scenarios.

UC-1: Upload Dataset	
Primary actor	User (Anyone can use this Application)
Pre-condition	Application must be opened with home interface at front and Dataset must exist on System
Post-condition	Dataset loaded into CBVRS and then can be indexed
Description	Load dataset Usecase will be used so that user can load dataset into system to perform indexing on it.
Main Success Scenario	1) User clicks on Load Dataset Button 2) User Browses the dataset and Selects it. 3) User clicks ok. 4) Dataset loaded in CBVRS and ready for indexing
Alternative Scenario	-
Frequency	Can be done once or multiple times.

Table 2.1 UC-1 Upload Dataset

UC-3: Create index	
Primary actor	User (Anyone can use this Application)
Pre-condition	Dataset must be selected to index
Post-condition	Dataset indexed and stored in database

Description	Index video collection UC will be used to index the dataset for efficient search by user.
Main Success Scenario	1) User clicks on Create Index for selected dataset 2) User selects Index type, include (Create Textual Index, Create Visual Index) 3) System creates index for selected type and stores it in database.
Alternative Scenario	-
Frequency	Can be done once or multiple times.

Table 2.2 UC-3 Create Index

UC-4: Create Textual index	
Primary actor	User (Anyone can use this Application)
Pre-condition	Create Index option is selected.
Post-condition	Dataset indexed and stored in database
Description	Create Textual Index UC will be used to index the dataset for efficient search by user.
Main Success Scenario	1) User clicks on Create textual index button. 2) System creates textual index for selected dataset and stores it in database.
Alternative Scenario	-
Frequency	Can be done once or multiple times.

Table 2.3 UC-4 Create Textual Index

UC-5: Create Visual index	
Primary actor	User (Anyone can use this Application)
Pre-condition	Create Index option is selected.
Post-condition	Dataset indexed and stored in database
Description	Create Visual Index UC will be used to index the dataset for efficient search by user.
Main Success Scenario	1) User clicks on Create Visual index button. 2) System creates visual index for selected dataset and stores it in database.
Alternative Scenario	-
Frequency	Can be done once or multiple times.

Table 2.4 UC-5 Create Visual Index

UC-6: Search Using Monomodal	
Primary actor	User (Anyone can use this Application)
Pre-condition	Interface containing search options displayed to user.
Post-condition	Interface leading to different query option for monomodal displayed to user
Description	Search using Monomodal UC will be used to search the indexed video collection using different query options.

Main Success Scenario	1) User selects the search using monomodal option. 2) System opens next interface with multiple search options, include (Search using Text, Search using Visual data)
Alternative Scenario	-
Frequency	Can be done once or multiple times.

Table 2.5 UC-6 Search Using Monomodal

UC-7: Search by Text	
Primary actor	User (Anyone can use this Application)
Pre-condition	Search using text option is selected from monomodal search
Post-condition	Results for textual search retrieved and displayed.
Description	Search using text UC will be used to search the indexed video collection using text query.
Main Success Scenario	1) User selects the text field 2) User enters text query to search video collection. 3) User clicks search button. 4) System retrieves results for textual query and displays them to user.
Alternative Scenario	3a) Textual index not created for video collection. 1. Create Textual index for selected dataset.
Frequency	Can be done once or multiple times.

Table 2.6 UC-7 Search by text

UC-8: Search by Visuals	
Primary actor	User (Anyone can use this Application)
Pre-condition	Search using Visual data option is selected from monomodal search
Post-condition	Results for Visual search retrieved and displayed.
Description	Search using Visual data UC will be used to search the indexed video collection using Visual query.
Main Success Scenario	1) User selects the attachment/upload option 2) User attaches image or video clip file to search video collection. 3) User clicks search button. 4) System retrieves results for Visual query and displays them to user.
Alternative Scenario	2a) Image/Video clip file not found or does not exist 1) Download or place an image/video clip on computer before. 3a) Visual index not created for video collection. 3. Create Visual index for selected dataset.
Frequency	Can be done once or multiple times.

Table 2.7 UC-8 ***Search by Visuals***

UC-9: Search Using Multimodal	
Primary actor	User (Anyone can use this Application)
Pre-condition	Interface containing search options displayed to user.

Post-condition	Interface leading to different query option for multimodal displayed to user
Description	Search using Multimodal UC will be used to search the indexed video collection using different query options.
Main Success Scenario	1) User selects the search using multimodal option. 2) System opens next interface with multiple search options, include (Search using Text and acoustic data, Search using Acoustic and visual data, Search using Text and Visual data, Search using Text, Acoustic and Visual data)
Alternative Scenario	-
Frequency	Can be done once or multiple times.

Table 2.8 UC-9 ***Search Using Multimodal***

UC-10: Search by Text and Visuals	
Primary actor	User (Anyone can use this Application)
Pre-condition	Search using Text and Visual data option is selected from multimodal search
Post-condition	Results for Textual, Visual and combined search retrieved and displayed.
Description	Search using Text and Visual data UC will be used to search the indexed video collection using text and Visual query.

Main Success Scenario	1) User selects the text field 2) User enters text query. 3) User selects the attachment/upload option 4) User attaches image or video clip file to search video collection. 5) User clicks search button. 6) System retrieves results for both text and Visual query.
Alternative Scenario	4a) Image/Video clip file not found or does not exist 2) Download or place an image/video clip on computer before. 5a) Textual or Visual index not created for video collection. 4. Create textual and Visual index for selected dataset.
Frequency	Can be done once or multiple times.

Table 2.9 UC-10 ***Search by Text and Visuals***

UC-11: View search results	
Primary actor	User (Anyone can use this Application)
Pre-condition	Search must be performed.
Post-condition	Results displayed and viewed by user
Description	View Results UC will be used to view results for performed search
Main Success Scenario	1) User selects View Results Option 2) System displays corresponding view results option for performed search, include (View textual Results, view visual results, View Textual and visual results)

Alternative Scenario	-
Frequency	Can be done once or multiple times.

Table 2.10 UC-11 ***View search results***

UC-12: View Textual search results	
Primary actor	User (Anyone can use this Application)
Pre-condition	User clicks the View Textual results button.
Post-condition	Results displayed and viewed by user
Description	View textual Results UC will be used to view results for performed textual search.
Main Success Scenario	1) System Displays the retrieved and ranked results (i.e. videos) on screen. 2) User Views Video results for textual query.
Alternative Scenario	-
Frequency	Can be done once or multiple times.

Table 2.11 UC-12 ***View Textual search results***

UC-13: View Visual search results	
Primary actor	User (Anyone can use this Application)
Pre-condition	User clicks the View Visual results button.
Post-condition	Results displayed and viewed by user

Description	View Visual Results UC will be used to view results for performed Visual search.
Main Success Scenario	1) System Displays the retrieved and ranked results (i.e. videos) on screen. 2) User Views Video results for Visual query.
Alternative Scenario	-
Frequency	Can be done once or multiple times.

Table 2.12 UC-13 ***View Visual search results***

UC-14: View Textual and Visual search results	
Primary actor	User (Anyone can use this Application)
Pre-condition	User clicks the View Textual and Visual results button.
Post-condition	Results displayed and viewed by user
Description	View Textual and Visual Results UC will be used to view results for performed Textual and Visual search.
Main Success Scenario	1) System retrieves videos for both textual and Visual query, merges them and displays ranked results (i.e. videos) on screen. 2) User Views merged Video results for Textual and Visual query.
Alternative Scenario	-
Frequency	Can be done once or multiple times.

