# I-CUBOID: An Online Tool to Explore Web Image Search Results


**Supervised By: Dr. Umer Rashid**

**Submitted By: Sadia Manzoor**

**Department of Computer Science**

**Quaid-i-Azam University Islamabad**

**Session (2015-2019)**

# ACKNOWLEDGEMENT

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. Countless salutations upon the Holy Prophet Hazrat Muhammad (S.A.W), source of knowledge for enlightening with the essence of faith in Allah and guiding the mankind, the true path of life.

First and foremost, I would like to thank everyone who had contributed to the successful completion of this project. I would like to express my gratitude to my Project Supervisor, Dr. Umer Rashid for his invaluable advice and guidance. I am sincerely grateful to him for sharing his truthful and illuminating views on a number of issues related to the project.

In addition, I would also like to express my gratitude to my family and friends who had helped and given me encouragement in my study.

# ABSTRACT

Search engines like Google, Bing, Flicker, etc., usually display image search results via 2D grid presentations which allow the user horizontal and vertical scanning of image results but not in-depth browsing. When thousands of images are retrieved, then horizontal and vertical scanning process can be tedious. I-Cuboid, an online web based tool provides an interactive exploration of image search results by arranging and displaying them in a cubic presentation. Cubic presentation allows horizontal, vertical and in-depth browsing of image results. I-Cuboid tool uses two image ranking methodologies, one is Flicker's own "sort as relevance" attribute and the other is by using meta-data along with images in Jaccard Index formula. Usability evaluation of I-Cuboid tool is done with eighteen real users and SUS and QUIS questionnaires are used as intruments.

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# LIST OF ABBREVIATIONS

| | |
|---|---|
| **UC** | Use Case |
| **UCD** | Use Case Description |
| **I-Cuboid** | Image Cuboid, proposed name of this system |
| **SPMP** | System Project Management Plan |
| **SRS** | Software Requirements Specifications |
| **SDD** | Software Design Description |
| **API** | Application Program Interface |
| **HTTP** | Hyper Text Transfer Protocol |
| **WWW** | World Wide Web |
| **TCP/IP** | Transmission Control Protocol/Internet Protocol |
| **User** | Someone who interacts with system |
| **ASQ** | After-Scenario Questionnaire |
| **CSUQ** | Computer System Usability Questionnaire |
| **QUIS** | Questionnaire for User Interface Satisfaction |

# CHAPTER # 01

# LITERATURE REVIEW

# IMPORTANT TERMS

## Keyword

Keywords are the words and phrases in the web content that make it possible for people to find web site via search engines.

## Search Engine

Search engine is a service that allows Internet users to search for content via the World Wide Web (WWW).

## Search Query

A web search query is a query based on a specific search term that a user enters into a web search engine to satisfy his or her information needs.

## Text Based Search Engine

A search engine that provides text-based searching on the World Wide Web is called Text-Based Search Engine.

## Image Retrieval

Image retrieval is the task of fetching relevant images against a query from a large scale image data or database.

## Ranking

A ranking is a relationship between a set of items such that, for any two items, the first is either 'ranked higher than', 'ranked lower than' or 'ranked equal to' the second.

## Image Ranking

Ranking of images, retrieved from the online image source.

## API

API stands for Application Programming Interface, a publicly available API that returns data, likely in JSON or XML. It just allows applications to communicate with one another.

## Meta Data

Meta-data is data or information that provides information about other data.

### Jaccard Index

Jaccard Index is one of the similarity measures, used to find the similarity between finite sample sets.

## EXISTING SYSTEM AND PROPOSED SYSTEM

Search engines usually display image search results in 2-dimensional grid presentation where image exploration is done via horizontal and vertical scanning of 2-dimensional grid. So, when large numbers of image search results are retrieved, horizontal and vertical scanning makes the image exploration process tedious. Therefore, in-depth browsing of image search results in not possible in existing presentation approaches.

Some full text based search engines that display image results in a 2d grid presentation are:

- Google ( www.google.com )
- Yahoo ( www.yahoo.com )
- Bing ( www.bing.com )
- Flicker ( www.flickr.com )
- Picsearch ( www.picsearch.com )
- Getty Images ( www.gettyimages.com )
- Shutter shock ( www.shutterstock.com )

Name the proposed system as “I-Cuboid” which is an online tool allowing image exploration in more interactive way. The tool retrieve image search results from a particular selected image search engine and arrange and display them in a grid and cubic presentation. The cubic presentation, not only allows horizontal and vertical scanning but also in-depth browsing of image search results.

Among the existing search engines that uses 2d grid presentation, the selected search engine is Flicker search engine.

2d grid presentation of Flicker Search Engine is given in Figure 1.1.

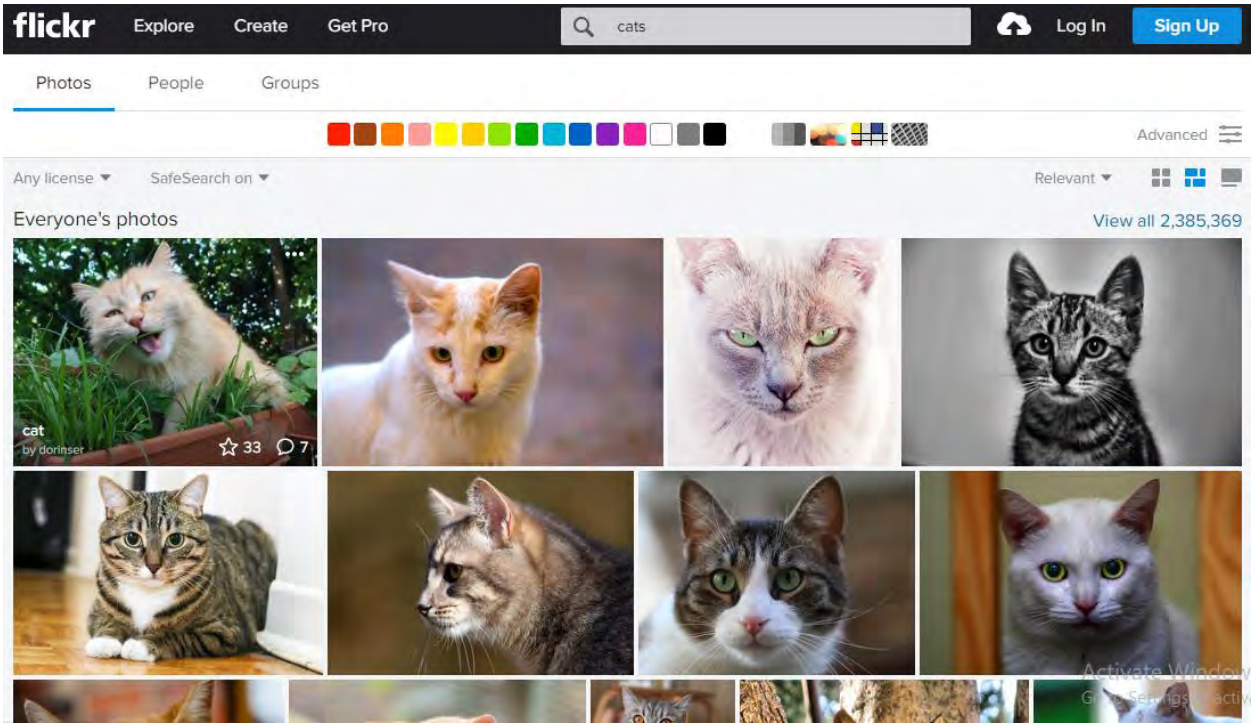



Figure 1.1 Existing Search Engine (Flicker)

Flicker search engine is used to retrieve image search results in the proposed system by using Flicker.photos.search API.

Proposed system will implement 5x5x5 cube, where there are five planes and on each plane we have 5x5 grid and an image is displayed on each small box named as cubie. The proposed system will look like as given in Figure 1.2.

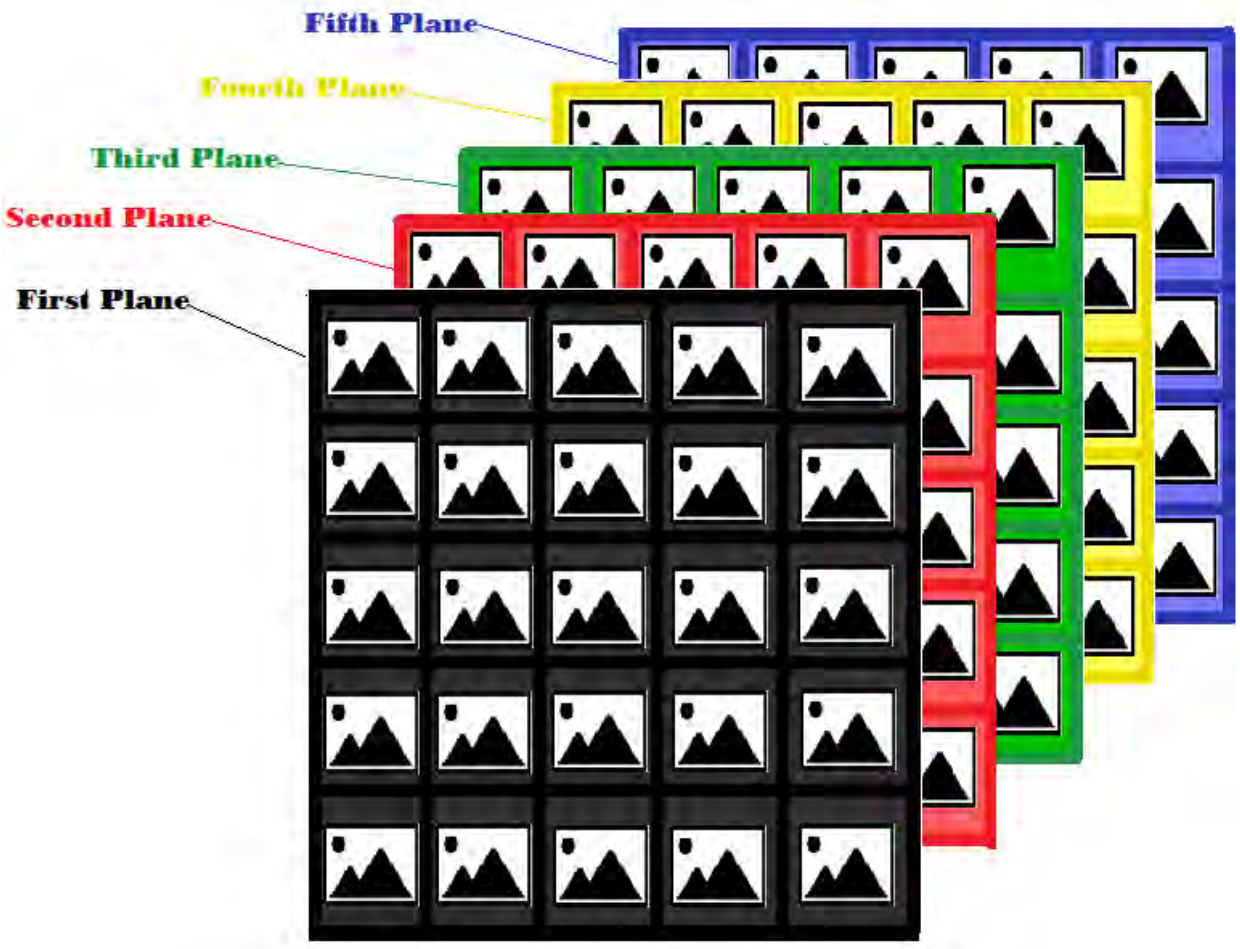



Figure 1.2 Concept of Cubic Presentation

For the sake of comparison, both of the presentations will be implemented i.e.

- Images displayed in Grid Presentation
- Images displayed in Cubic Presentation

As cube has five planes so each plane will have twenty five images and total of one hundred and twenty five images will be displayed on the entire Cubic Structure at a time whereas for grid presentation, same number of images (one hundred and twenty five) will be displayed in grid at one page.

Flicker search engine provides different sorting parameters based on which returned image results can be ranked. The selected sort parameter is “relevance”. This ranking of images is provided by Flicker so named it as Default_Ranking. The default_ranking is revised by using textual meta-data available with images via Jaccard Index. This new ranking based on textual similarity is named as New_Ranking or Revised_Ranking.

For both the presentations grid as well as cube, user can see image results with both of the rankings. By default, images will be displayed in grid or in cube via Default_Ranking. Then an option will be provided to user to toggle the rank mode. When the option is selected, images will be displayed via New_Ranking.

Images ranking against textual query is totally text based i.e. by using the textual meta-data available, as returned by flicker API along with images and Jaccard Index is used for this purpose.

After images are ranked and get arranged in cube, then In-depth browsing schemes are applied.

## JACCARD INDEX

Jaccard index, also known as Jaccard similarity coefficient measures similarity between finite sample sets and is defined as :

"Size of intersection divided by the size of union of the sample sets."

Mathematically, it is represented as:

Suppose

A = Set1

B = Set2

Then Jaccard Index of Set1 and Set2 is:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}.$$

The value of Jaccard Index lies from zero to one. Zero means there is no similarity between the two sets and one means the two sets are completely similar.

$$0 \leq J(A,B) \leq 1.$$

## IMAGES RANKING

The proposed system is text-based system i.e. it accepts textual queries and also, compute query-image similarity based on textual meta-data available with images. So images are ranked based on their textual similarities by using the Jaccard Index values in a decreasing order.