Table 2.13 UC-14 ***View Textual and Visual search results***

2.6 SYSTEM SEQUENCE DIAGRAMS

SSD-1: Load dataset

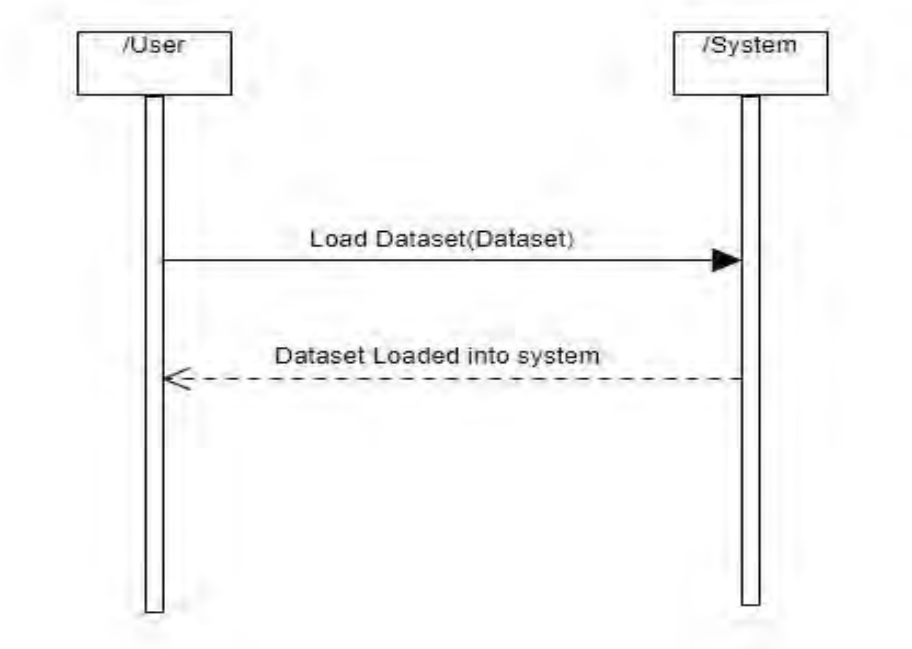


Figure 2.3 SSD For Load Dataset

SSD-2: Select Indexed Dataset

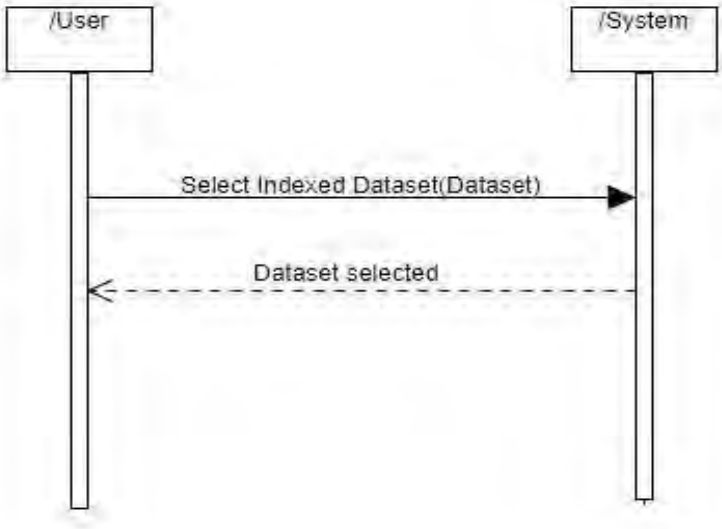


Figure 2.4 SSD for Select Indexed dataset

SSD-3: Create Index

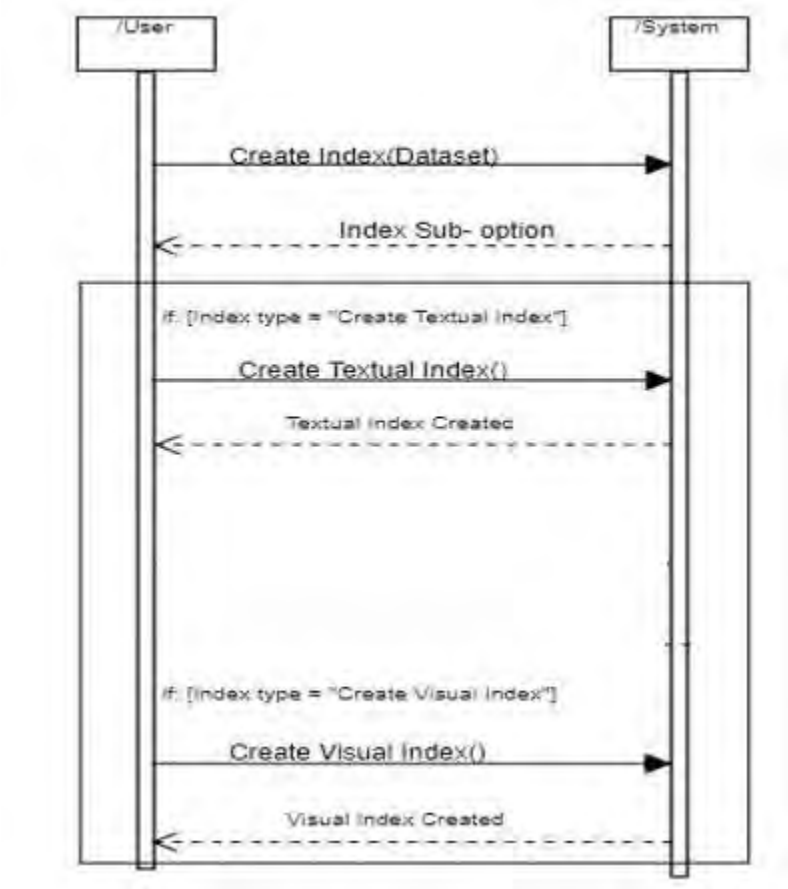


Figure 2.5 SSD for Create Index

SSD-

4: Search using Monomodal

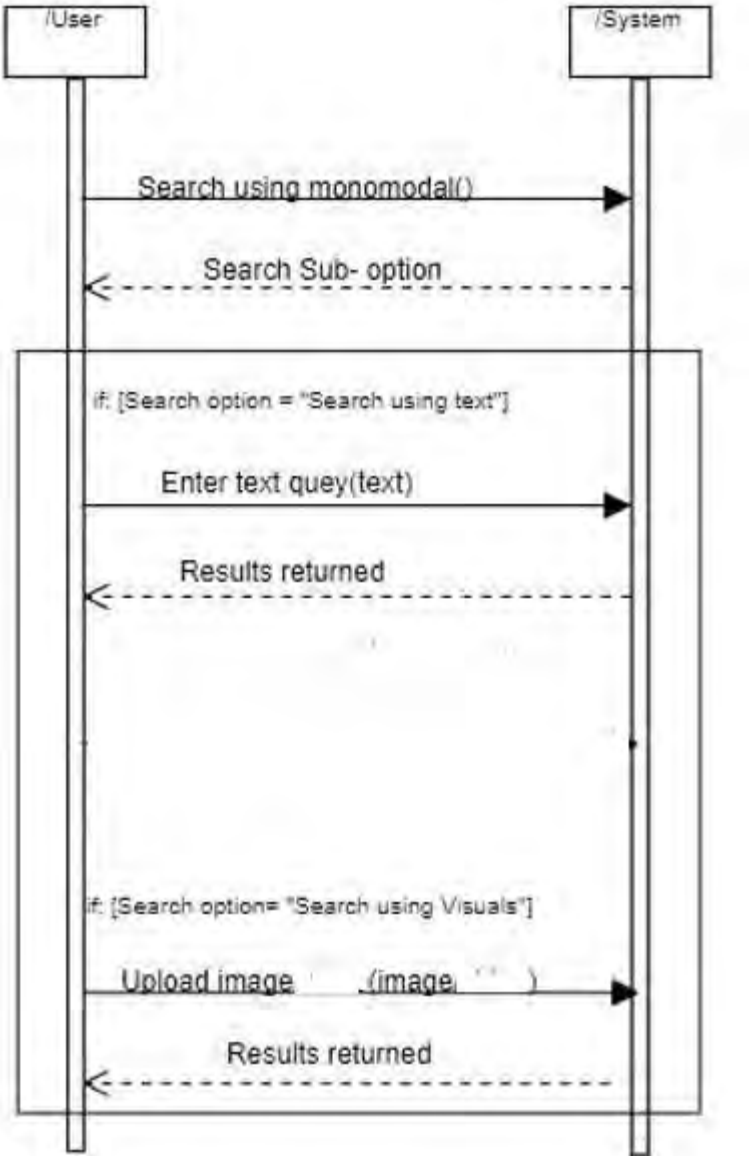


Figure 2.6 SSD Monomodal Search

SSD-

5: Search using Multimodal

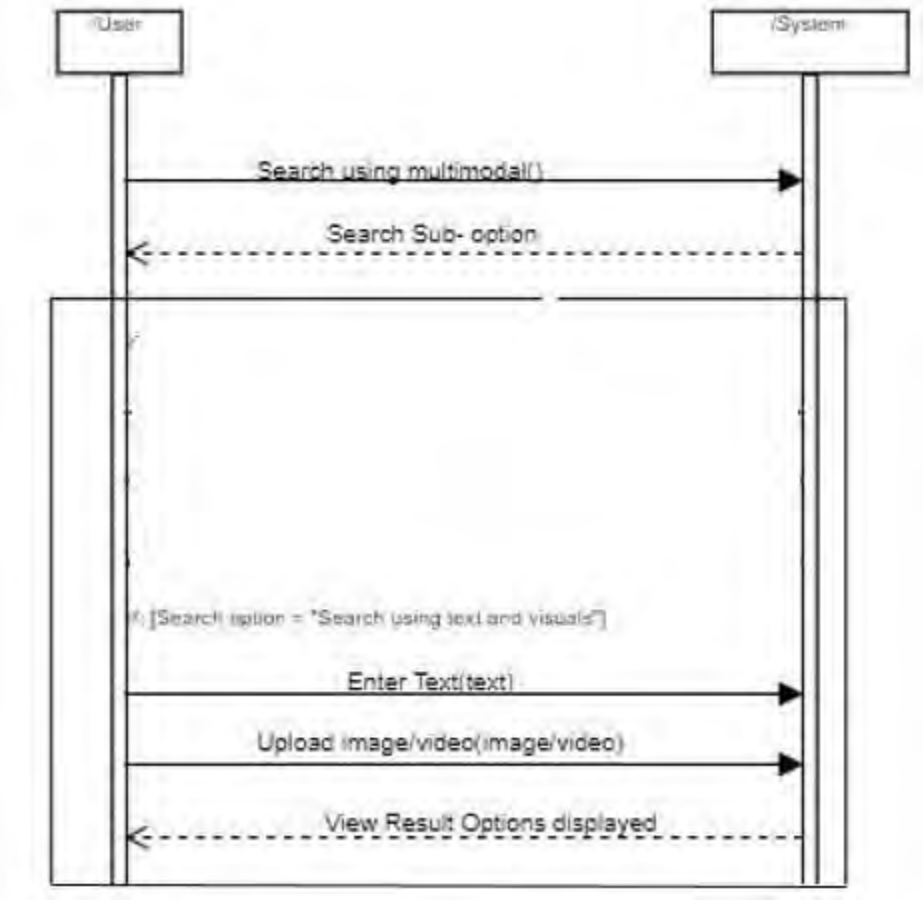


Figure 2.7 SSD Multimodal Search

SSD-

6: View Search Results

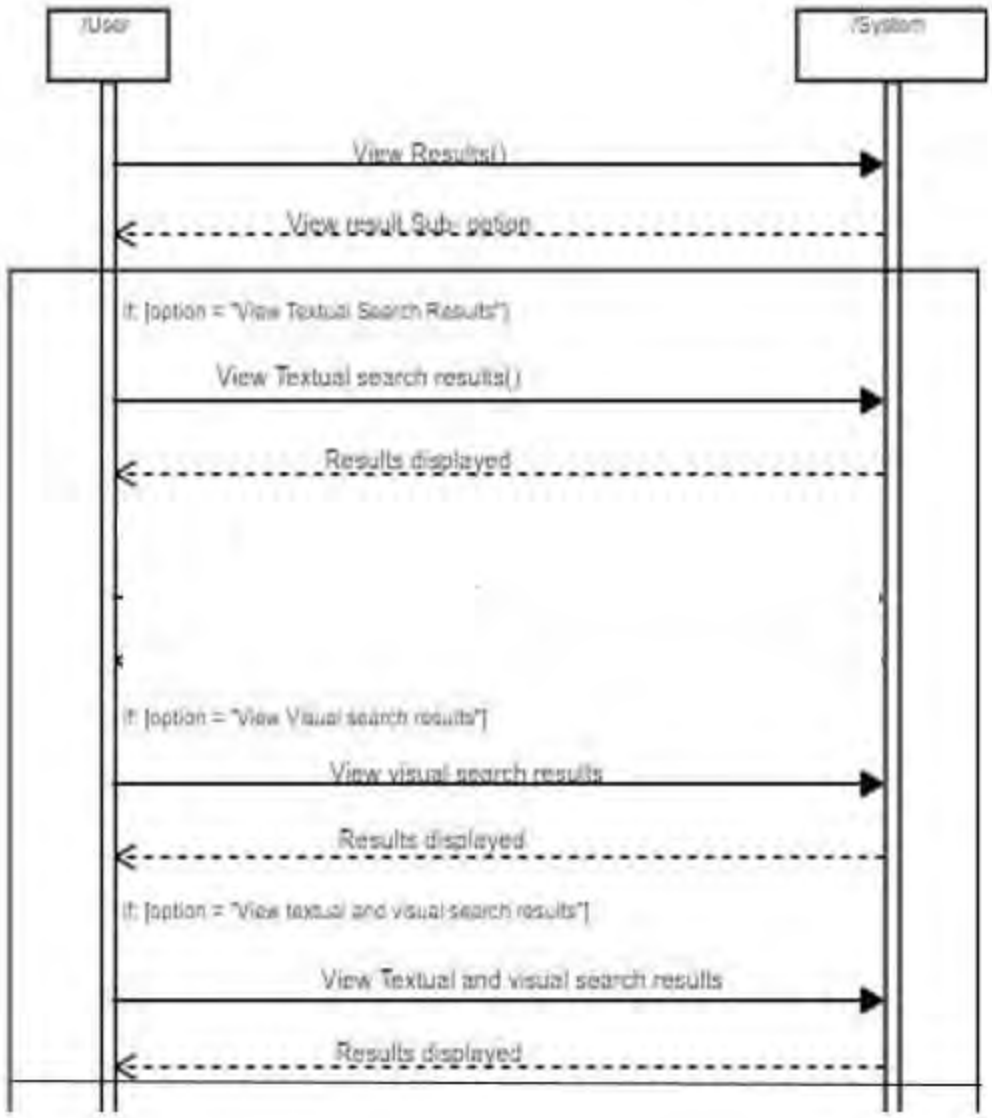


Figure 2.8 SSD for View Results

2.7 ACTIVITY DIAGRAM

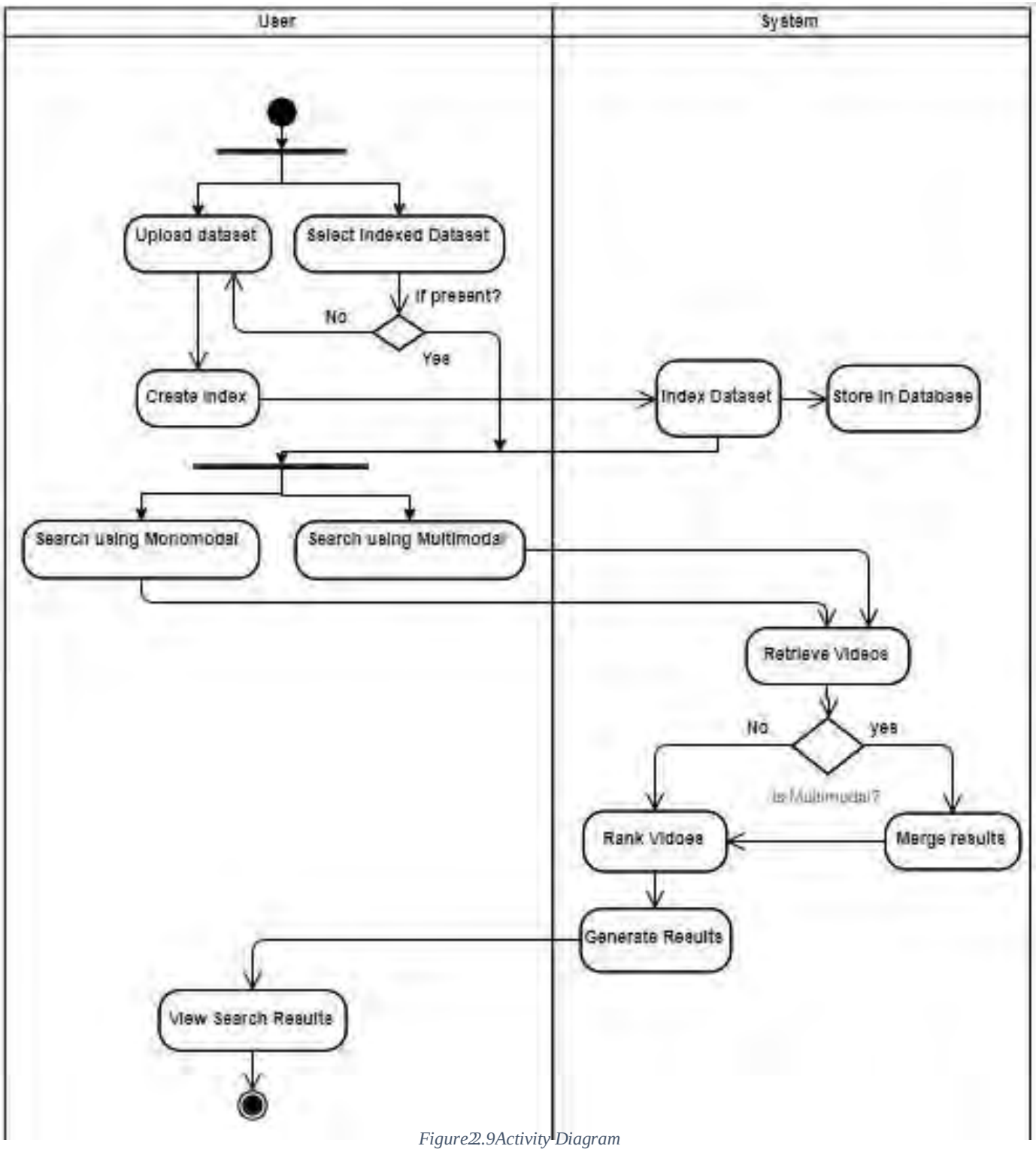


Figure2.9Activity Diagram

2.8 DOMAIN MODEL

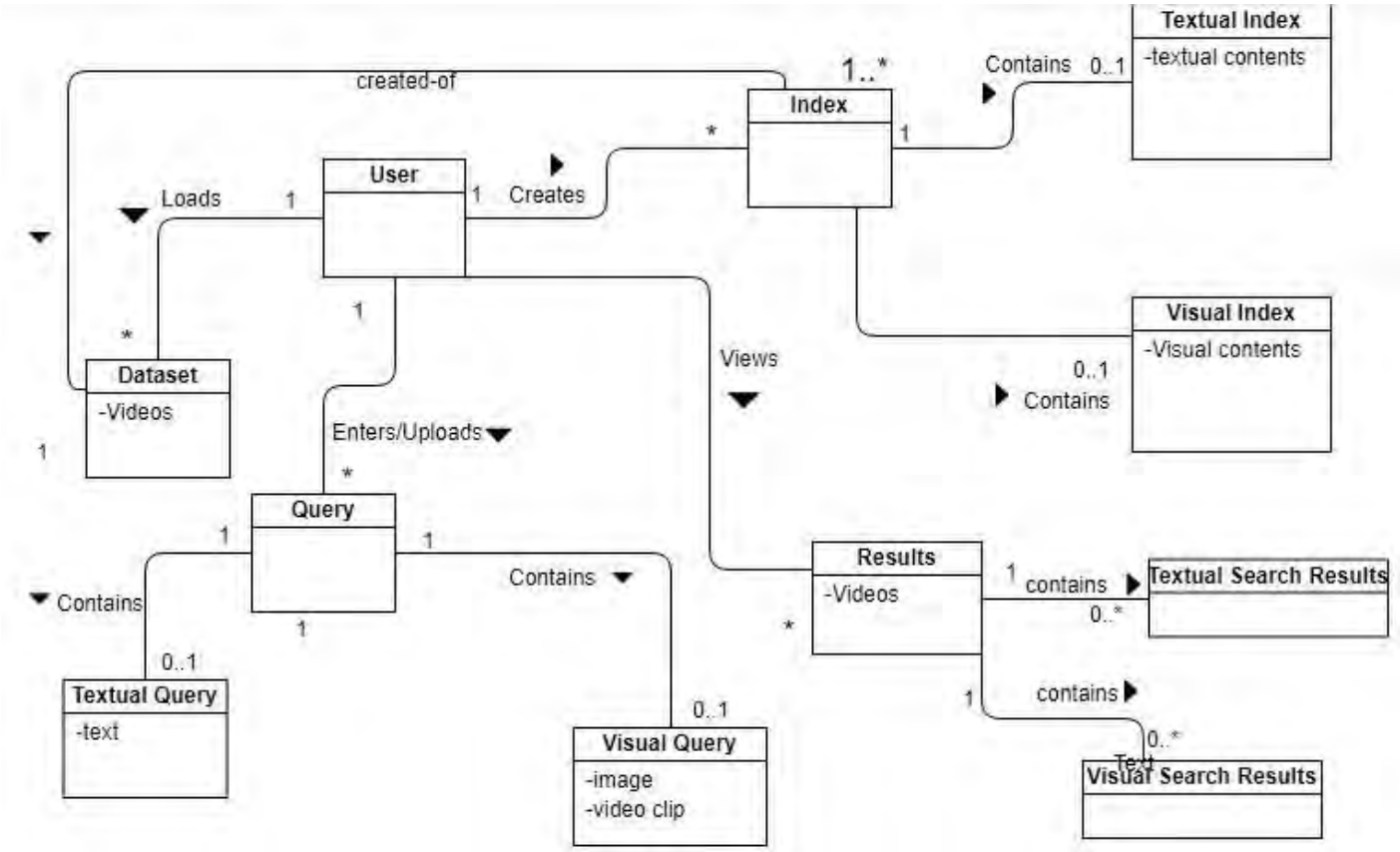


Figure 2.10 Domain Model

2.9 SOFTWARE QUALITY ATTRIBUTES

This section defines the non-functional requirements which are used to evaluate the performance of the system.

2.9.1 Availability

System should be available to the user when installed.

2.9.2 Maintainability

During the development period, all the code will be properly documented so that we can easily make changes and upgrade our application.

2.9.3 Reliability

The System should be reliable and does not counter any failure. System will use separation of concerns. The system should never crash or hang, other than as the result of an operating system error.

2.9.4 Portability

Since the application needs to be installed for use so it's portable with all Windows based PC's.

2.9.5 Security

When using the application, there shouldn't be any threats from other software/applications

2.10 Database Requirements

No Database required for this project.

CHAPTER #3
Literature Review and Existing Systems

What is Content Based Video Retrieval?

"Content based video retrieval is an approach for facilitating the searching and browsing of large image collections over World Wide Web. In this approach, video analysis is conducted on low level visual properties extracted from video frame. We believed that to create an effective video retrieval system, visual perception must be considered. We conjectured that a technique which employs multiple features for indexing and retrieval would be more effective in the discrimination and search tasks of videos. To validate this claim, content-based indexing and retrieval systems were implemented using color histogram, various texture features and other approaches". [2]

What is Content:

Content in this context refers to colors, shapes, textures, or any other information that can be derived from the image itself. An individual video possesses unique motion features, color histograms, motion histograms, text features, audio features, features extracted from faces and objects existing in its frames.

Monomodality:

Monomodality means communication is limited to one mode only. In our system Monomodal Search stands for searching using one query type or mode i.e. textual, acoustic or Visual.

Multimodality:

Multimodality describes communication practices in terms of the textual, aural, linguistic, spatial, and visual resources or modes, used to compose messages. Where media are concerned, multimodality is the use of several modes (media) to create a single artifact.

Multimodal interaction provides the user with multiple modes of interacting with a system, such as speech, text, images, video, audio, etc. A multimodal system allows for several distinct means for input and output of data. [5]

One critical task in content-based video retrieval is to rank search results with combinations of multimodal resources effectively.

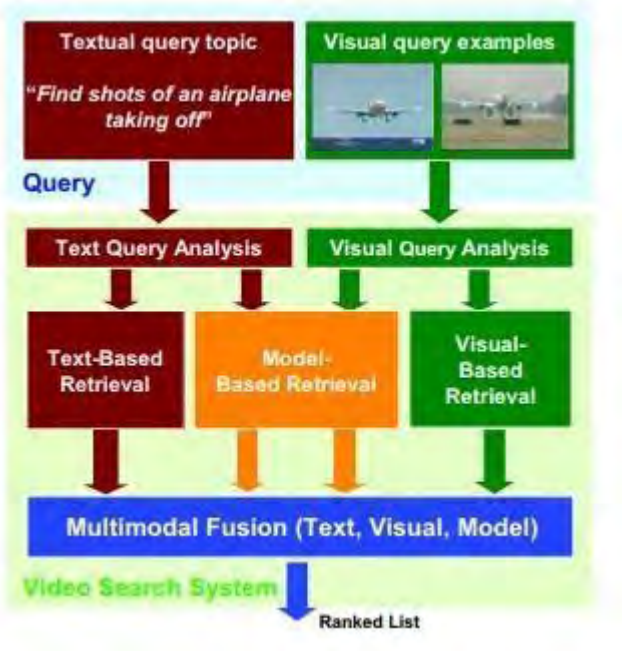


Figure 3.1 Multimodality

Indexing:

Text Indexing:

Text Indexing is the process of describing or classifying a document by index terms in order to indicate what the document is about.

Inverted Index:

It is a mapping from terms to documents. This is called an inverted index because it reverses the usual mapping of a document to the terms it contains.

In CBVRS, inverted index is generated by an open source Lucene, Lucene is a library in java that provides indexing and search over documents.