Among all the meta-data returned by the Flicker along with images, there are just three kinds of textual data by using one of them, ranking can be done. These are:

- Image Title
- Image Description
- Tags

The Json Format returned by Flicker API for images against a query is shown below in Figure 1.3 as an example where for one image, title, description and tags are marked with orange lines and for another image, title, description and tags are marked with green lines:


```
{"id":"49344051212","owner":"96657002@N06","secret":"0810a9a5c1","server":"65535","farm":66,
"title":"Blyde River Canyon",
"ispublic":1,"isfriend":0,"isfamily":0,
"license":"0",
"description":{"_content":"Mpumalanga, South Africa"},
"o_width":"5184","o_height":"3456","dateupload":"1578385315","lastupdate":"1578392185",
"datetaken":"2019-07-02 13:11:08","datetakengranularity":"0","datetakenunknown":"0","ownername":"Rckr88","iconserver":"7438","iconfarm":8,"views":"99",
"tags":"blyde river canyon blyderivercanyon mpumalanga southafrica south africa rivers water mountains mountain cliffs cliff rocks rock greenery green grass travel
travelling trees tree nature naturalworld outdoors",
"machine_tags":"","originalsecret":"05d000b585","originalformat":"jpg","latitude":0,"longitude":0,"accuracy":0,"context":0,"media":"photo","media_status":"ready",
"url_sq":"https:\/\/live.staticflickr.com\/65535\/49344051212_0810a9a5c1_s.jpg","height_sq":75,"width_sq":75,
"url_t":"https:\/\/live.staticflickr.com\/65535\/49344051212_0810a9a5c1_t.jpg","height_t":67,"width_t":100,
"url_s":"https:\/\/live.staticflickr.com\/65535\/49344051212_0810a9a5c1_m.jpg","height_s":160,"width_s":240,
"url_q":"https:\/\/live.staticflickr.com\/65535\/49344051212_0810a9a5c1_q.jpg","height_q":150,"width_q":150,
"url_m":"https:\/\/live.staticflickr.com\/65535\/49344051212_0810a9a5c1.jpg","height_m":333,"width_m":500,
"url_n":"https:\/\/live.staticflickr.com\/65535\/49344051212_0810a9a5c1_n.jpg","height_n":213,"width_n":320,
"url_z":"https:\/\/live.staticflickr.com\/65535\/49344051212_0810a9a5c1_z.jpg","height_z":427,"width_z":640,
"url_c":"https:\/\/live.staticflickr.com\/65535\/49344051212_0810a9a5c1_c.jpg","height_c":533,"width_c":800,
"url_l":"https:\/\/live.staticflickr.com\/65535\/49344051212_0810a9a5c1_b.jpg","height_l":683,"width_l":1024,
"url_o":"https:\/\/live.staticflickr.com\/65535\/49344051212_05d000b585_o.jpg","height_o":3456,"width_o":5184,"pathalias":"rckr88"},

{"id":"49343833301","owner":"22695788@N05","secret":"6dc5129792","server":"65535","farm":66,
"title":"LAGO YAMDROK",
"ispublic":1,"isfriend":0,"isfamily":0,
"license":"0",
"description":{"_content":"\u00a9Todos los derechos reservados."},
"dateupload":"1578385222","lastupdate":"1578391310","datetaken":"2019-09-28 06:10:36",
"datetakengranularity":"0","datetakenunknown":"0","ownername":"RLuna (Instagram @rluna1982)","iconserver":"2853","iconfarm":3,"views":"92",
"tags":"shigatse gyantse smila lake mania tibet mountain nature asia canon viaje landascape travel holidays vacaciones rluna rluna1982 unesco trip ecologia spotlight
instagramapp photography natural lhasa china lasa yak yamdrok",
"machine_tags":"","latitude":"29.132970","longitude":"90.483398","accuracy":"7","context":0,
"place_id":"","woeid":"10273968","geo_is_family":0,"geo_is_friend":0,"geo_is_contact":0,"geo_is_public":1,"media":"photo","media_status":"ready",
"url_sq":"https:\/\/live.staticflickr.com\/65535\/49343833301_6dc5129792_s.jpg","height_sq":75,"width_sq":75,
"url_t":"https:\/\/live.staticflickr.com\/65535\/49343833301_6dc5129792_t.jpg","height_t":67,"width_t":100,
"url_s":"https:\/\/live.staticflickr.com\/65535\/49343833301_6dc5129792_m.jpg","height_s":160,"width_s":240,
"url_q":"https:\/\/live.staticflickr.com\/65535\/49343833301_6dc5129792_q.jpg","height_q":150,"width_q":150,
"url_m":"https:\/\/live.staticflickr.com\/65535\/49343833301_6dc5129792.jpg","height_m":333,"width_m":500,
```




Figure 1.3 Json Format for Images

- Description is not available with all the images. It is NULL for a large number of images.
- Title is set by image owner, so in many cases title is not relevant to the query.

A list of tags is defined by Flicker itself for each image. So the selected textual-data upon which images ranking can be done against textual query is Tags.

Now suppose

Query = Q = “horse donkey zebra”

Image1 Tags = “horse landscape sky donkey zebra clouds”

Image2 Tags = “zebra horse”

Image3 Tags = “donkey sky forest horse”

Image4 Tags = “horse trees zebra donkey”

Image 5 Tags = “zebra”

Then Jaccard indexes of images against query are

J( Q, Image1) = 0.5, J( Q, Image2) = 0.67, J( Q, Image3) = 0.4, J( Q, Image4) = 0.75,

J( Q, Image5) = 0.33

The images are ranked in descending order of their Jaccard indexes against query. The higher the value of Jaccard index, the more similar the image is to the query.

Ranked order of images is:

Image4

Image2

Image1

Image3

Image5

# IN-DEPTH BROWSING

Usually users do browsing by horizontal and vertical scanning of image results arranged in grid. For the purpose of in-depth browsing, images are arranged in cubic presentation. Cube is a structure which has five planes behind one another as shown in Figure 1.2. So the idea of in-depth browsing in cube means that user is browsing images horizontally and vertically of planes as well as behind the planes i.e. user can see the images of plane 1, then user navigate to the plane 2 and so on, going in-depth.

In-depth browsing is enabled by the following two schemes:

## First Scheme

After images are arranged in the entire cube, options are provided to select each plane. So there are total five options available for each plane to be selected. These are Show_Plane1, Show_Plane2, Show_Plane3, Show_Plane4 and Show_Plane5. Each of these options allow user

to see all of the images arranged in that plane, which were otherwise hidden behind that plane which is forward to it.

In default setting of cube as shown in Figure 1.4, first plane is selected by default and all the remaining four planes are behind the first plane. So a user can see all of the images of first plane but cannot see all of the images of planes 2, 3, 4 and 5.

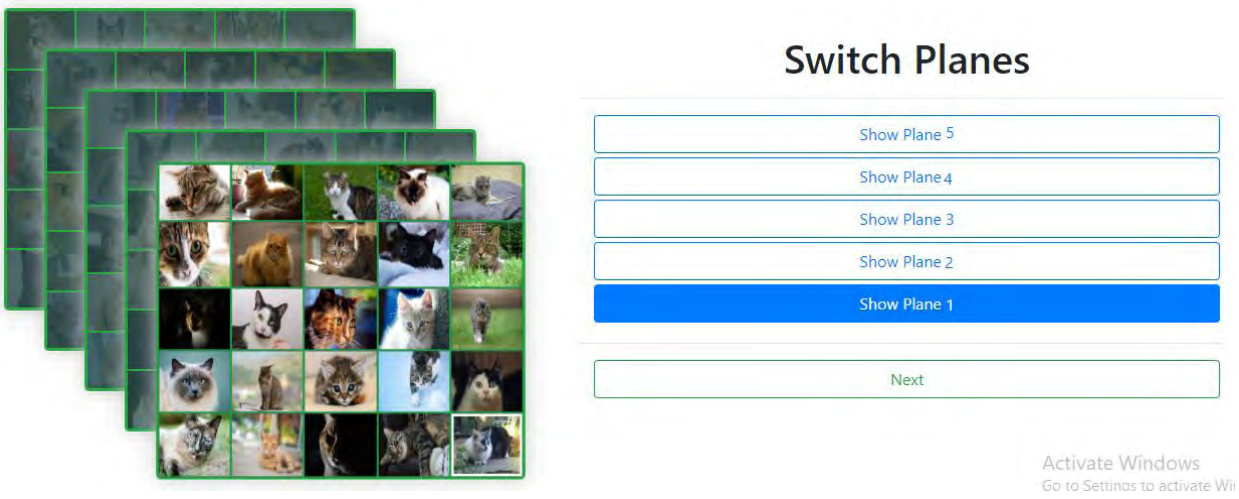



Figure 1.4 Default Setting of Cubic Presentation

So in order to view all of the images of a plane which is behind some other plane, user can select that plane. In this way, user can navigate plane to plane, browsing images in-depth.

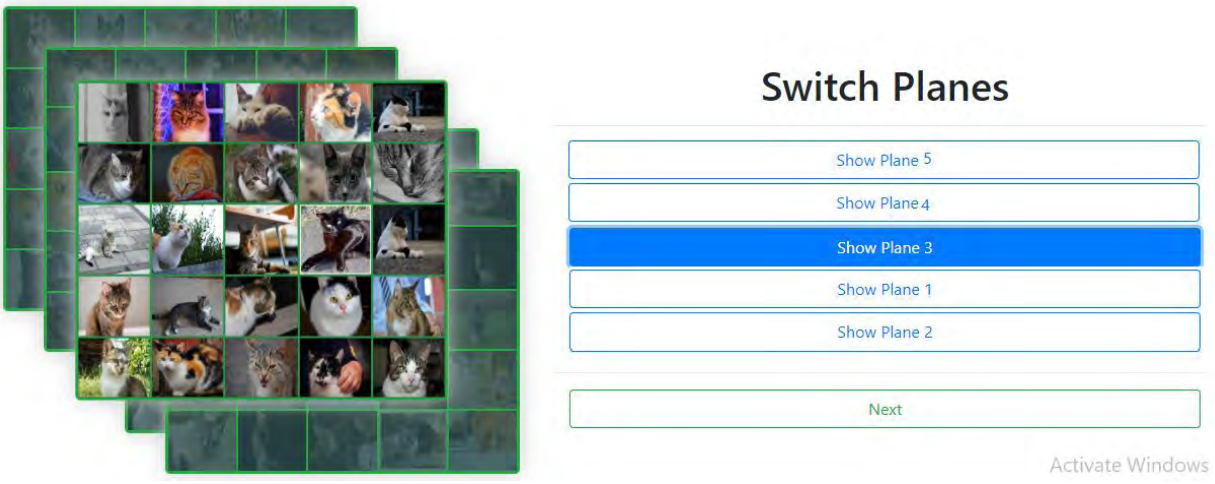
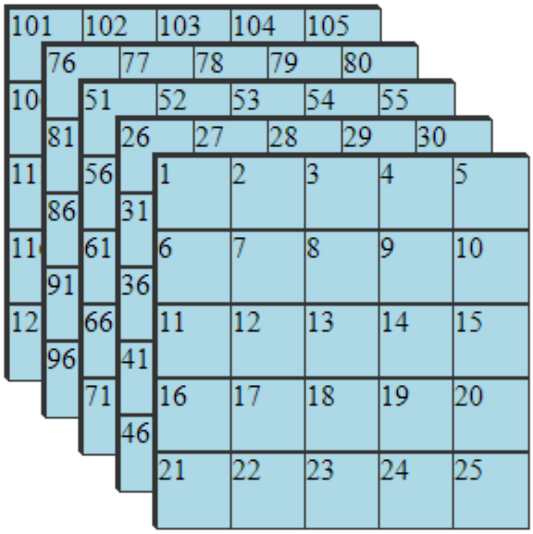



Figure 1.5 Browsing Images of Plane 3

## Second Scheme

After images are arranged in the entire cube and user clicks on an image in a plane, then that image will be opened along with its neighbor images. These neighbor images can be in the same plane, in the forward plane and in the backward plane. Minimum number of neighbor images an image can have is three and maximum number is six.

For simplicity, consider the following cube where numbering is defined to mark each cell and suppose images are arranged in it.




**Case 1: Three Neighbor Images of Planes 1 and 5**

Images that are at corners of Plane_1 and Plane_5 i.e. at positions 1, 5, 21, 25 of first plane and 101, 105, 121, 125 of fifth plane have three neighbor images.

Neighbor Images of image_at_position_1 are images at positions 2(right neighbor), 6(down neighbor) and 26(back neighbor). Its left, up and front neighbors are NULL. Same goes for image_at_position_5 except that it has left neighbor (at position 4) and its right neighbor is NULL.

Neighbor Images of image_at_position_21 are images at positions 16rup neighbor), 22(right neighbor) and 46(back neighbor). Its left, down and front neighbors are NULL. Same goes for image_at_position_25 except that it has left neighbor (at position 24) and its right neighbor is NULL.

Neighbor Images of image_at_position_101 are images at positions 102(right neighbor), 106(down neighbor) and 76(front neighbor). Its left, up and back neighbors are NULL. Same

goes for image_at_position_105 except that it has left neighbor (at position 104) and its right neighbor is NULL.

Neighbor Images of image_at_position_121 are images at positions 116(up neighbor), 122(right neighbor) and 96(front neighbor). Its left, down and back neighbors are NULL. Same goes for image_at_position_125 except that it has left neighbor (at position 124) and its right neighbor is NULL.

**Case 2: Four Neighbor Images of Planes 1 and 5**

For Plane_1 and Plane_5, images that are in Row1, Row5, Col1 and Col5, excluding the corner images ( at positions 1, 5, 21, 25, 101, 105, 120, 125 ), have four neighbor images.

Neighbor Images of image_at_position_2 are images at positions 1(left neighbor), 3(right neighbor), 7(down neighbor) and 27( back neighbor). Its up and front neighbors are NULL. Similarly for image_at_position_3 and image_at_position_4. Same goes for image_at_position_22 except that it has up neighbor (at position 17) and its down neighbor is NULL. Similarly for image_at_position_23 and image_at_position_24.

Neighbor Images of image_at_position_6 are images at positions 1(up neighbor), 11(down neighbor), 7(right neighbor) and 31( back neighbor). Its left and front neighbors are NULL. Similarly for image_at_position_11 and image_at_position_16. Same goes for image_at_position_10 except that it has left neighbor (at position 9) and its right neighbor is NULL. Similarly for image_at_position_15 and image_at_position_20.

Same is the case with images at 102, 103, 104, 122, 123, 124, 106, 111, 116, 110, 115 and 120 with the difference is, they have front neighbor and their back neighbor in NULL.

**Case 3: Five Neighbor Images of Planes 1 and 5**

All the remaining images of Plane_1 and Plane_2 have five neighbor images.

For Plane_1 images, neighbors are: [up, down, left, right, back]

For Plane_5 images, neighbors are: [up, down, left, right, front]

**Case 4: Four Neighbor Images of Planes 2, 3 and 4**

Images that are at corners of plane 2, 3 and 4, at positions 26, 30, 46, 50 of second plane, 51, 55, 71, 75 of third plane and 76, 80, 96, 100 of third plane have four neighbor images.

For images at positions 26, 51, 76, neighbors are: [right, down, front, back]

For images at positions 30, 55, 80, neighbors are: [left, down, front, back]

For images at positions 46, 71, 96, neighbors are: [right, up, front, back]

For images at positions 50, 75, 100, neighbors are: [left, up, front, back]

**Case 5: Five Neighbor Images of Planes 2, 3 and 4**

For images that are in Row1, Row5, Col1 and Col5, excluding corner images, have five neighbor images.

**Case 6: Six Neighbor Images of Planes 2, 3 and 4**

All the remaining images in planes 2, 3 and 4 have six neighbor images [ up, down, left, right, front, back ].

# CHAPTER # 02

# SOFTWARE PROJECT MANAGEMENT PLAN

# 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. Software Project Management Plan is the document for managing a software project; it describes the software approach, milestones and other necessary details to develop software work products that satisfy the product requirements.

## Project Overview

I-Cuboid is an online web based tool designed for exploration of image search results in an interactive way. The tool takes textual query as input from user, forward it to Flicker search engine that provides images along with meta-data, retrieve the image search results from Flicker search engine and rank them. The tool then, arranges the images in a cubic presentation and in grid presentation. Cubic presentation enables the horizontal, vertical and in-depth browsing of the search results interactively.

## 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 I-Cuboid are as follows:

Table 2. 1 Deliverables

| **Document Deliverables** | **Description** | **Phase** |
|---|---|---|
| SPMP,<br><br>SRS | Description of the software approach and associated milestones.<br><br>Description of expected software features, constraints, interfaces and other attributes | 1st Phase |
| SDD, | Description of how the software will meet the requirements. Also describes | |

| | | |
|---|---|---|
| STD | the rationale of design decisions taken.<br><br>Description of the plan and specifications to verify and validate the software and the results. | 2nd Phase |
| Implementation | Implementation of the Project | 3rd Phase |

# PROJECT ORGANIZATION

Project Organization contains description of software process model used for the project, roles and responsibilities of the people involved in making the project and tools and techniques to be used in this project.