Image Indexing:

Image Indexing is done using the features such as colour, texture and shape. An open source tool Lucene Image Retrieval (LIRE) will be used for Indexing and Search using images. It allows indexing and search using multiples features of images. A feature used for image indexing is CEDD (Colour and Edge Directivity Descriptor)

CEDD: A low level feature extracted from images and can be used for indexing and retrieval incorporates color and texture information in a histogram.

Text Retrieval and Ranking:

The Boolean Model of Iinformation Retrieval is based on Boolean logic and classical set theory in that both the documents to be searched and the user's query are conceived as sets of terms. Retrieval is based on whether or not the documents contain the query terms.[7B]

Vector Space Model is used for text and image retrievals, In text retrieval "A collection of documents can be viewed as vectors in high dimensional vector space. Every term in the index language becomes an independent dimension. A document is reprented as n-dimensional vector."[7]

In Image Retrieval "Collection of images and queries are represented as feature vectors in an ndimensional vector space."[7]

- Searching for entered textual query in created index and ranking them using **scoredoc** function provided in Library.
- Scoring in Lucene combines Vector Space and Boolean Model:
- Idea behind VSM is the more times a query term appears in a document relative to the number of times the term appears in all the documents in the collection, the more relevant that document is to the query
- It uses the Boolean model to first narrow down the documents that need to be scored based on the use of boolean logic in the Query specification.
- Lucene scoring works on Fields and then combines the results to return Documents

Image Retrieval and Ranking:

- Searching for uploaded image query in created index and ranking them using score function provided in Library.
- Feature used is Color and Edge Directivity Descriptor.
- Lire uses Dissimilarity Search i.e. Two images having least difference in them are ranked highest.

Existing or Similar Systems:

1. IMOTION:

The IMOTION system is a multimodal content-based video search and browsing application offering a rich set of query modes based on a broad range of different features. It is able to scale with the size of the collection due to its underlying flexible polystore called ADAMpro and its very effective retrieval engine Cineast, optimized for multifeature fusion. IMOTION is simultaneously geared towards precision-focused searches, i.e., known-item search with image or text queries, and recall-focused, exploratory searches. [3]

The IMOTION project has developed and evaluate innovative multi-modal user interfaces for interacting with augmented videos. Starting with an extension of existing query paradigms (keyword search in manual annotations), image search (query by example in key frames), IMOTION will consider novel sketch- and speech-based user interfaces. In particular, novel types of motion queries will be supported where users can specify motion paths of objects, via sketches, gestures, natural language interfaces, or combinations thereof. Several types of user interfaces (voice, tablets, multi-touch tables, interactive paper) will be supported and seamlessly combined so as to smoothly migrate a session from one type of user interface to another during the process of specifying and refining a query. This will be based on novel approaches to representation learning and the extraction of high-level motion descriptors from augmented videos, based on a motion ontology. In addition, IMOTION will develop novel index structures that jointly support traditional video features and the additional motion metadata. [13]

Parts of IMOTION have been released within the vitrivr project as open source projects

2. Vitrivr

Vitrivr is a modern, open-source video retrieval system for searching in large collections of video using a great variety of query modes, including query-by-sketch, query-by-example and query-by-motion. With the multimodal user interface, prospective users

benefit from being able to naturally interact with the vitrivr system by using spoken commands and also by applying multimodal commands which combine spoken instructions with manual pointing. While the main strength of the UI is the seamless combination of speech-based and sketch-based interaction for multimedia similarity search, the speech modality has shown to be very effective for retrieval on its own. In particular, it helps overcoming accessibility boundaries and offering retrieval functionality for users with disabilities. [4]

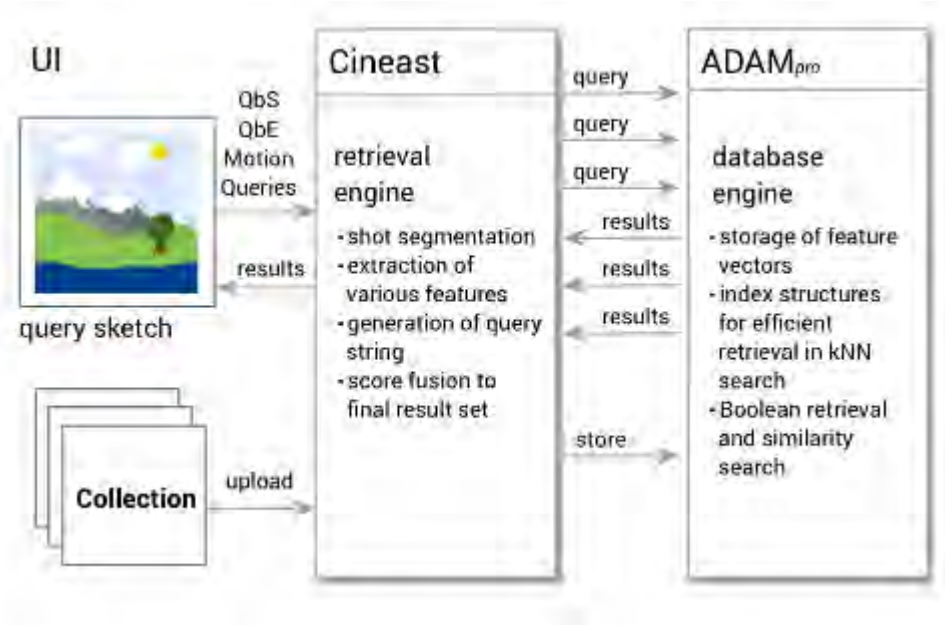


Figure 3.2 vitrivr existing system

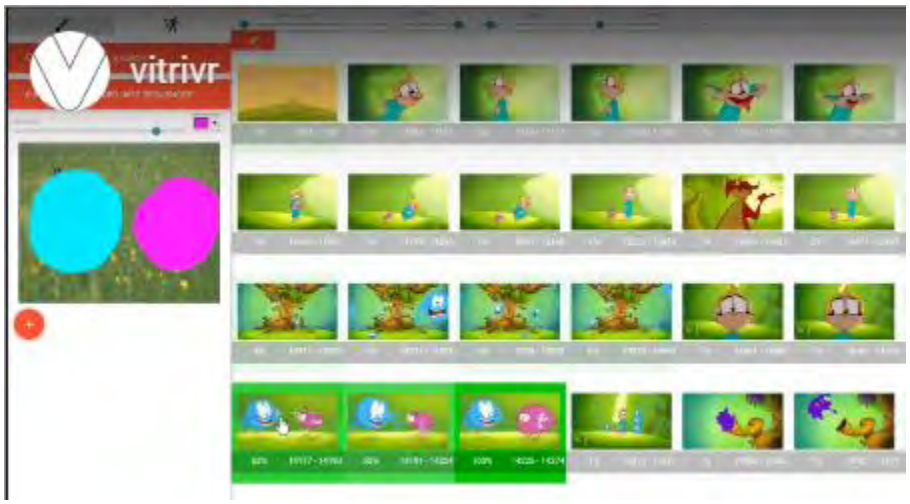


Figure 3.3 Vitvir UI

CHAPTER #4
SOFTWARE DESIGN DOCUMENTATION

4.1 INTRODUCTION

The Software Design Document is a document to provide documentation which will be used to aid in software development by providing the details for how the software should be built. Within the Software Design Document are narrative and graphical documentation of the software design for the project including architecture diagram, sequence diagrams, collaboration models, object behavior models, and other supporting requirement information. It includes the description of how the software will meet the requirements.

4.1.1 Design Overview

The software design document provides design details of Content Based Video Retrieval System. The document contains a complete low-level description of the system, providing insight into the structure and design of each component.

4.2 SYSTEM ARCHITECHTURAL DESIGN

A system architecture is a conceptual model that defines the structure, behavior, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system.

4.2.1 Chosen System Architecture

Chosen system architecture is 3-tier. An Architecture in which user interface, functionality, data storage is developed and maintained as independent modules.

It has three layers:

- Presentation Layer
- Application Layer
- Data Layer

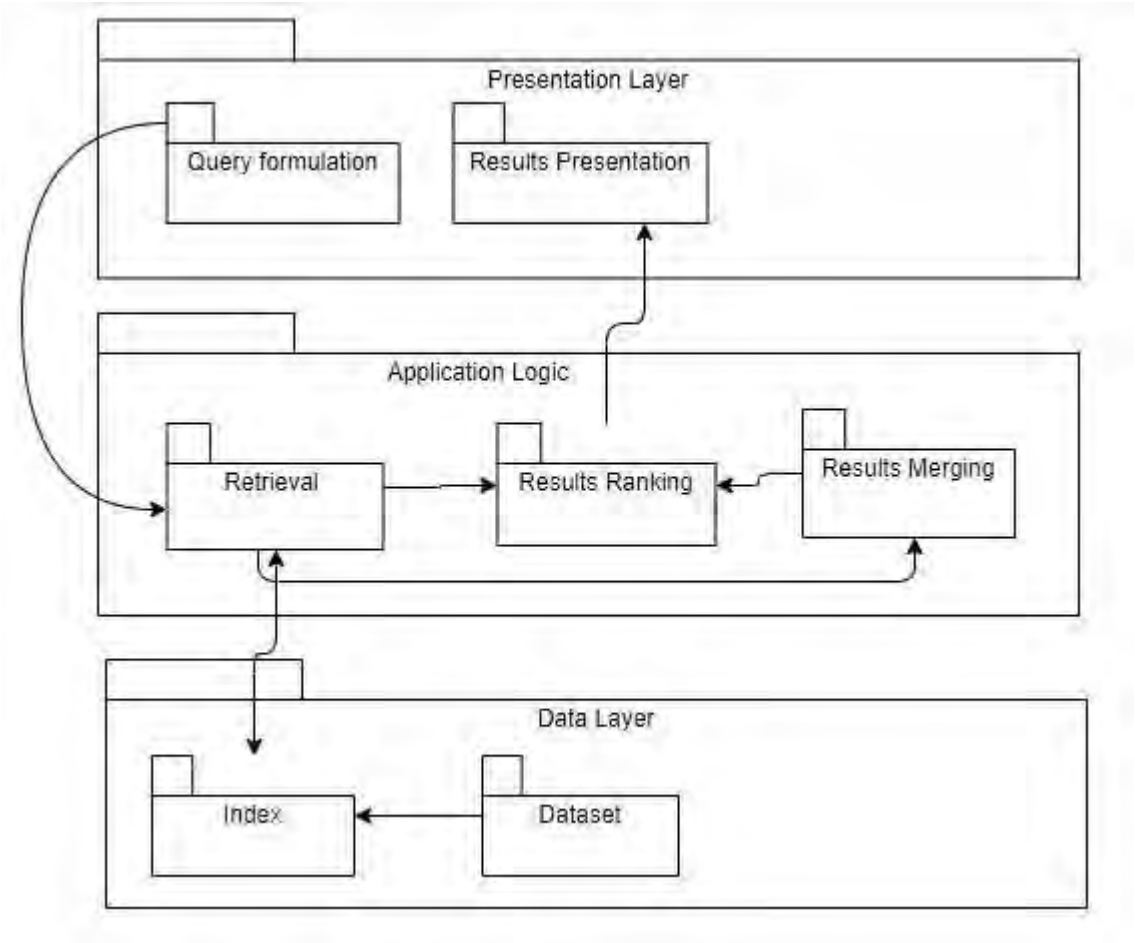


Figure 4.1 Architecture Diagram

4.2.2 System Interface Description

System interface describes the flow of resources. It is the logical characteristics of each interface between the software product and the hardware components of the system. It clearly shows how different entities of tool are interacting with each other.