## Software Process Model

In this project, Prototype model will be used because:

- Prototype provides an early overview of the product.
- Prototype evolves with time so newer functionality can be added.
- Client can get the idea of finished product from prototype and can recommend further change in requirements.

## Roles and Responsibilities

I am single developer of this project so there is no division of roles and responsibilities.

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

## Tools and Techniques

The following table shows the tools and techniques to be used in the project:

Table 2. 2 Tools and Techniques

| Name | Version | Description |
|---|---|---|
| MS Word | 2010 | Used for documentation purposes |
| Project Libre | 1.7 | Used to make project plan. |
| Draw.io | Online Tool | Used for making system diagrams. |
| Visual Studio Code | 1.39.2 | A code editor redefined and optimized for building and debugging modern web and cloud applications. |
| Html and CSS | Html v5 | HTML is the standard markup language for creating Web pages.<br>CSS is a style sheet language used for describing the presentation of a document written in a markup language like HTML. |
| Bootstrap | v4 | Bootstrap4 is the newest version of Bootstrap,<br>Bootstrap is an open source Toolkit for developing responsive websites with HTML, CSS and JavaScript. |
| JavaScript | - | An object-oriented computer programming language commonly used to create interactive effects within web browsers. |

| | | |
|---|---|---|
| | Vue.js v2.x | For java script, Vue.js framework of java script is used. |
| Php | -<br><br><br><br><br>Laravel v6 | HyperText Preprocessor<br>PHP is an HTML-embeded, server side scripting language designed for web development.<br>For Php, Laravel framework of php is used. |
| Flicker.photos.search API | - | The API returns a list of relevant images from a search query which includes thumbnails, full image URLs, publishing website information, and image metadata. |
| Jaccard Index | - | Jaccard Index is used to rank images. |

# PROJECT MANAGEMENT PLAN

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

- ***Requirements Analysis Phase:***

➢ **Task 1: Identify Requirements**

- **Task Description**
  The initial step in the development of the project is the identification of requirements. The requirements include functional and non-functional requirements.
- **Deliverables and Milestones**
  Requirements are collected and reviewed.

- **Resources Needed**
  **People:** Sadia Manzoor, Project Supervisor
  **Software:** MS Word
  **Hardware**: Laptop
- **Dependencies and Constraints**
  None
- **Risk and Contingencies**
  None

- **Task 2: Define Use cases**
  - **Task Description**
    This task includes defining and writing use cases and making use case diagram.
  - **Deliverables and Milestones**
    Its milestone is use case description including use case diagram.
    Use cases are written down and reviewed.
  - **Resources Needed**
    **People:** Sadia Manzoor, Project Supervisor
    **Software:** MS Word
    **Hardware**: Laptop
  - **Dependencies and Constraints**
    None
  - **Risk and Contingencies**
    None

- **Task 3: Develop Analysis Model**
  - **Task Description**
    This task includes making domain model for the system.
  - **Deliverables and Milestones**
    Its deliverable is Domain model of the system.
  - **Resources Needed**
    **People:** Sadia Manzoor, Project Supervisor
    **Software:** Draw.io
    **Hardware**: Laptop
  - **Dependencies and Constraints**
    None
  - **Risk and Contingencies**
    None

- **Task 4: Develop SRS**
  - **Task Description**
    This task includes making Software Requirements Specification document which contains the description of functional and non-functional requirements.
  - **Deliverables and Milestones**
    Its deliverable is SRS Document
    Its milestone is completion of Requirement Analysis phase.
  - **Resources Needed**
    **People:** Sadia Manzoor, Project Supervisor
    **Software:** MS Word
    **Hardware**: Laptop
  - **Dependencies and Constraints**
    None
  - **Risk and Contingencies**
    None

- ***Design Phase:***

- **Task 1: Develop Design**
  - **Task Description**
    This task includes the development of architectural design and detailed design of the system.
  - **Deliverables and Milestones**
    Deliverables are Architectural and detailed designs.
  - **Resources Needed**
    **People:** Sadia Manzoor, Project Supervisor
    **Software:** MS Word, Draw.io
    **Hardware**: Laptop
  - **Dependencies and Constraints**
    The development of designs requires the completion of requirements analysis phase.
  - **Risk and Contingencies**
    None

- **Task 2: Evaluate Design**
  - **Task Description**
    This task includes evaluation and verification of the design.
  - **Deliverables and Milestones**
    Milestone is completion of Design phase.
  - **Resources Needed**

**People:** Sadia Manzoor, Project Supervisor
**Software:** MS Word
**Hardware:** Laptop

- **Dependencies and Constraints**
  Architectural Design and Detailed Design of the system should be complete.
- **Risk and Contingencies**
  None

➢ **Task 3: Develop Software Test Documentation**

- **Task Description**
  This task includes defining test cases for the system.
- **Deliverables and Milestones**
  Its deliverable is Software Test Document.
  Test cases are reviewed.
- **Resources Needed**
  **People:** Sadia Manzoor, Project Supervisor
  **Software:** MS Word
  **Hardware:** Laptop
- **Dependencies and Constraints**
  To define the test cases, design phase should be complete.
- **Risk and Contingencies**
  None

- ***Implementation Phase***

➢ **Task 1: System Implementation**

- **Task Description**
  This phase includes the development of application.
- **Deliverables and Milestones**
  Implementation of the application is complete.
- **Resources Needed**
  **People:** Sadia Manzoor, Project Supervisor
  **Software:** Visual Studio Code Editor, Flicker.photos.search API, Web Browser
  **Hardware:** Laptop
- **Dependencies and Constraints**
  The development of application requires the completion of design phase.
- **Risk and Contingencies**
  None

- ***Integration Phase***

➢ **Task 1: System Integration**

- **Task Description**
  This phase includes the integration of all the software modules.
- **Deliverables and Milestones**
  Application is developed completely.
- **Resources Needed**
  **People:** Sadia Manzoor, Project Supervisor
  **Software:** Visual Studio Code Editor, Flicker.photos.search API, Web Browser
  **Hardware:** Laptop
- **Dependencies and Constraints**
  Implementation phase should be completed.
- **Risk and Contingencies**
  None

- ***Testing Phase***

➢ **Task 1: System Testing**

- **Task Description**
  This phase includes the testing of application based on different testing parameters.
- **Deliverables and Milestones**
  Application is tested.
- **Resources Needed**
  **People:** Sadia Manzoor, Project Supervisor, Sample User
  **Software:** Flicker.photos.search API, Web Browser
  **Hardware:** Laptop
- **Dependencies and Constraints**
  For the testing of application integration phase should be complete.
- **Risk and Contingencies**
  None

## Timetable and Gantt Chart

This section describes the timetable of the Project.

| Name | Duration | Start | Finish | Predecessors | Resource Names |
|---|---|---|---|---|---|
| **Exploration of Web Image Search Results Tool** | 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** | **Sadia Manzoor;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** | **Sadia Manzoor;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** | **Sadia Manzoor;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 |

| Name | Duration | Start | Finish | Predecessors | Resource Names |
|---|---|---|---|---|---|
| Make Domain Model | 2 days | 12/18/18 8:00 AM | 12/19/18 5:00 PM | 26 | Argo Uml |
| Review and Refine SRS | 2 days | 12/20/18 8:00 AM | 12/21/18 5:00 PM | 28 | Dr Umar Rasheed |
| **Make Software Design Description Document** | 16 days | **12/24/18 8:00 AM** | **1/14/19 5:00 PM** | **29** | **Sadia Manzoor;PC;MS W...** |
| Give Introduction and Overview | 1 day | 12/24/18 8:00 AM | 12/24/18 5:00 PM | | |
| Make Activity Diagrams | 4 days | 12/24/18 8:00 AM | 12/27/18 5:00 PM | | Argo Uml |
| Review and Refine Activity Diagram | 2 days | 12/28/18 8:00 AM | 12/31/18 5:00 PM | 32 | Dr Umar Rasheed |
| Make System Architectural Design | 2 days | 12/24/18 8:00 AM | 12/25/18 5:00 PM | | Argo Uml |
| Review and Refine Architecture Diagram | 2 days | 12/26/18 8:00 AM | 12/27/18 5:00 PM | 34 | Dr Umar Rasheed |
| Make Sequence Diagrams | 2 days | 12/28/18 8:00 AM | 12/31/18 5:00 PM | 34;35 | Argo Uml |
| Review and Refine SD | 2 days | 1/1/19 8:00 AM | 1/2/19 5:00 PM | 36 | Dr Umar Rasheed |
| Identify Classes | 2 days | 1/3/19 8:00 AM | 1/4/19 5:00 PM | 37 | |
| Make Class Diagram | 2 days | 1/7/19 8:00 AM | 1/8/19 5:00 PM | 38 | Argo Uml |
| Review and Refine Class Diagram | 2 days | 1/9/19 8:00 AM | 1/10/19 5:00 PM | 39 | Dr Umar Rasheed |
| Review and Refine Software Design Description | 2 days | 1/11/19 8:00 AM | 1/14/19 5:00 PM | 40 | Dr Umar Rasheed |
| **Make User Manual** | 8 days | **1/15/19 8:00 AM** | **1/24/19 5:00 PM** | **41** | **Sadia Manzoor;PC;MS W...** |
| Select Tools and Technologies | 2 days | 1/15/19 8:00 AM | 1/16/19 5:00 PM | | |
| Make User Interfaces | 4 days | 1/17/19 8:00 AM | 1/22/19 5:00 PM | 43 | |
| Give Description of UI | 4 days | 1/17/19 8:00 AM | 1/22/19 5:00 PM | 43 | |
| Review and Refine UI | 2 days | 1/23/19 8:00 AM | 1/24/19 5:00 PM | 45 | Dr Umar Rasheed |
| **Make Software Test Document** | 6 days | **1/25/19 8:00 AM** | **2/1/19 5:00 PM** | **46** | |
| Make Test Cases | 4 days | 1/25/19 8:00 AM | 1/30/19 5:00 PM | | MS Word |
| Review and Refine Test Document | 2 days | 1/31/19 8:00 AM | 2/1/19 5:00 PM | 48 | Dr Umar Rasheed |
| Review Analysis and Design Document | 10 days | 2/4/19 8:00 AM | 2/15/19 5:00 PM | 46;49 | Dr Umar Rasheed |
| Provide 1st Deliverable | 1 day | 2/18/19 8:00 AM | 2/18/19 5:00 PM | 50 | |
| Project Implementation | 100 days | 2/19/19 8:00 AM | 7/8/19 5:00 PM | 51 | Eclipse;Sadia Manzoor;Dr U... |

Figure 2.1 Project Timetable

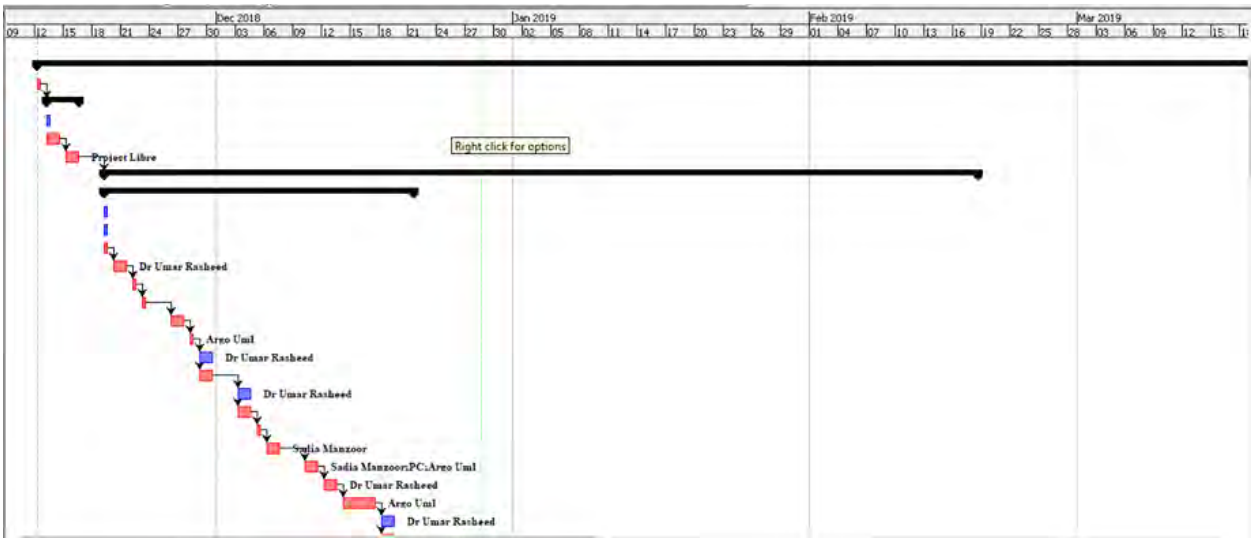



Figure 2.2 Project Gantt Chart

# CHAPTER # 03

# SOFTWARE REQUIREMENTS SPECIFICATION

# INTRODUCTION

This chapter aims at defining the overall Software Requirements for "I-Cuboid" tool. Efforts have been made to define the requirements accurately. The final product will be having features mentioned in this document. In short, we can say that this chapter provides an overview of the proposed system.

## Purpose

The purpose of this SRS document is to provide a detailed overview of software product. It will illustrate the purpose and complete declaration for the development of system. This document covers the functional as well as non-functional requirements of the system. All of the required feature of the project expressed by the client and agreed upon by the developer will be expressed in this document.

## Problem Definition

Typically, search engines display image search results arranged in 2-dimensional grid presentation where user can browse images left to right and top to bottom i.e., image exploration via horizontal and vertical scanning of 2-dimensional grid. So, when we have large number of image search results, horizontal and vertical scanning makes the image exploration process tedious. Therefore, in-depth browsing of image search results in not possible in existing representation approaches.

## Proposed Solution

I-Cuboid provides an online image exploration tool allowing image exploration in more interactive way. The tool retrieve image search results from Flicker search engine and arrange and display them in a grid and in cubic representation. The cubic representation, not only allows horizontal and vertical scanning but also in-depth browsing of image search results in which user can navigate plane by plane.

## Scope

I-Cuboid in an online, web based, image exploration tool that can be used by any user.

The software will allow the user to enter textual query as input. The query is then forwarded to Flicker search engine by calling its Flicker.photos.search API and the set of image results (along with meta-data) are retrieved. The retrieved images are arranged in grid and cubic presentations.

Then, the image results are ranked by considering the similarity between user's query and the images. Images ranking is totally based and dependent on textual meta-data available with retrieved images. After ranking, the tool arranges the ranked images in grid and in cubic presentations.

The major functionality of the system is as following:

- Query formulation by simple keywords.
- Send query to Online Image Source (Flicker Search Engine).
- Get textual information associated image results (textual metadata).
- Display and arrange the images in grid and cubic presentations.
- Rank image results by using textual meta-data of images via Jaccard Index.
- Display and arrange ranked images in grid and cubic presentations.
- Switch presentation mode (from grid to cube and vice versa).
- Switch images rank mode (from Default_Rank to New_Rank and vice versa).
- Provide options to see the images of each plane in cube (Show Plane 1, Show Plane 2 etc).
- Display neighbor images of each image in cubic planes.
- Error Messages:
    - When user enters any invalid query, an error message of "No Images Found" will be displayed on the screen.
    - When user start typing query in the search bar, only then search button will be enabled, otherwise it will remain disable.