In CBVRS Interface will be used to query the indexed dataset and view results for entered/uploaded query.

4.3 DETAILED DESCRIPTION OF COMPONENTS

The Component Diagram helps to model the physical aspect of an Object-Oriented software system. It illustrates the architectures of the software components and the dependencies

between them. Those software components including run-time components, executable components also the source code components. Component Diagram for CBVRS:

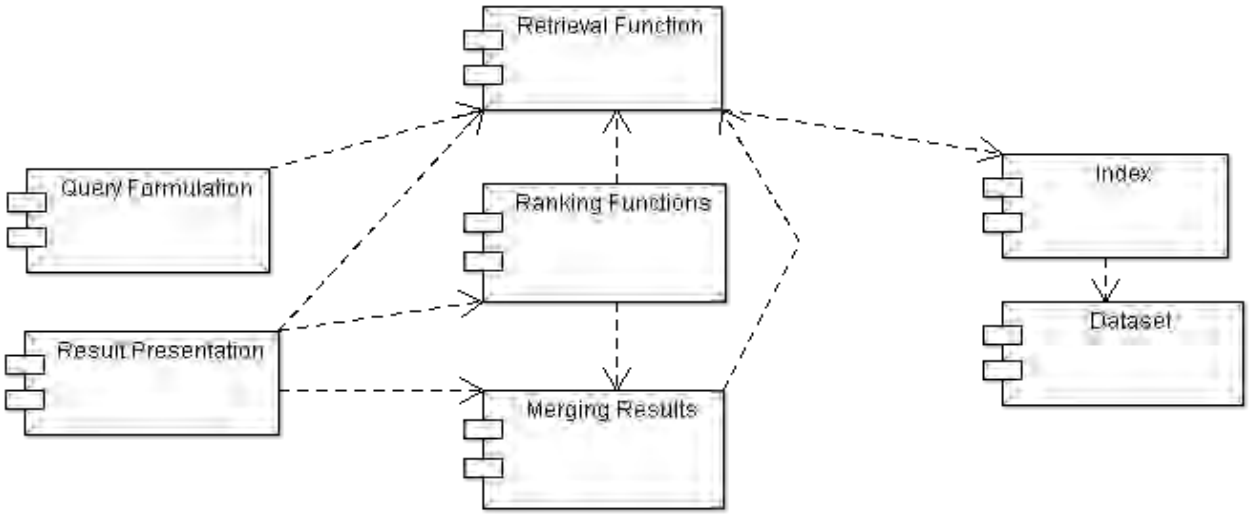


Figure 4.2 Component Diagram

4.4 CLASS DIAGRAM

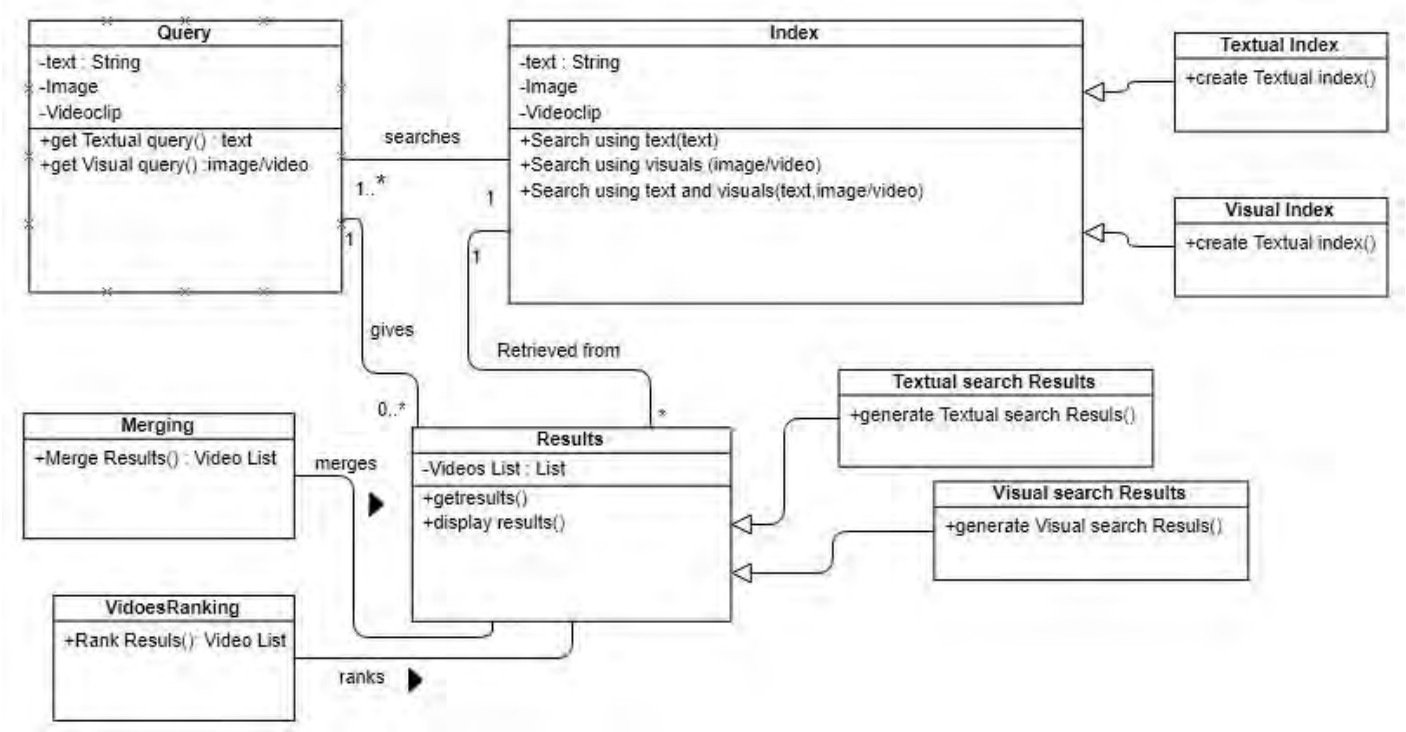


Figure 4.3 Class Diagram

4.5 USER INTERFACE DESIGN

User Interface (UI) Design focuses on anticipating what users might need to do and ensuring that the interface has elements that are easy to access, understand, and use to facilitate those actions.

4.5.1 Description of User Interfaces

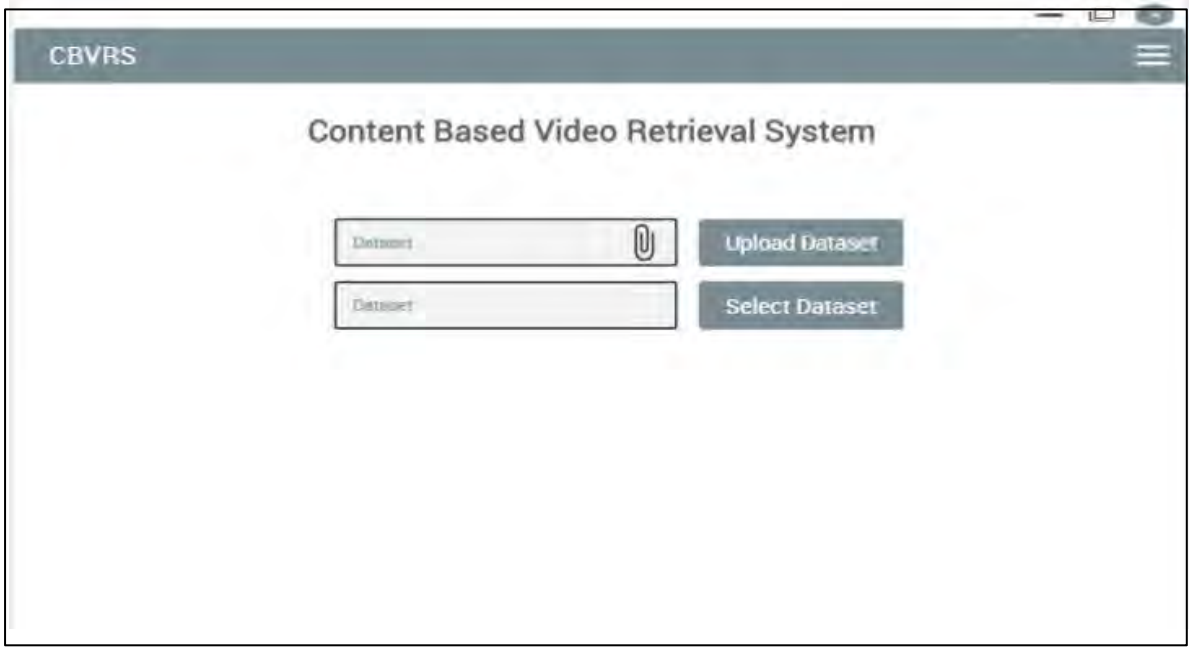
The interface is simple to use, a naïve user can also use it very easily and efficiently to get his required information. The interface is in way so that the user spends most of the time using the application rather than figuring out how to use it.

Home screen displays a menu of functions system can perform. Every screen has menu in left side so user can easily navigate through system.

4.5.2 Screen Images

Following are few screen images of system

Home screen for CBVRS:



After Upload Dataset Screen:

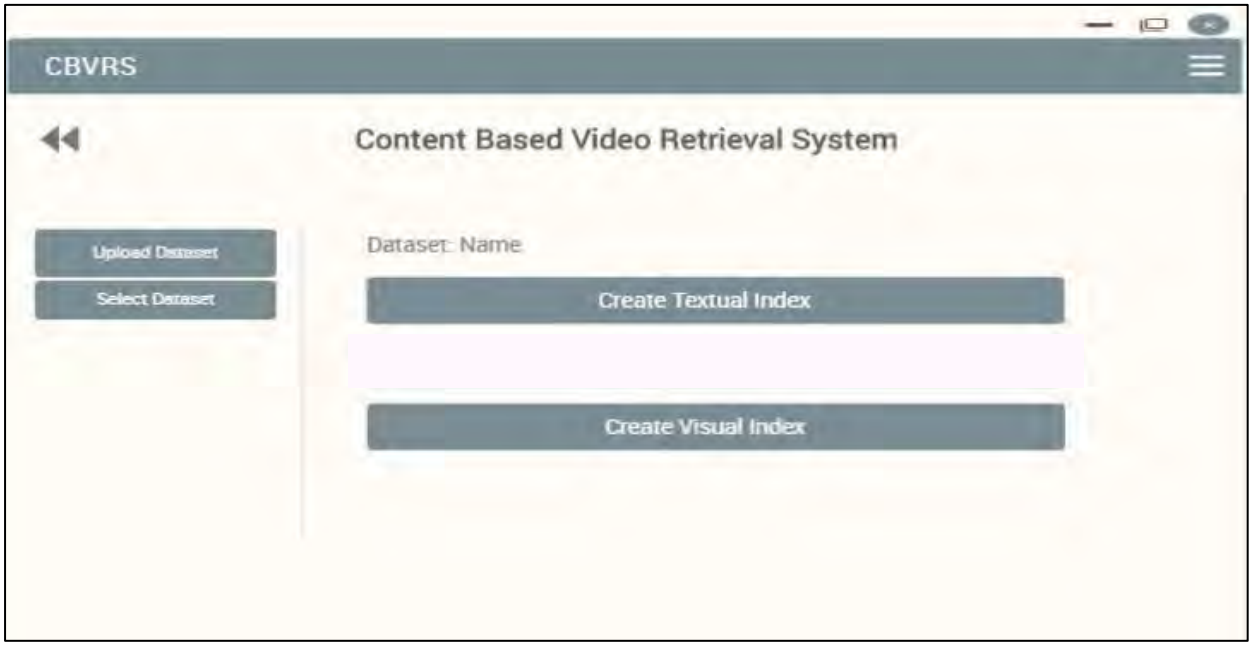


Figure 4.4 CBVRS Settings

Monomodal Search Screen:

Figure 4.5 Monomodal Search

Multimodal Search Screen:

Figure 4.6 Multimodal Search

Monomodal Search Results:

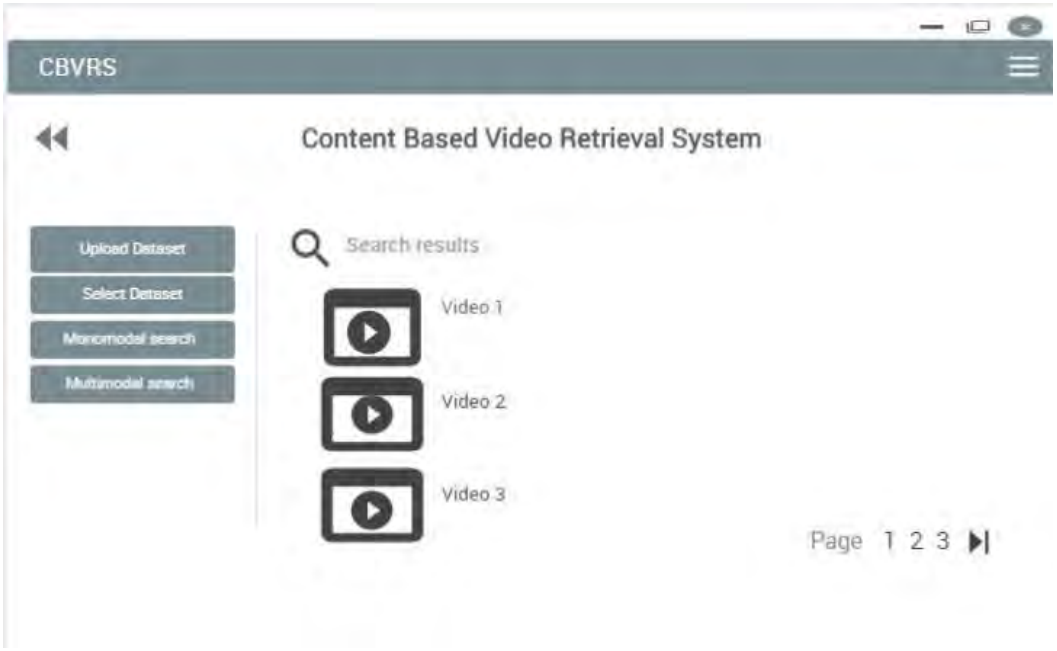


Figure 4.7 Monomodal Result Display

Multimodal Search Results:

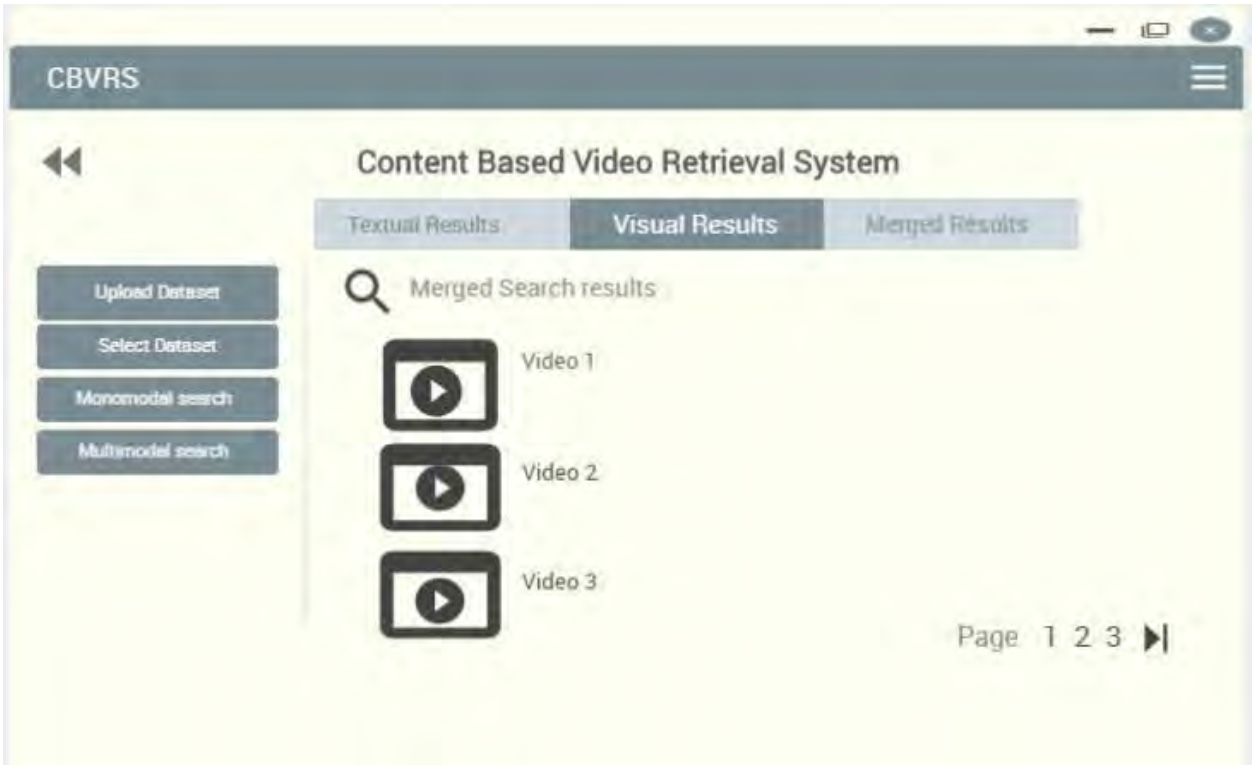


Figure 4.8 Multimodal Results Display

CHAPTER #5
SOFTWARE TEST DOCUMENTATION

5.1 INTRODUCTION

Software test document is a type of document under which tester will determine whether a system under test satisfies requirements or works correctly. The process of developing test cases can also help find problems in the requirements or design of an application

5.1.1 System Overview

Content Based Video Retrieval System is a Desktop Application to retrieve videos from indexed dataset by using monomodal (i.e. search using text, audio or visuals) or multimodal (i.e. combined two or more query options) queries. System merges (in case of multimodal query) and ranks the result and displays them to user.