## Constraints

- Web based tool
- User must have internet connection to use this tool.

## Objectives

- Search engines usually give the presentation of image search results via 2-D grid presentation, where in-depth browsing of the search results is not possible. This tool

represents the image search results in a cubic representation in three dimensions that enables the horizontal, vertical as well as in-depth browsing of the search results.

- Being an interactive tool, it provides convenient interface to the user for image searching and exploring.
- As this tool gets images from a search engine which are already ranked and then finding their similarities with user query will provide more relevant set of images against the user query.

### Overview

The remaining part of this chapter is divided into two sections, Section 2 and Section 3 and covers a general description, including characteristics of the users of this project, the product's hardware, and the functional and data requirements of the product.

General description of the project is discussed in section 2 of this document. Section 3 gives the functional requirements, data requirements and constraints and assumptions made while designing the software, the specific requirements of the product, the external interface requirements and gives detailed description of functional requirements.

## OVERALL DESCRIPTION

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

### Product Perspective

I-Cuboid is dependent tool as it depends on a search engine for retrieving images from it.

Figure 3.1 given below illustrates the interactions between the user, I-Cuboid tool and Flicker search engine.

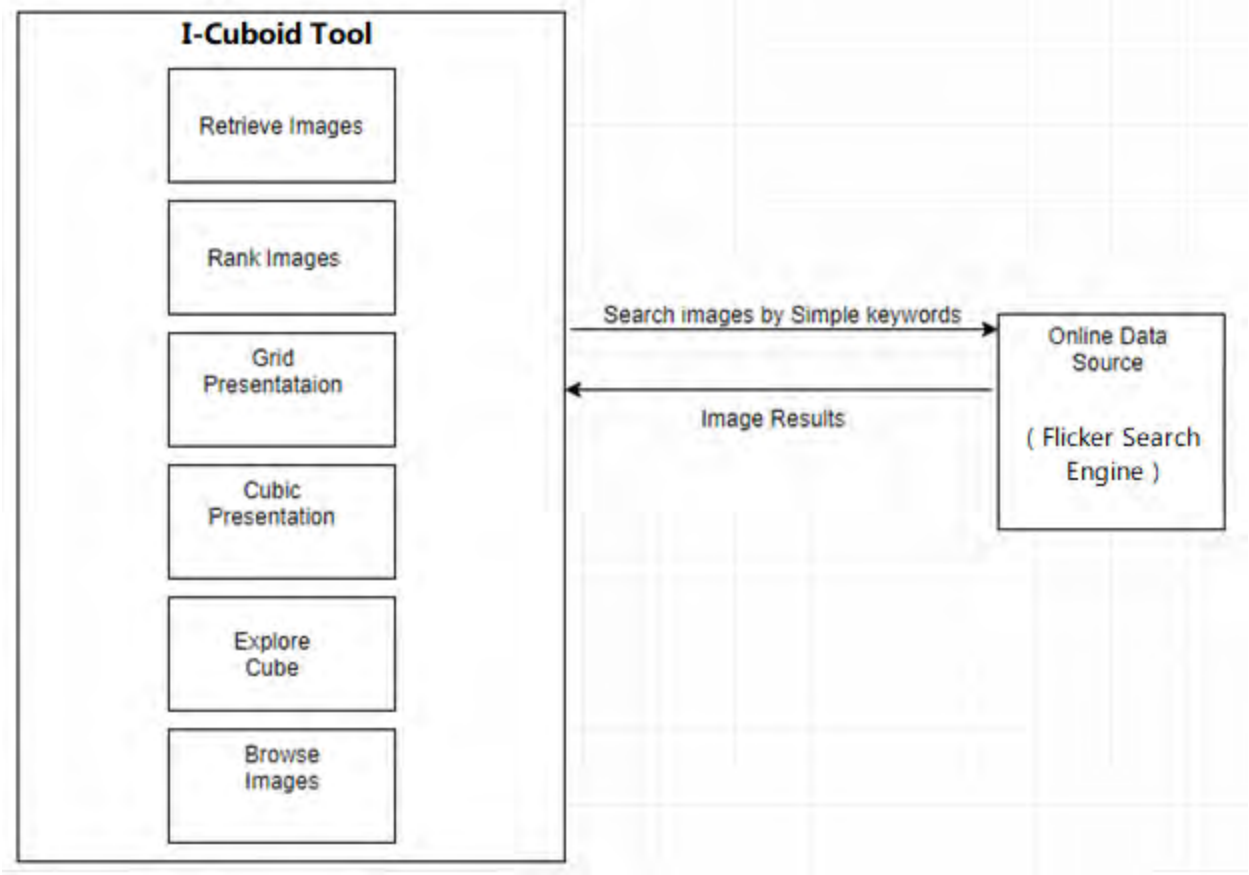



Figure 3.1 System Block Diagram

## User Characteristics

This tool is for all the users.

User is expected to be

- Educated
- Familiar with computer
- Internet literate and be able to use a search engine.

## Assumptions and Dependencies

- Tool is dependent on access to Internet.
- Tool is dependent on search engine for image retrieval.
- Since this is a web based tool, so there is a need for the web browser.

# EXTERNAL INTERFACE REQUIREMENTS

## User Interfaces

The background will be light in color with dark colored font to enhance the contrast and visibility. Controls which allow the user to interact with the application will be clear and imply their functionality within the application. Icons and fonts will be kept simple to make users understand the function of features easily. Error messages will be displayed to the user for example Result Not Found etc.

## Hardware Interfaces

Keyboard and mouse is used for input and monitor is to display output.

Since the tool is online, so internet connection is must.

## Software Interfaces

- Operating System: Windows (Windows 7 and above)
- Web Brower: Internet Explorer (8.0 and above), Mozilla Firefox (3.0 and above), or Google Chrome

## Communication Interfaces

As the system is internet based, therefore it will require some standard networking protocols for communication. These protocols are usually installed automatically by the operating system running on server.

- HTTP: It is a protocol used by the WWW service to make communication possible between a web server and a web browser.
- TCP/IP: It is a protocol used to communicate data all around the internet.

# SOFTWARE PRODUCT FEATURES

Summary of functions that the system should perform are listed below:

**Search by Keywords:** This function will allow the user to enter textual query for retrieving images.

**Retrieve Images:** This function will automatically retrieve a set of images from Flicker search engine.

**Arrange Images In Grid**: This function will arrange the images in grid presentation automatically after image retrieval.

**Browse Images In Grid:** This function will allow the user to browse images arranged and displayed in grid by horizontal vertical scanning (scrolling vertically).

**Rank Images**: This function will rank the retrieved set of images.

**Switch Results:** This function allows switching between retrieved results and ranked_retrieved results.

**Arrange Images In Cube:** This function will arrange the images in cubic presentation after image retrieval and ranking.

**Browse Images:** This function will allow in-depth browsing (horizontal, vertical and in-depth scanning) of images arranged in cube.

**View Image from Grid:** This function will allow the user to view any image from grid simply by clicking on that image and it will be opened.

**View Image from Cube:** This function will allow the user to view any image from cube simply by clicking on that image and it will be opened.

**Select Plane:** This function will allow the user to select any plane from the cube to view its images.

**Display Neighbors:** This function will allow the user to view neighbor images of any image in cube.

**Download Image:** This function will allow the user to download any image from grid or cube.

**Next Images in Grid:** This function will allow the user to arrange next one hundred and twenty five images in grid simply by clicking on next button.

**Previous Images in Grid:** This function will allow the user to go back to the previous one hundred and twenty five images in grid by simply clicking on previous button.

**Next Images in Cube:** This function will allow the user to arrange next one hundred and twenty five images in cube simply by clicking on next cube button.

**Previous Images in Cube:** This function will allow the user to go back to the previous one hundred and twenty five images in cube by simply clicking on previous cube button.

# USE CASES

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

## Use Cases List

- Search By Keywords
- View Images In Grid
- View Images In Cube
- Show Image View
- Download Image
- Hide Image View
- Select Plane
- View Neighbor Images
- Switch Rank Results
- Switch Presentation mode
- Get Next Images
- Get Previous Images

## Use Case Diagram

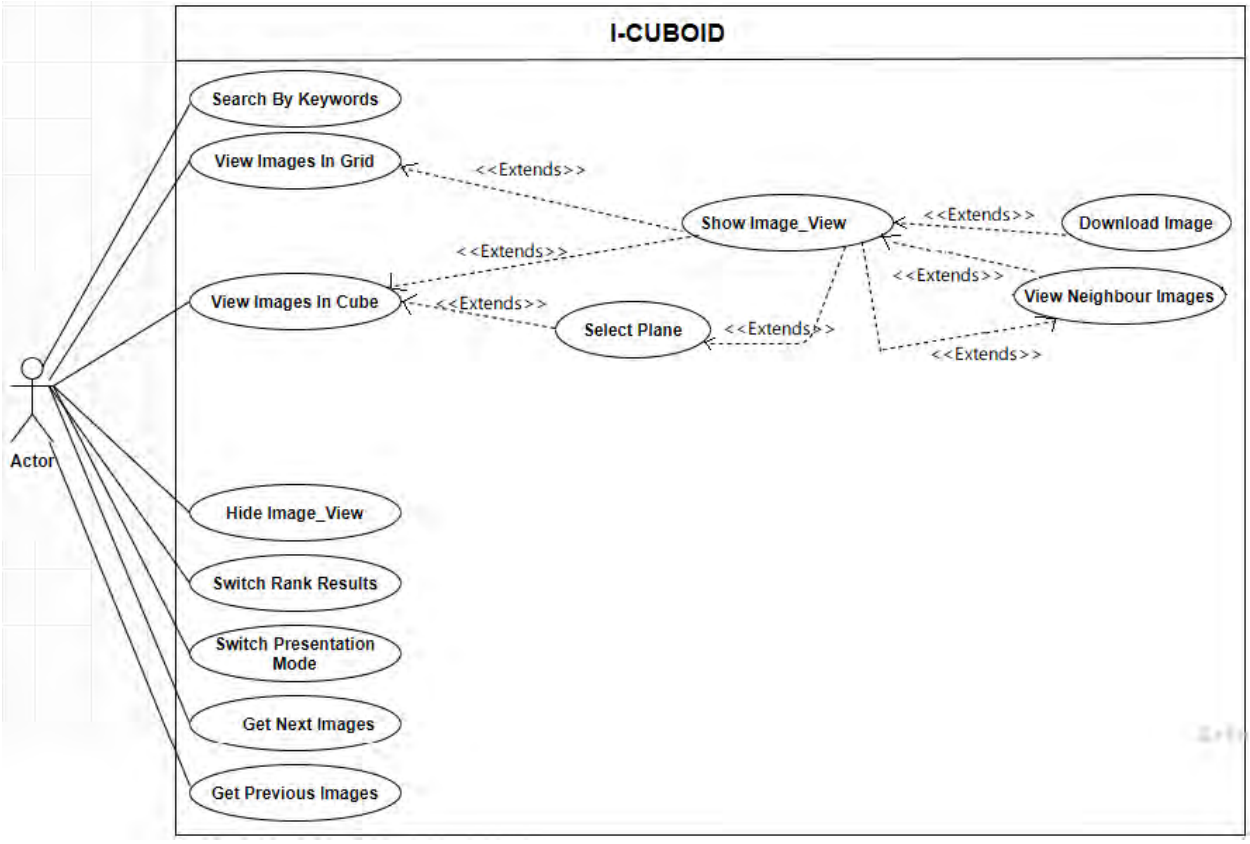



Figure 3.2 Use case Diagram

## Use Case Description

### Special Requirements

- Internet connection is available.
- Text must be visible to the users.

Table 3. 1 UCD for Search by Keywords

| **UC-1: Search By Keywords** | |
|---|---|
| **Primary actor** | User |
| **Description** | This use case allows the user to enter textual query in search field and search images. |
| **Pre-condition** | User must have opened the tool. |

| | |
|---|---|
| **Post-condition** | System opens up the screen on which image results are displayed in grid. |
| **Main Success Scenario** | 1. System displays home page to user to enter query in search field.<br>2. User clicks on the search field to enable it.<br>3. System makes the cursor to start blinking in the search field.<br>4. User starts inputting textual query in the search field.<br>5. User submits the query.<br>6. System starts processing query and a loader appears.<br>7. System comes up with image results displayed in grid. |
| **Alternative Scenario** | *Server is down.<br>*Internet is not available.<br>5a) If user enters nothing in the search field<br>System never allow user to submit any empty query because the submission of query is allowed only when user start typing in the search field.<br>7a) If user enters any invalid query having random characters<br>Systems gives error message that no images are found and give a suggestion to use another query. |
| **Frequency** | Many times a day. |

Table 3. 2 UCD for View Images in Grid

| **UC-2: View Images in Grid** | |
|---|---|
| **Primary actor** | User |
| **Description** | This use case enables the user to view images arranged in grid. |
| **Pre-condition** | User entered query for images. |
| **Post-condition** | Images are viewed by user in grid. |
| **Main Success Scenario** | 1. System retrieved image results against query.<br>2. System arranges image results in grid and displayed to user.<br>3. User views images in grid.<br>4. System provides the option of scrolling web page down.<br>5. User scrolls down the web page to view more images.<br>6. System scrolled the web page down. |

| | |
|---|---|
| | |
| **Alternative Scenario** | *Server is down.<br>*Internet is not available. |
| **Frequency** | Many times a day. |
| **Extension Point** | *Image_View* may occur after step 3. |

Table 3. 3 UCD for View Images in Cube

| **UC-3: View Images in Cube** | |
|---|---|
| **Primary actor** | User |
| **Description** | This use case enables the user to view images arranged in cube. |
| **Pre-condition** | User switched view from Grid to Cube. |
| **Post-condition** | Images are viewed by user in Cube. |
| **Main Success Scenario** | 1. System arranges image results in cube and displayed to user.<br>2. User views images in cube. |
| **Alternative Scenario** | *Server is down.<br>*Internet is not available. |
| **Frequency** | Many times a day. |
| **Extension Point** | *Image_View* may occur after step 2.<br>*Switch_Plane* may occur after step 2. |

Table 3. 4 UCD for Show Image View

| **UC-4: Show Image View** | |
|---|---|
| **Primary actor** | User |
| **Description** | This use case allows the user to open and view an image in grid as well as in cube.<br>This use case extends "View Images in Grid, View Image in Cube, Select Plane and View Neighbour Images" use cases at extension point *Image_View*. |
| **Pre-condition** | Images are fetched against the query and arranged in grid as well as in cube. |
| **Post-condition** | An image is viewed by user. |
| **Main Success Scenario** | 1. System displays image results arranged in any presentation mode.<br>2. User selects an image.<br>3. System gives detailed view of the selected image along with information including its owner name, number of likes, number of comments, number of views, date posted, time posted and its format.<br>4. User views the image as well as the information. |
| **Alternative Scenario** | *Server is down.<br>*Internet is not available.<br>1a) Presentation mode can be one of the following:<br>-Grid View<br>-Cubic View |
| **Frequency** | Many times a day. |
| **Extension Point** | *Image Save* may occur after step 3. |