5.1.2 Test Approach

User Acceptance Testing methodology will be used for testing. It consists of a process of verifying that a solution works for the user. It is not system testing (ensuring software does no crash and meets documented requirements), but rather is there to ensure that the solution will work for the user.

5.2 TEST PLAN

A test plan is a document detailing the objectives, target market and processes for a specific test for a software or hardware product. The plan contains a detailed understanding of the eventual workflow.

5.2.1 Features to be Tested

Features to be tested are following:

- Load dataset
- Select dataset
- Create Index
- Search by text
- Search by visuals(image/video)
- Search by text and visuals
- View Results

5.2.1 Testing Tools and Environment

Tools and environment used for testing are as following:

- Laptop/PC
- Windows Operating System

5.3 TEST CASES

Following are the test cases of CBVRS:

Test Case 1: Upload Dataset

ID	TC-1
DESCRIPTION	Test that dataset is uploaded onto system
SETUP	Open the CBVRS Application
INSTRUCTIONS	1) Open home page. 2) Click on Attachment Button. 3) Browse and select dataset. 4) Click the Upload Dataset button.
INPUTS	Dataset.
EXPECTED RESULT	Dataset uploaded leading to next interface.
ACTUAL RESULT	Dataset uploaded leading to next interface.
VERDICT	Pass.

Table 5.1 TC-1 Upload Dataset

Test Case 2: Select Indexed Dataset

ID	TC-2
DESCRIPTION	Test that dataset is selected from existing indexed dataset in system
SETUP	Open the CBVRS Application
INSTRUCTIONS	1) Open home page. 2) Click on Select dataset. 3) Select from the datasets available to search. 4) Click Ok.

INPUTS	Dataset
	Dataset selected from existing indexed datasets in system
EXPECTED RESULT	Dataset not found. No existing datasets indexed in system.
ACTUAL RESULT	Dataset selected from existing indexed datasets in system
	Dataset not found. No existing datasets indexed in system.
VERDICT	Pass.
	Pass.

Table 5.2 TC-2 ***Select Indexed Dataset***

Test Case 3: Search by text

ID	TC-3
DESCRIPTION	Test that textual query is entered, and results are displayed
SETUP	• Open the CBVRS Application • Dataset selected to search.
INSTRUCTIONS	1) Click on text field. 2) Enter text in text field to search. 3) Click the search button.
INPUTS	text
EXPECTED RESULT	Results displayed to the user
	An error message that "No videos found for entered text"
	Results displayed to the user
ACTUAL RESULT	An error message that "No videos found for entered text"
VERDICT	Pass.

	Pass.

Table 5.3 TC-3 ***Search by text***

Test Case 4: Search by visuals

ID	TC-4
DESCRIPTION	Test that visual query is uploaded, and results are displayed
SETUP	• Open the CBVRS Application • Dataset selected to search.
INSTRUCTIONS	1) Click on Attachment button. 2) Browse and select image/video for search. 3) Click the search button.
INPUTS	Image/video
EXPECTED RESULT	Results displayed to the user
	An error message that "No videos found for uploaded query"
ACTUAL RESULT	Results displayed to the user
	An error message that "No videos found for uploaded query"
VERDICT	Pass.
	Pass.

Table 5.4 TC-4 ***Search by visuals***

Test Case 5: Search by text and visuals

ID	TC-5
DESCRIPTION	Test that text and image/video query is uploaded, and Interface leading to result option displayed.

SETUP	• Open the CBVRS Application • Dataset selected to search.
INSTRUCTIONS	1) Click on text field. 2) Enter text in text field to search. 3) Click on Attachment button. 4) Browse and select image/video for search.
	5) Click the search button.
INPUTS	Text, image, video
EXPECTED RESULT	Interface leading to result option displayed.
	An error message that "No videos found for given query"
ACTUAL RESULT	Interface leading to result option displayed.
	An error message that "No videos found for given query"
VERDICT	Pass.
	Pass.

Table 5.5 TC-5 ***Search by text and visuals***

Test Case 6: View Result

ID	TC-6
DESCRIPTION	Test that multimodal results are displayed if multiple sets.
SETUP	• Open the CBVRS Application • Dataset selected to search. • Multimodal search is performed. • Multiple Result options displayed corresponding to multimodal search.

INSTRUCTIONS	1) Select any Result option (View text search Results, View Visual search results, View audio search results, View merged results) 2) View Displayed results
EXPECTED RESULT	Results displayed for each corresponding option of multimodal search.
	An error message that "No videos found for this query"
ACTUAL RESULT	Results displayed for each option.
	An error message that "No videos found for given query"
VERDICT	Pass.
	Pass.

Table 5.6 TC-6 ***View Result***

CHAPTER #6

SOFTWARE IMPLEMENTATION DOCUMENTATION

6.1 INTRODUCTION

This document describes the project implementation for Content Based Video Retrieval System.

6.1.1 Language Selection:

Project is implemented using

Java:

"Java is a general-purpose programming language that is class-based, object-oriented. It has a huge community support (tens of millions of developers)" [8].

I have used java for this project because I could find open source libraries for both text and image indexing and retrieval in java so there will be no cross-platform issues.

Therefore, in this project java is used for backend development that includes indexing and retrieval of text and images.

Java-Swing:

"Swing is a GUI widget toolkit for Java. It is part of Oracle's Java Foundation Classes (JFC) – an API for providing a graphical user interface (GUI) for Java programs".[9]

Used java-swing for interfaces development because for this project interfaces were to be kept simple and easy to use for searching videos.

MATLAB:

MATLAB is a high-level language and interactive environment for numerical computation, visualization and programming. MATLAB can be used for a range of applications, including signal processing and communications, image and video processing and control systems etc. [10]

For Merging multimodal results.

6.1.2 Tools Selection:

Selected tools is:

- Eclipse for java

6.1.3 Resources:

Dataset:

I-Search Multimodal dataset is being used for this project which contains 51 different classes of objects including 1500 xml files containing textual description of videos.

It contains about 1400 video clips in seconds.

Libraries:

- Lucene Library for Text Indexing:

  Apache Lucene is a search library written in Java. Lucene provides search over documents. Lucene has a highly expressive search API that takes a search query and returns a set of documents ranked by relevancy with documents most similar to the query having the highest score.[11]

  A textual index is created of xml documents containing video descriptions by using Lucene library. Based on its results videos for that query are retrieved from the dataset.

- Lire Library for Image Indexing:

  "LIRE is a Java library that provides a simple way to retrieve images and photos based on color and texture characteristics. LIRE creates a Lucene index of image features for content based image retrieval (CBIR) using local and global state-of-the-art methods. Easy to use methods for searching the index and result browsing are provided. Best of all: it's all open source".[12]

6.2 APPLICATION SCREENSHOTS

Figure 6.1 Home screen for CBVRS

Figure 6.2 Result Screen for textual search results

Figure 6.3 Settings for dataset

CHAPTER #7
CONCLUSION AND FUTURE ENHANCEMENTS

7.1 INTRODUCTION

This document describes the project conclusions and future enhancements for project i.e. what type of new features can be added to it in future.

7.1.1 Conclusions

With this project implemented:

- ✓ We can search video index using textual query.
- ✓ We can view retrieved textual search results.
- ✓ We can search video index using visual query.
- ✓ We can view retrieved visual search results.
- ✓ We can view textual and visual search results.

7.1.2 Future Enhancements:

During period of five months, it is difficult to completely cover the contents of Content Based Video Retrieval System like audio content.

So, in order to cover complete requirements of this new development, many future enhancements can be carried on. Some enhancements can be:

- Accuracy Measurements can be performed to know how accurate this system works.
- Audio Search can also be included as well. User will be allowed to search videos using audio files.

REFERENCES

[1] IEEE Software Engineering Standards Committee, "IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications", June 25, 1998.

[2] B. V. Patel, B. B. Meshram, "Content Based Retrieval Systems", International journal of UbiComp, Vol. 3, No. 2, April 2012, pp.13-30.

[3] ICMR '17 Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval Pages 457-460 "Multimodal Video Retrieval with the 2017 IMOTION System".

[4] https://www.researchgate.net/publication/315858344_Hey_vitrivr_A_Multimodal_UI_for_Video_Retrieval

[5] Lexing Xie, Apostol (Paul) Natsev, Jelena Tesi˘ c´ ∗IBM T. J. Watson Research Center, Hawthorne, NY "Dynamic Multimodal Fusion in Video Search"

[6] "Include relation in usecase" http://www.karonaconsulting.com/downloads/usecases_includesandextends. pdf

[7] Nawei Chen "A survey of Indexing and retrieval of Multimodal Documents: Text and Images"

[7B] https://en.wikipedia.org/wiki/Boolean_model_of_information_retrieval

[8] https://en.wikipedia.org/wiki/Java_(programming_language)

[9] https://en.wikipedia.org/wiki/Swing_(Java)

[10] https://cimss.ssec.wisc.edu/wxwise/class/aos340/spr00/whatismatlab.htm

[11] https://lingpipe-blog.com/2014/03/08/lucene-4-essentials-for-text-search- and-indexing/

[12] http://www.lire-projectnet/

[13] IMOTION - Intelligent Multi-Modal Augmented Video Motion Retrieval System, https://dbis.dmi.unibas.ch/research/projects/imotion/