Table 3. 5 UCD for Download Image

| **UC-5: Download Image** | |
|---|---|
| **Primary actor** | User |
| **Description** | Use case "Download Image" is performed by user after which user can download any desired image.<br>This use case extends " Show Image View" use case. It is inserted at extension point Image Save. |
| **Pre-condition** | User has selected an image by clicking on it and image detailed view opened. |
| **Post-condition** | Image is downloaded. |
| **Main Success Scenario** | 1. System provides option to the user for image downloading.<br>2. User selects the option.<br>3. System downloads that image. |
| **Alternative Scenario** | *Server is down.<br>*Internet is not available. |
| **Frequency** | Many times a day. |

Table 3. 6 UCD for Hide Image View

| **UC-6: Hide Image View** | |
|---|---|
| **Primary actor** | User |
| **Description** | This use case enables the user to hide the detailed view of an image. |
| **Pre-condition** | User has opened an image in grid or in cube by clicking on it. |
| **Post-condition** | Detailed view of the image is hidden. |
| **Main Success Scenario** | 1. User clicks anywhere on the screen other than image detailed_view.<br>2. System hides the detailed_view. |

| | |
|---|---|
| **Alternative Scenario** | *Server is down.<br>*Internet is not available. |
| **Frequency** | Many times a day. |

Table 3. 7 UCD for Select Plane

| **UC-7: Select Plane** | |
|---|---|
| **Primary actor** | User |
| **Description** | This use case is performed by user after which user can select any plane to view its images.<br>It extends "View Images in Cube" use case at extension point *Switch_Plane*. |
| **Pre-condition** | User has switched view from grid to cube and image results are displayed in cube. |
| **Post-condition** | The desired plane got selected by the system in cubic presentation. |
| **Main Success Scenario** | 1. System provides options for each plane to be selected.<br>2. User selects an option.<br>3. System selected the chosen plane. |
| **Alternative Scenario** | *Server is down.<br>*Internet is not available.<br>2a) As the cubic presentation consists of five planes so the options available to select planes are:<br>-Show Plane 1<br>-Show Plane 2<br>-Show Plane 3<br>-Show Plane 4<br>-Show Plane 5 |
| **Frequency** | Many times a day. |
| **Extension Point** | *Image_View* may occur after step 3. |

Table 3. 8 UCD for View Neighbour Images

| **UC-8: View Neighbour Images** | |
|---|---|
| **Primary actor** | User |
| **Description** | This use case is performed by user after which user can view neighbour images of any image in cube.<br>It extends "Show Image View" use case at extension point *Show_Neighbours*. |
| **Pre-condition** | User has switched view from grid to cube and opened an image. |
| **Post-condition** | The desired plane got selected by the system in cubic presentation. |
| **Main Success Scenario** | 1. System display selected image along with its details and an option to show neighbours.<br>2. User selects that option.<br>3. System display neighbour images of selected image. |
| **Alternative Scenario** | *Server is down.<br>*Internet is not available. |
| **Frequency** | Many times a day. |
| **Extension Point** | *Image_View* may occur after step 3. |

Table 3. 9 UCD for Switch Rank Results

| **UC-9: Switch Rank Results** | |
|---|---|
| **Primary actor** | User |
| **Description** | This use case is performed by user in which image results are toggled between default rank results and new rank results. |
| **Pre-condition** | User entered a query upon which image results are fetched and arranged in grid and in cube. |
| **Post-condition** | Image results are switched, either from Default rank results to New rank results or vice versa. |

| | |
|---|---|
| **Main Success Scenario** | 1. System displays image results arranged in grid.<br>2. System provides option to switch the rank results.<br>3. User selects the switch option.<br>4. System switched the ranked image results. |
| **Alternative Scenario** | *Server is down.<br>*Internet is not available.<br>2a) If the current rank mode is default_rank, then system switches it to new_rank mode.<br>If the current rank mode is new_rank, then system switches it to new_rank mode. |
| **Frequency** | Many times a day. |

Table 3. 10 UCD for Switch Presentation Mode

| **UC-10: Switch Presentation Mode** | |
|---|---|
| **Primary actor** | User |
| **Description** | This use case is performed by user after which user can switch presentation mode from grid to cube or from cube to grid. |
| **Pre-condition** | User entered a query upon which image results are fetched and arranged in grid and in cube. |
| **Post-condition** | Presentation mode is switched, either from Grid_to_Cube or vice versa. |
| **Main Success Scenario** | 5. System provides option to switch the presentation mode.<br>6. User selects the switch option.<br>7. System switched the presentation mode. |
| **Alternative Scenario** | *Server is down.<br>*Internet is not available.<br>2a) If the current presentation mode is Grid, then system switches it to cubic presentation.<br>If the current presentation mode is Cube, then system switches it to grid presentation. |
| **Frequency** | Many times a day. |

Table 3. 11 UCD for Get Next Images

| **UC-11: Get Next Images** | |
|---|---|
| **Primary actor** | User |
| **Description** | One hundred and twenty five images will be displayed in grid as well as in cube at a time. So this use case is performed by user after which user can get next one hundred and twenty five images displayed in grid or in cube. |
| **Pre-condition** | User entered a query upon which image results are fetched and arranged in grid as well as in cube. |
| **Post-condition** | System arranged and displayed next one hundred and twenty five images in grid or cube. |
| **Main Success Scenario** | 1. System provides option to get next images.<br>2. User selects the next option.<br>3. System comes up with next set of images arranged in whatever the current presentation mode is. |
| **Alternative Scenario** | *Server is down.<br>*Internet is not available.<br>4a) Presentation mode can be:<br>-Grid<br>-Cube |
| **Frequency** | Many times a day. |

Table 3. 12 UCD for Get Previous Images

| **UC-12: Get Previous Images** | |
|---|---|
| **Primary actor** | User |
| **Description** | This use case is performed by user after which user can go back to previous one hundred and twenty five images displayed in grid or in cube. |
| **Pre-condition** | For at least once, user had selected get next images option. |
| **Post-condition** | System arranged and displayed previous one hundred and twenty five images in grid or cube. |

| | |
|---|---|
| **Main Success Scenario** | 1. System provides option to get previous images.<br>2. User selects the previous option.<br>3. System goes back to the previous set of images arranged in whatever the current presentation mode is. |
| **Alternative Scenario** | *Server is down.<br>*Internet is not available.<br>4a) Presentation mode can be:<br>-Grid<br>-Cube |
| **Frequency** | Many times a day. |

# SYSTEM SEQUENCE DIAGRAMS

An interaction diagram which shows the sequence of interactions between the external actor and the system and the events generated by these actors is called System Sequence Diagram.

## SSD for Search by Keywords

Following Figure 3.3 shows the system sequence diagram of search by keywords:

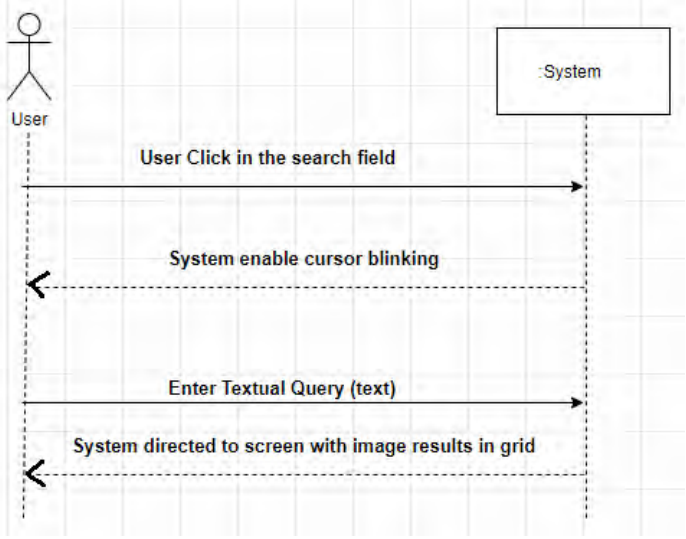



Figure 3.3 SSD for Search by Keywords

## SSD for View Images in Grid

Following Figure 3.4 shows the system sequence diagram of view images in grid:

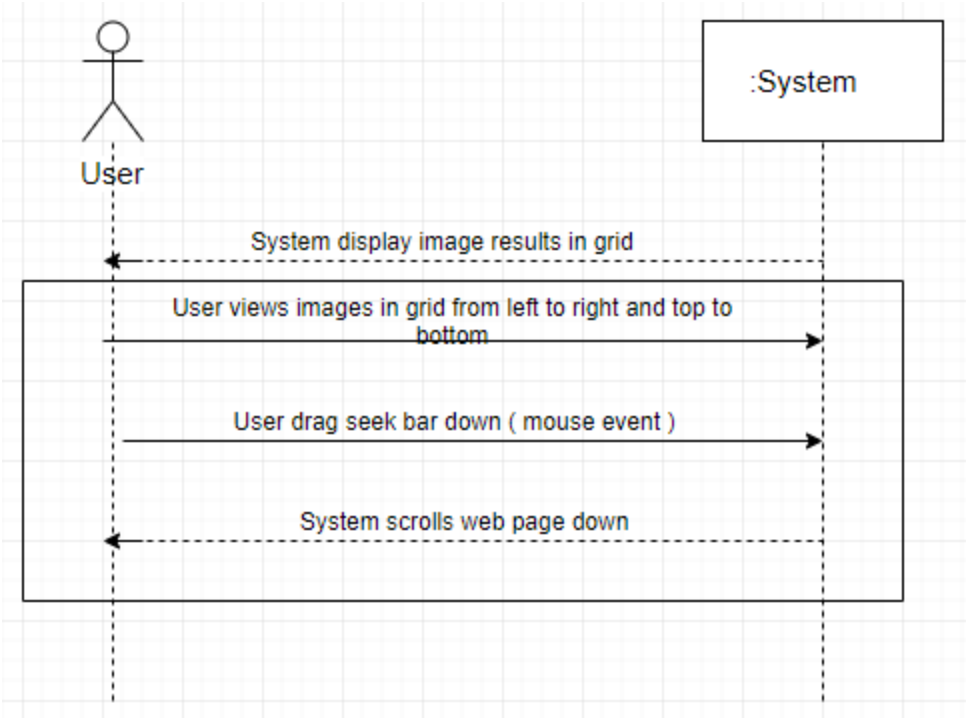



Figure 3.4 SSD for View Images in Grid

## SSD for View Images in Cube

Following Figure 3.5 shows the system sequence diagram of view images in cube:

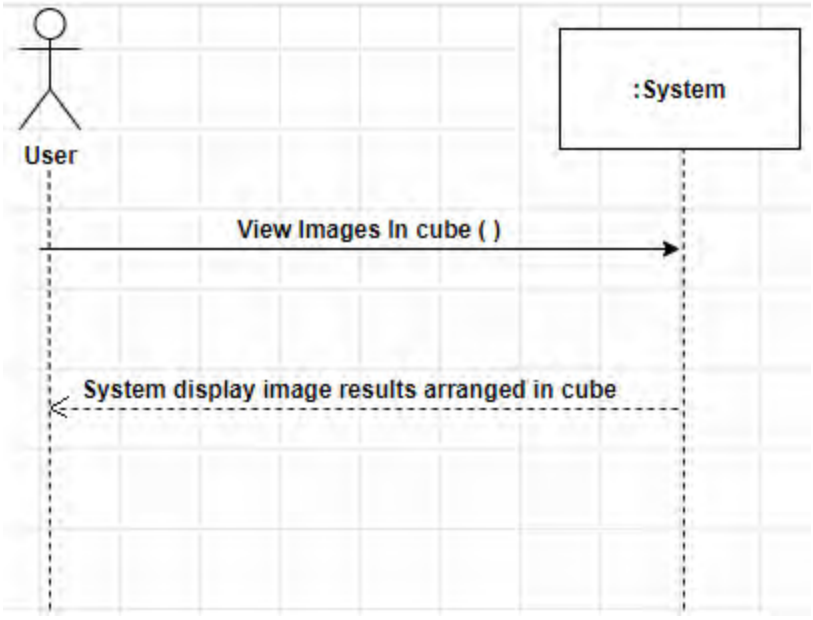



Figure 3.5 SSD for View Images in Cube

## SSD for Show Image View

Following Figure 3.6 shows the system sequence diagram of show image view:

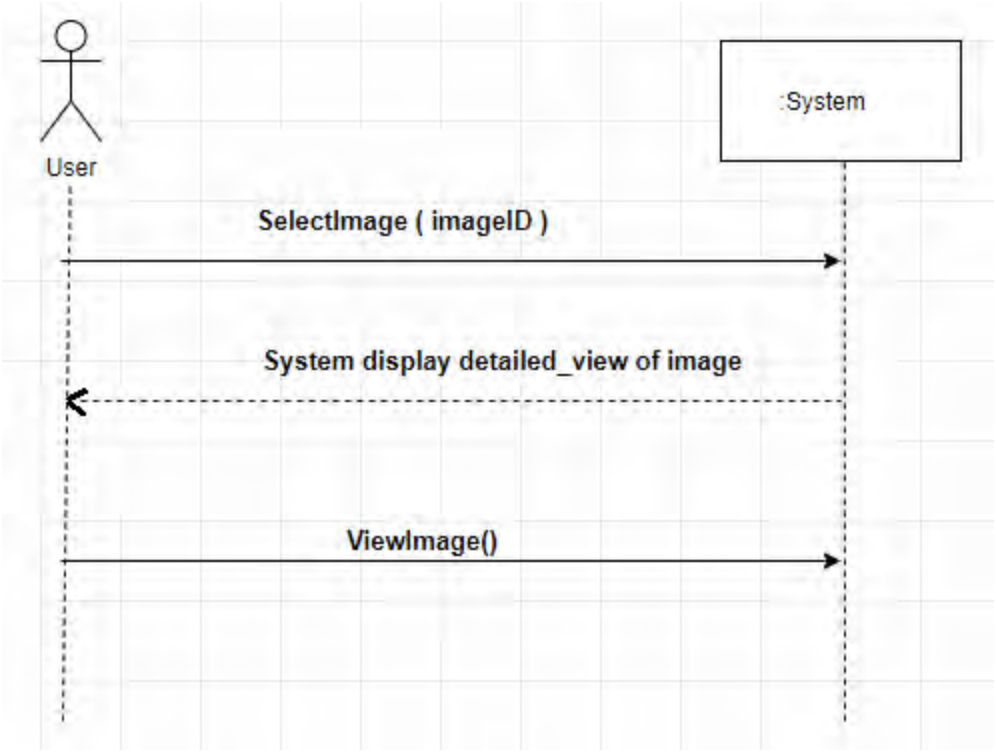



Figure 3.6 SSD for Show Image View

## SSD for Download Image

Following Figure 3.7 shows the system sequence diagram of download image:

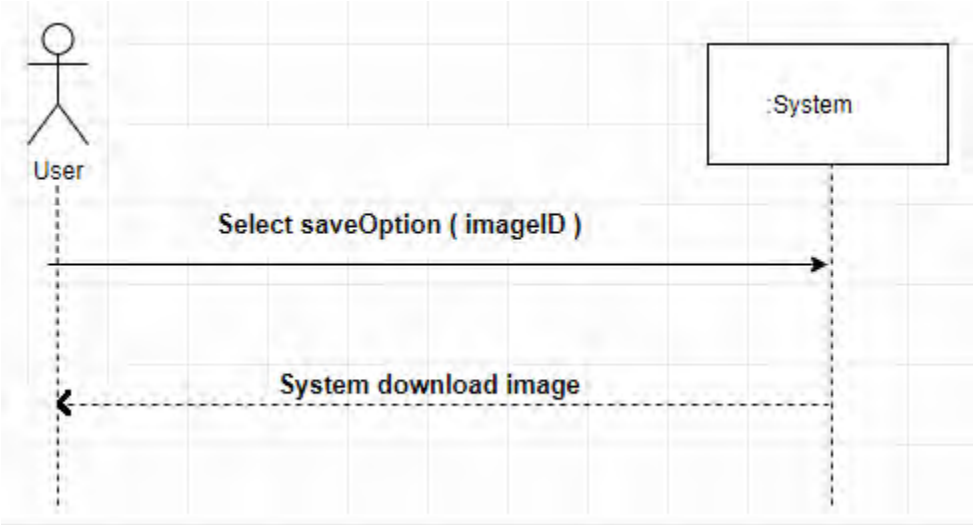



Figure 3.7 SSD for Download Image

## SSD for Hide Image View

Following Figure 3.8 shows the system sequence diagram of hide image view:

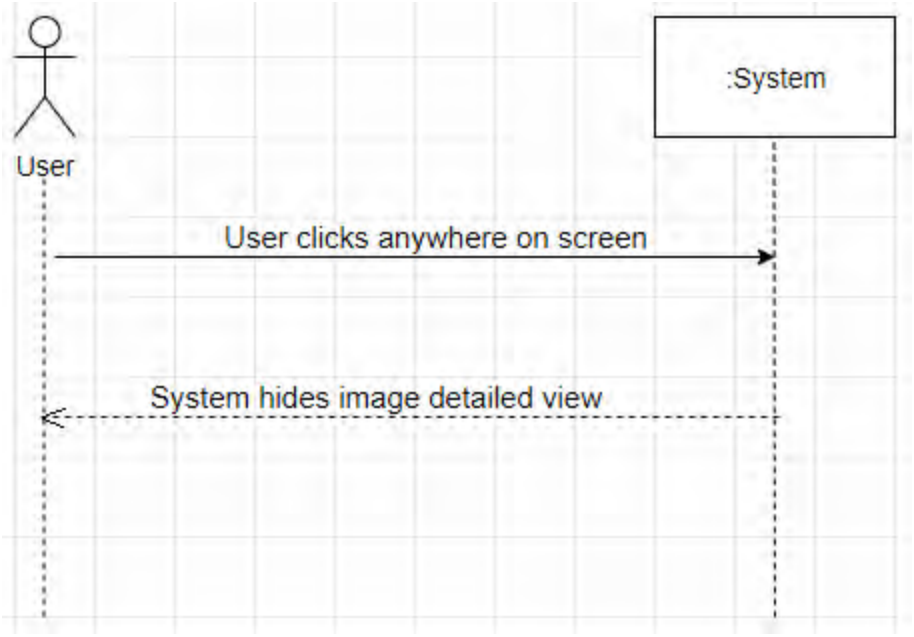



Figure 3.8 SSD for Hide Image View

## SSD for Select Plane

Following Figure 3.9 shows the system sequence diagram of select plane:

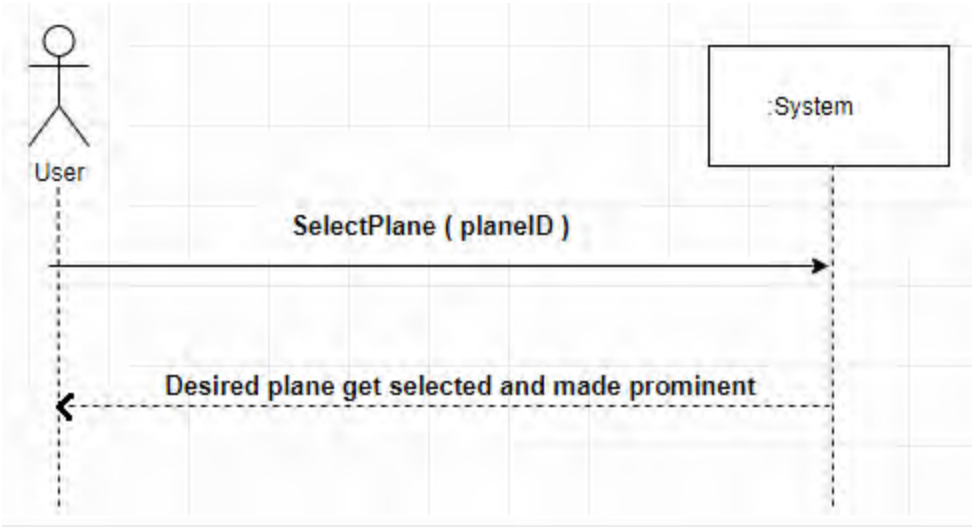



Figure 3.9 SSD for Select Plane

## SSD for View Neighbor Images

Following Figure 3.10 shows the system sequence diagram of view neighbor images:

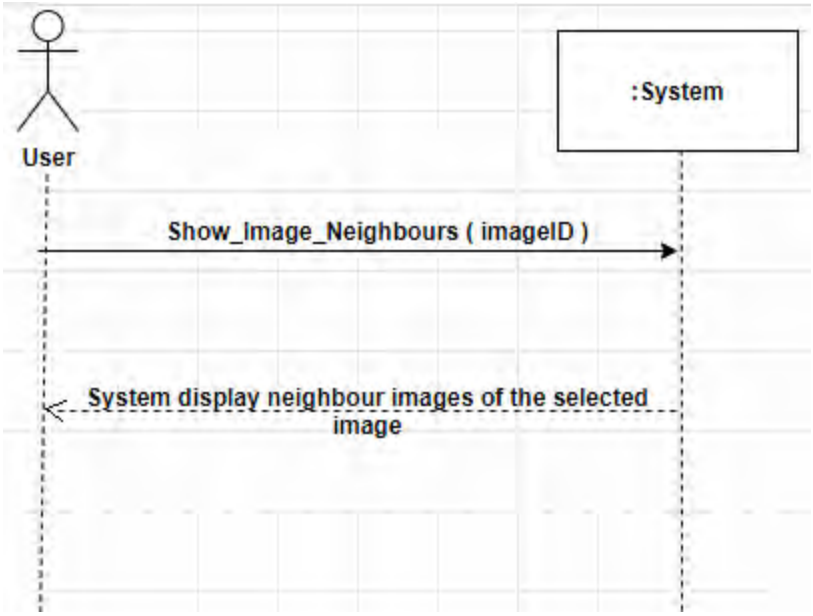



Figure 3.10 SSD for View Neighbor Images

## SSD for Switch Rank Results

Following Figure 3.11 shows the system sequence diagram of switch rank results:

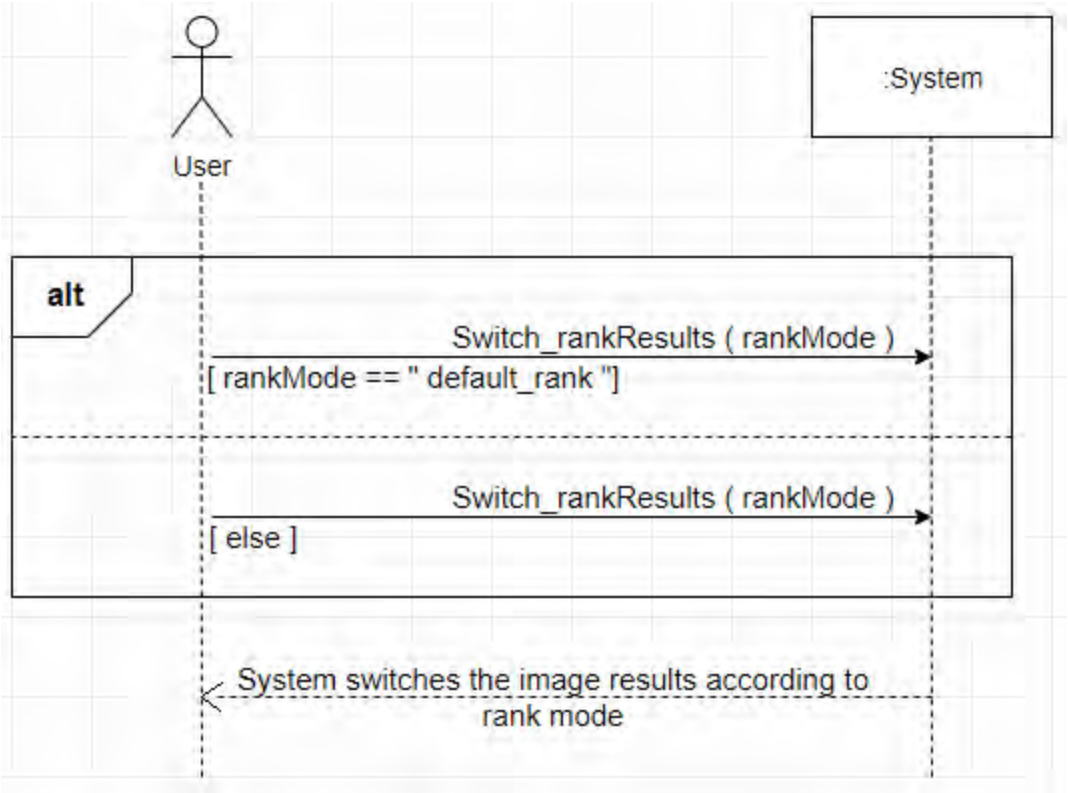



Figure 3.11 SSD for Switch Rank Results

## SSD for Switch Presentation Mode

Following Figure 3.12 shows the system sequence diagram of switch presentation mode:

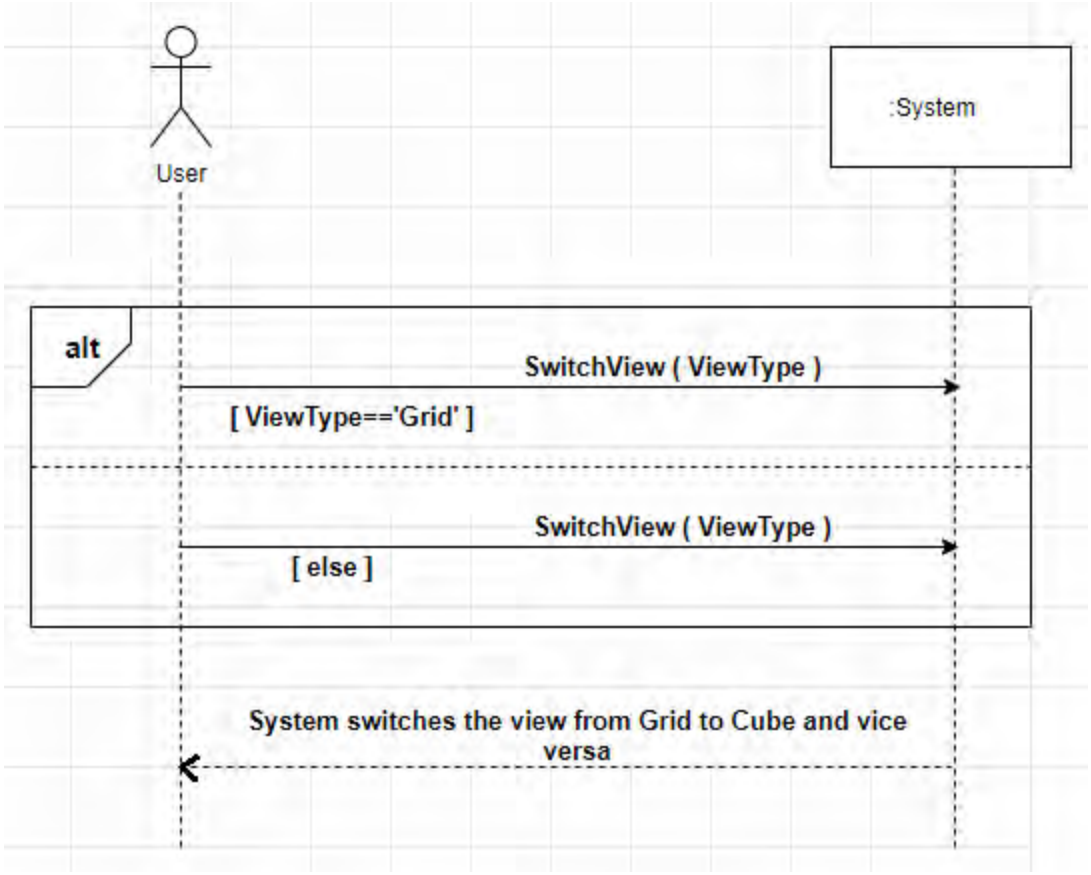



Figure 3.12 SSD for Switch Presentation Mode

## SSD for Get Next Images

Following Figure 3.13 shows the system sequence diagram of get next images:

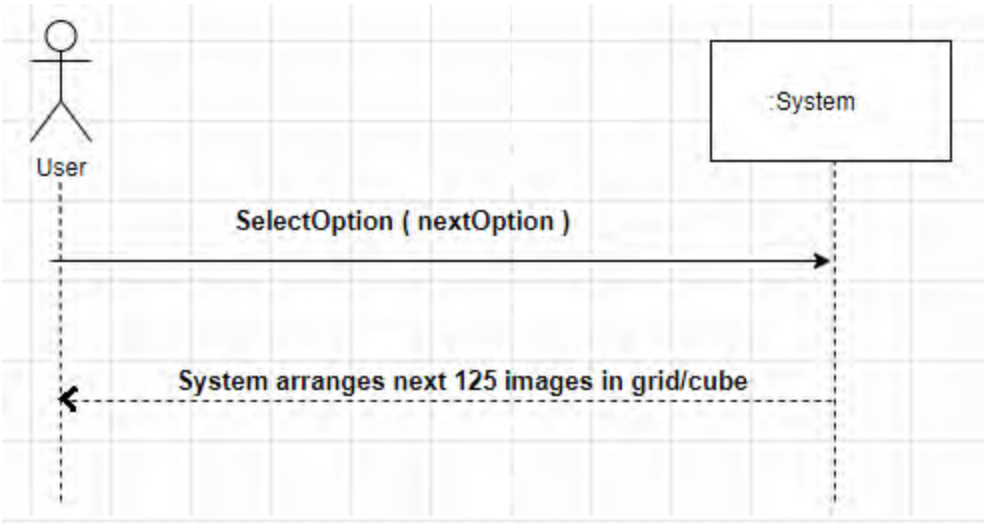



Figure 3.13 SSD for Get Next Images

## SSD for Get Previous Images

Following Figure 3.14 shows the system sequence diagram of get previous images:

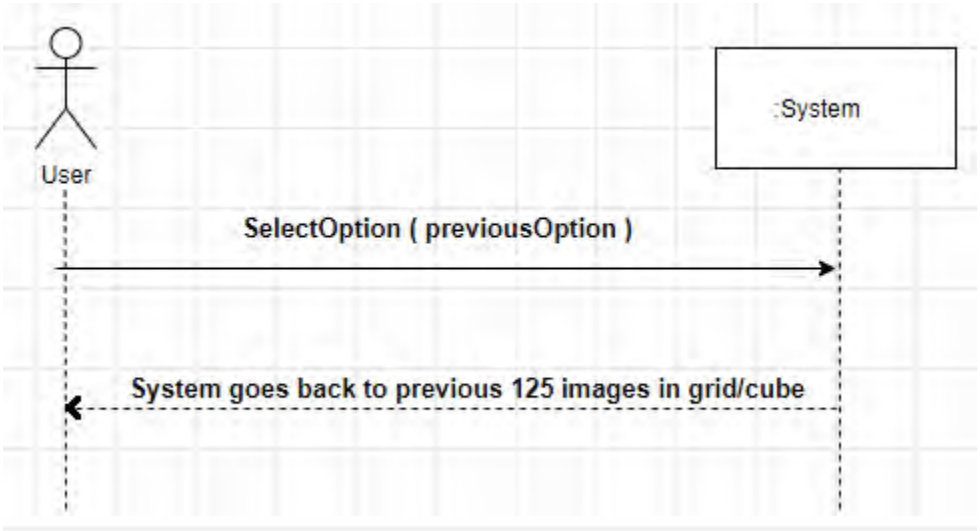



Figure 3.14 SSD for Get Previous Images

# DOMAIN MODEL

Domain model is the visual representation of the decomposition of a domain into individual conceptual classes or objects. It is a way to describe and model real world entities and the relationships between them.

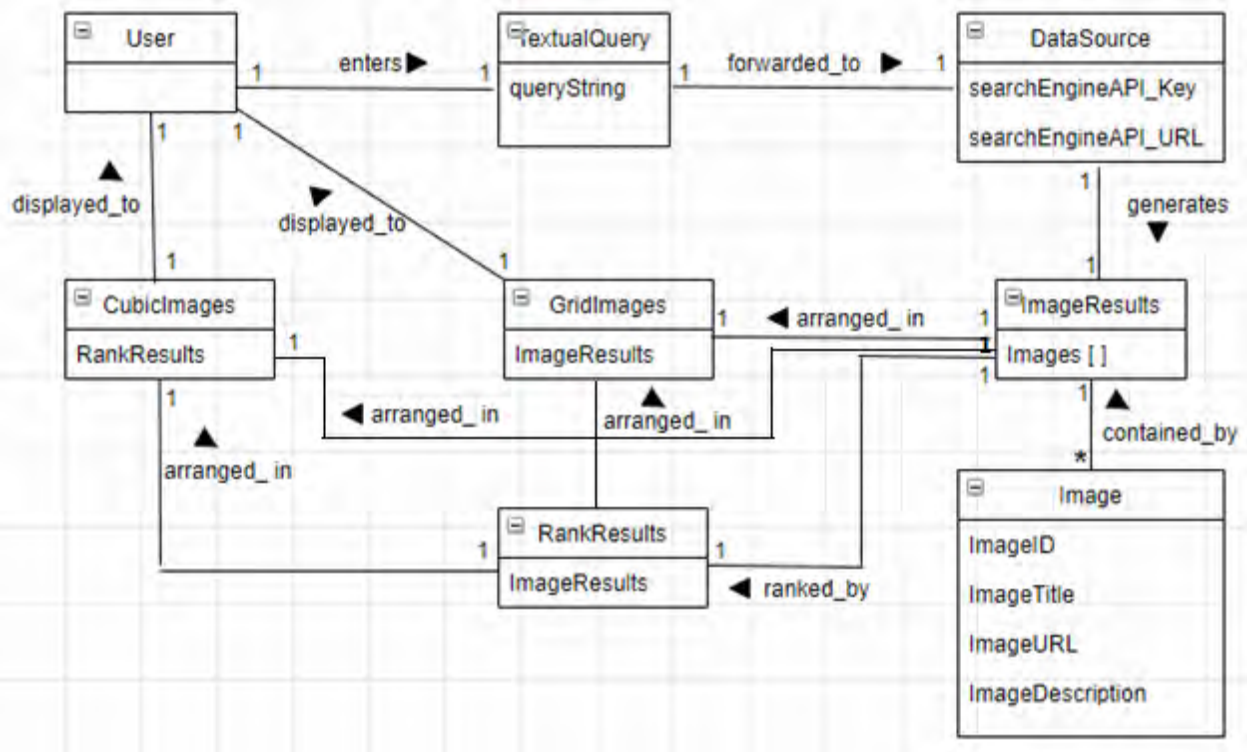



Figure 3.15 Domain Model

# SYSTEM ATTRIBUTES

## Portability

The tool is online web based so it is portable on any system having internet connection.

## Reliability

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

## Availability

The tool should be available to the users all the time provided that internet is available.

## Usability

It specifies how often, how efficiently, how easily and/or how correctly people use this software.

## Performance

As I-Cuboid is an online tool so its performance could be affected by the internet speed and also by the specifications of the device and the web browser. If the user has internet connection then it will display query results to user quickly.

## Maintainability

Code will be well commented such that it will help software developer to understand the code easily that will enable him to make changes or upgrade. Every requirement that user will implement, will be well documented.

# CHAPTER # 04

# SOFTWARE DESIGN DESCRIPTION

# INTRODUCTION

Software Design Document is a written description of a software product. It provides documentation which will be used to aid in software development by providing the details for how the software should be built. Along with written description, this document also has graphical documentation of the software design. It includes the description of how the software will meet the requirements.

## Purpose

The purpose of Software Design Document is to provide design details of I-Cuboid tool. The document contains low level description of the tool, providing insight into the structure and design of each component.

# SYSTEM ARCHITECTURAL DESIGN

System architecture is concerned with understanding of how a system should be organized and designing the overall structure of the system. The output of the architectural design process is an architectural model that describes how the system is organized as a set of communicating components. In the model of the software development process, architectural design is the first stage in the software design process.

## Chosen System Architecture

In this section, architecture of the system is described. Chosen system architecture is MVC (Model View Controller).

**Model**

The Model component corresponds to all the data-related logic that the user works with. This can represent either the data that is being transferred between the View and Controller components or any other business logic-related data.

**View**

The View component is used for all the UI logic of the application.

**Controller**

Controllers act as an interface between Model and View components to process all the business logic and incoming requests, manipulate data using the Model component and interact with the Views to render the final output.

Following Figure 4.1 shows the architecture diagram of the system:

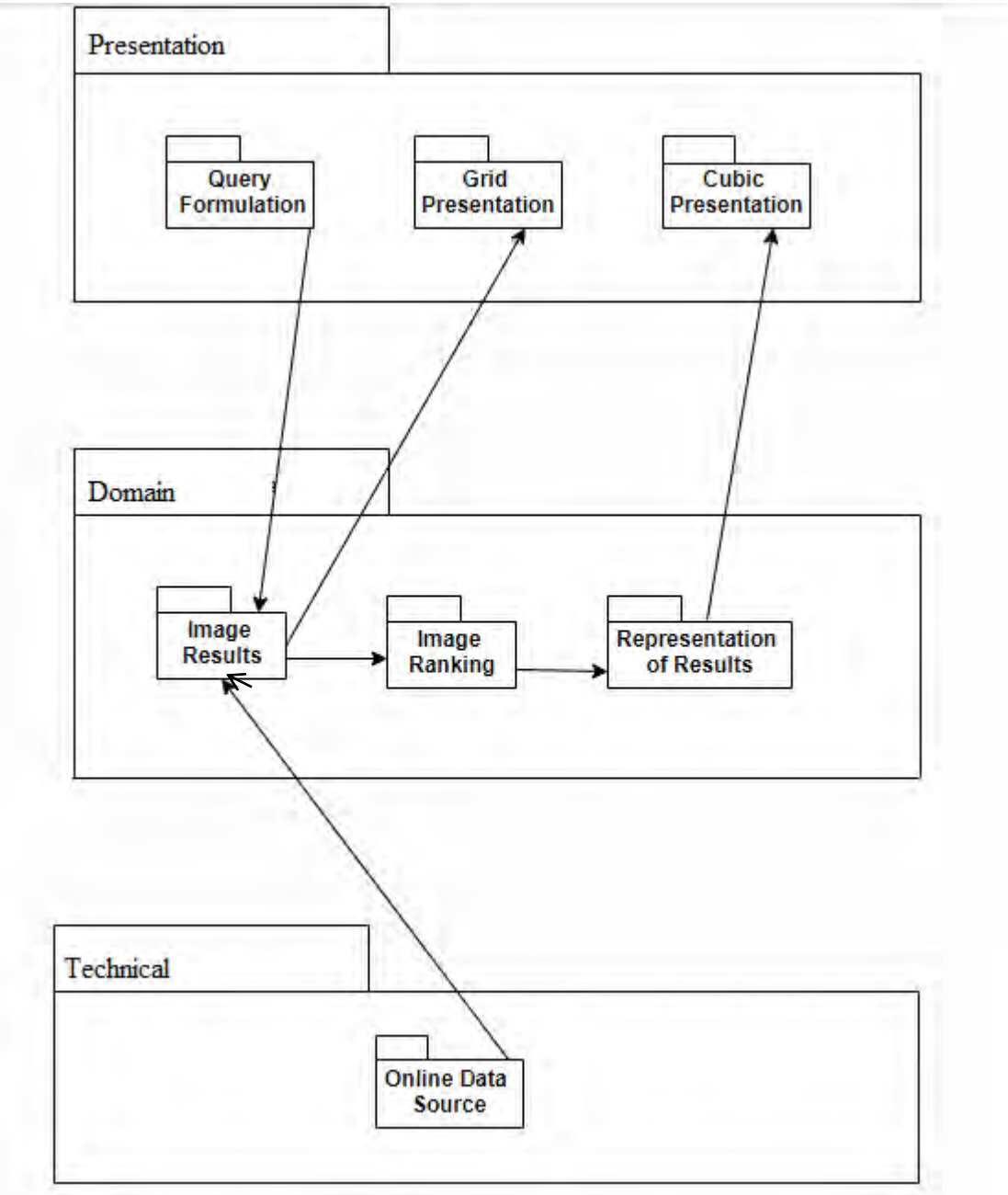



Figure 4.1 System Architecture Diagram

## System Interface Description

Software interface describes the logical characteristics of each interface between the software product and the hardware components of the system. It describes the flow of resources and shows interaction between different entities of the software product.

# CLASS DIAGRAM

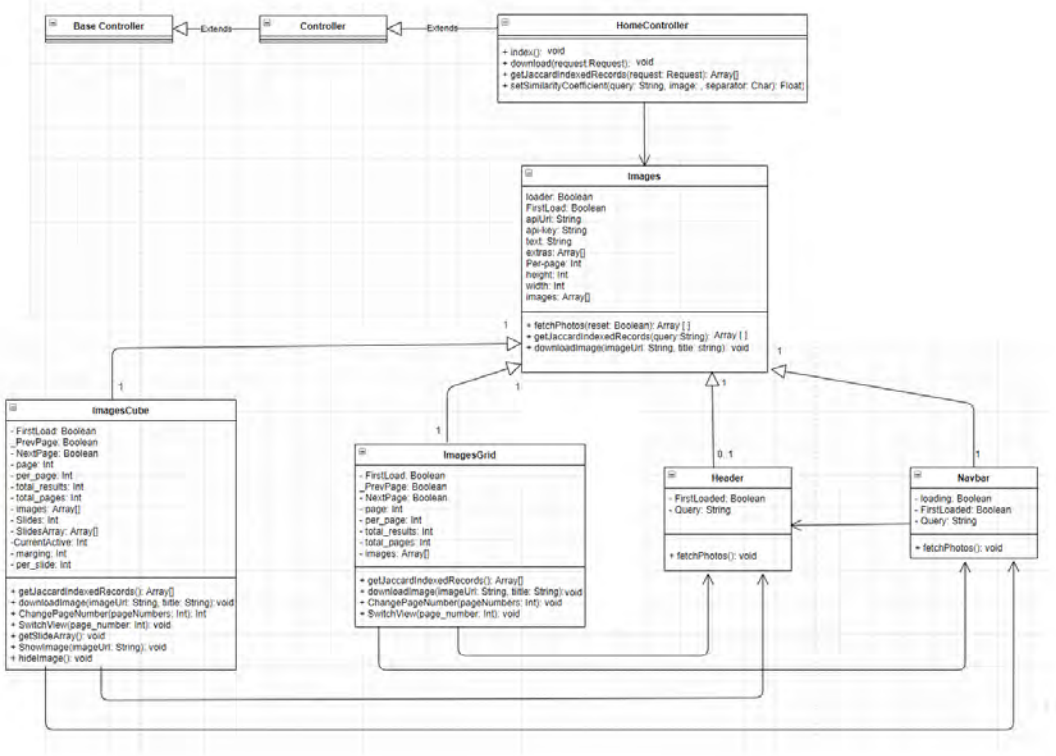



Figure 4.2 Class Diagram

# USER INTERFACE DESIGN

The focus of user interface design is on what actions users might need to do and ensuring that interface provides elements that are easy to understand, easy to access and used to facilitate user actions.

I-Cuboid tool makes sure that user spends most of the time using the application rather than figuring out how to use it. Background will be light in color and font will be dark to enhance

visibility. Cubic representation provides options that will be applied on cubic arrangement of images.

## Screen Images

### Home

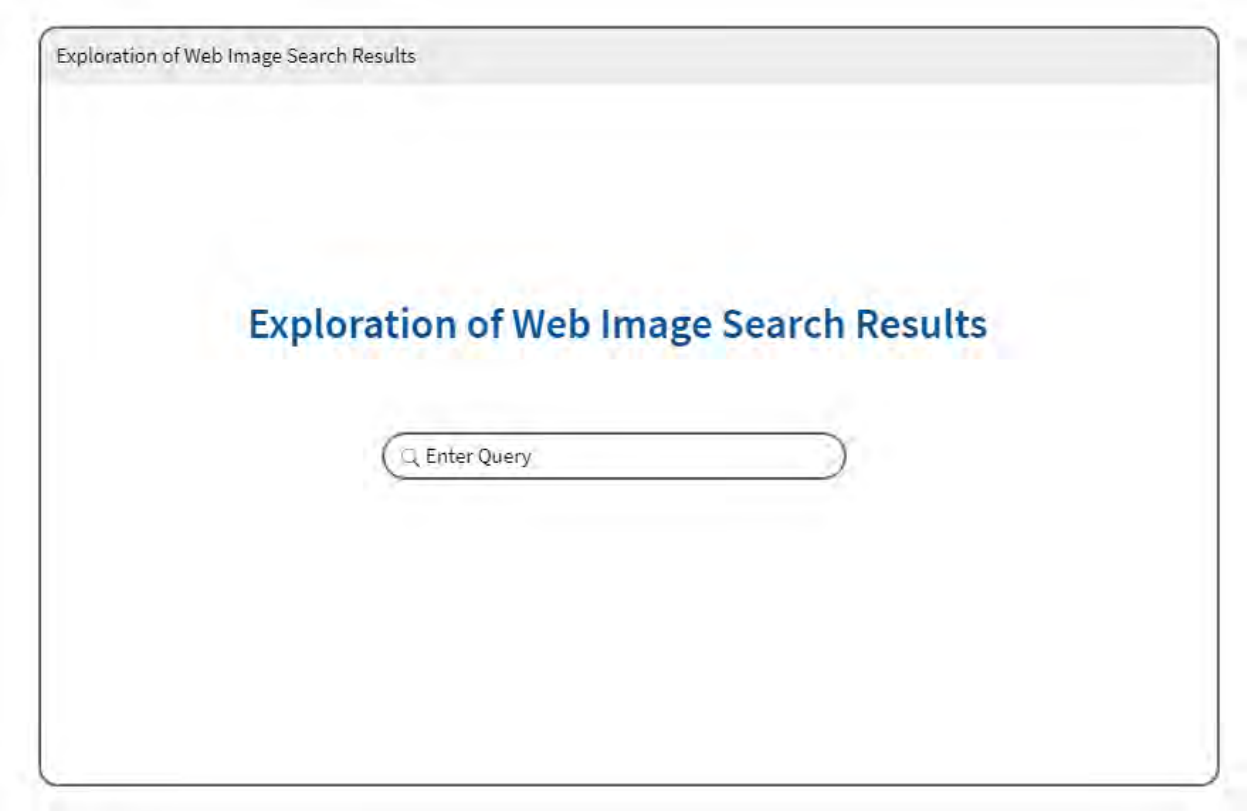



Figure 4.3 Interface for Home Page

## Grid View

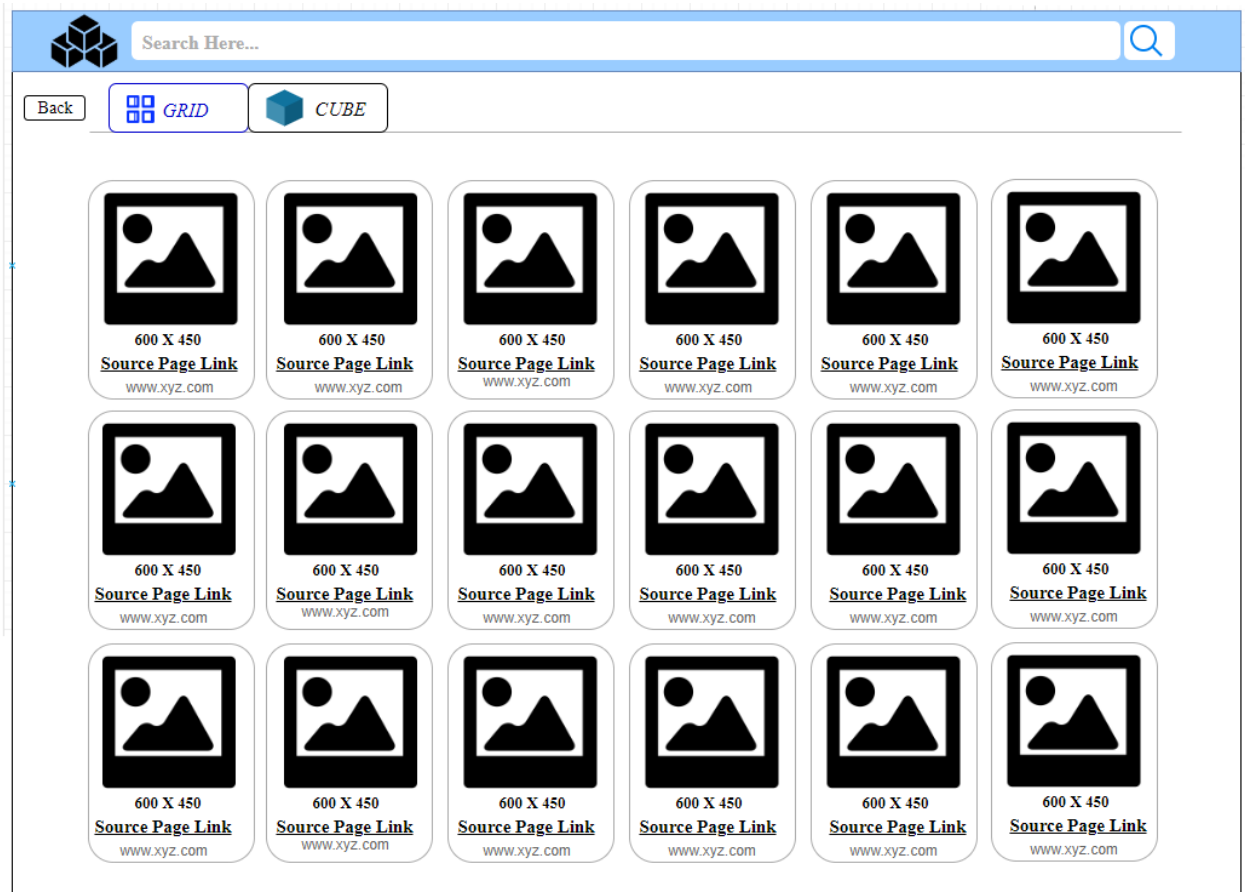



Figure 4.4 Interface for Grid Representation

## Cubic View

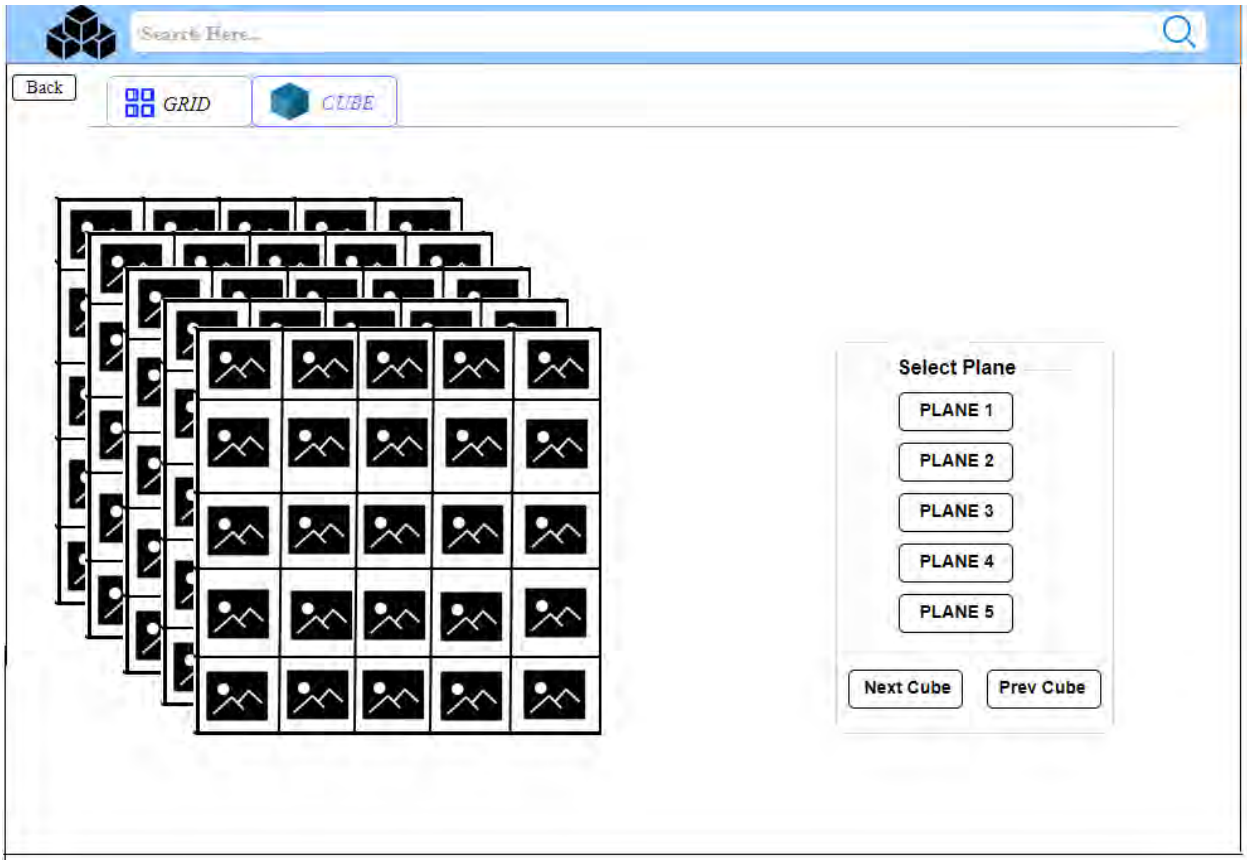



Figure 4.5 Interface for Cubic Representation

# CHAPTER # 05

# SOFTWARE TEST DOCUMENTATION

# INTRODUCTION

Testing is executing a system in order to identify any gaps, errors or missing requirements in contrast to actual requirements. Software test document is a document under which tester will determine whether a system under test satisfies requirements or works correctly or not. Testing analyzes a software item and evaluates its features.

## System Overview

I-Cuboid is a web based tool designed for the exploration of image search results in an interactive way by retrieving image results from an online source (Flicker search engine) and then ranking and arranging those images in a grid and cubic representation.

## Test Approach

Two testing techniques are used for this system.

- User Acceptance Testing consists of a process of verifying that a solution works for the user. It is not System Testing (ensuring that software does not crash or meets documented requirements) but rather ensures that the solution will work for the user (i.e., tests that the user accepts the solution).
- Usability testing is a way to see how easy to use a software product by testing it with real users.

# TEST PLANS

## Features to be Tested

Features to be tested are as following:

- Search By Keywords
- View An Image In Grid
- View An Image In Cubic Arrangement
- Select Plane

## Testing Tools and Environment

Tools and environment used for testing are as following:

- Web Browser
- Laptop/PC

- Windows Operating System

# TEST CASES

Following are the test cases of Exploration of Web Image Search Results tool:

## Search by Keywords

Table 5.1 and Table 5.2 describe the test cases for Search by Keywords.

Table 5. 1 Test case for Search by Keywords for Valid Query

| | |
|---|---|
| **ID** | TC-1 |
| **DESCRIPTION** | Check that valid textual query is entered and results are displayed in a grid. |
| **SETUP** | • Open the tool. |
| **INSTRUCTIONS** | • Display home page.<br>• To search images, input textual query in search field.<br>• Click the submit button. |
| **INPUT** | User inputs query that is easy to understand. |
| **EXPECTED RESULT** | Query is executed, resulting in displaying image results in grid. |
| **ACTUAL RESULT** | As expected. |
| **VERDICT** | Pass. |

Table 5. 2 Test case for Search by Keywords for Invalid Query

| | |
|---|---|
| **ID** | TC-1 |
| **DESCRIPTION** | Check that an invalid textual query is entered and no results are obtained. |
| **SETUP** | • Open the tool. |

| | |
|---|---|
| **INSTRUCTIONS** | • Display home page.<br>• To search images, input textual query in search field.<br>• Click the submit button. |
| **INPUT** | User inputs any random characters in search field. |
| **EXPECTED RESULT** | An error message is displayed that "User query did not match any document, try different keywords." |
| **ACTUAL RESULT** | As expected. |
| **VERDICT** | Pass. |

## View an Image in Grid

Table 5.3 describes the test case for View an Image in Grid.

Table 5. 3 Test case for View Image in Grid

| | |
|---|---|
| **ID** | TC-3 |
| **DESCRIPTION** | To check that user can view an image from grid presentation. |
| **SETUP** | • Open the tool.<br>• Search images by entering query.<br>• Images are arranged in grid. |
| **INSTRUCTIONS** | • Select an image from grid. |
| **INPUTS** | User selects an image. |
| **EXPECTED RESULT** | Selected image appears on the same screen as image slider. |
| **ACTUAL RESULT** | As expected. |
| **VERDICT** | Pass. |

## View an Image in Cubic Arrangement

Table 5.4 describes the test case for View an Image in Cubic Arrangement.

Table 5. 4 Test case for View Image in Cubic Arrangement

| | |
|---|---|
| **ID** | TC-4 |
| **DESCRIPTION** | To check that user can view an image from cubic presentation. |
| **SETUP** | • Open the tool.<br>• Search images by entering query.<br>• Images are arranged in cube. |
| **INSTRUCTIONS** | • Select an image from cube. |
| **INPUTS** | User selects an image. |
| **EXPECTED RESULT** | Selected image appears on the same screen as image slider. |
| **ACTUAL RESULT** | As expected. |
| **VERDICT** | Pass. |

## Select Plane

Table 5.5 describes the test case for Select Plane.

Table 5. 5 Test case for Select Plane

| | |
|---|---|
| **ID** | TC-5 |
| **DESCRIPTION** | To check that any particular plane is selected. |
| **SETUP** | • Open the tool.<br>• Search images by entering query.<br>• Images are arranged in cube. |
| **INSTRUCTIONS** | • Select an option for the selection of a particular plane. |
| **INPUTS** | User selects the option. |

| | |
|---|---|
| **EXPECTED RESULT** | Particular plane is selected on the cube and displayed. |
| **ACTUAL RESULT** | As expected. |
| **VERDICT** | Pass. |

# USABILITY TESTING

Usability testing is a technique used in user-centered interaction design to evaluate a product by testing it on users. This can be seen as an irreplaceable usability practice, since it gives direct input on how real users use the system.

## Usability Evaluation Procedure

### Users

A sample of real users is defined, about the size 25 to 30 people. Selected users are those who have background knowledge of searching and browsing of information over web.

### Overview of Product

An overview of the software product is provided to the users in order to understand the product.

### Tasks and Scenarios

Setting up a usability test involves carefully creating a scenario, where users perform a list of tasks using the product being tested.

### Performing Tasks

Users are observed while performing the tasks, to see where they encounter problems and experience confusion. If more people encounter similar problems, recommendations will be made to overcome these usability issues.

### Test Instrument

Several test instruments such as scripted instructions, paper prototypes, and pre- and post-test questionnaires are used to gather feedback on the product being tested.

**Evaluation**

Usability testing evaluates ease of use. Evaluation of the software product can be done by using questionnaire(s) as test instrument e.g., After-Scenario Questionnaire (ASQ), Questionnaire for User Interface Satisfaction (QUIS), Computer System Usability Questionnaire (CSUQ) and System Usability Scale (SUS). The chosen questionnaires are QUIS and SUS.

# CHAPTER # 06

# SOFTWARE IMPLEMENTATION DOCUMENT

# INTRODUCTION

This document describes the project implementation for developing the I-Cuboid Tool.

## Language Selection

The project is implemented in the following languages:

- **PHP**

  PHP is a general-purpose scripting language used for server side web development.

  Laravel Framework is used.

- **HTML and CSS**

  Used for designing of web pages.

- **Bootstrap4**

  Used for making web pages responsive.

- **JavaScript**

  Used for scripting and validation.

  Vue.js Framework is used.

## Tools Selection

- Visual Studio Code, v1.39.2
- Web Browser

## Resources

- **Flicker.photos.search API**
  Used for image retrieval from Flicker Search Engine.

# 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 easy to use.

## Description of User Interface

Despite the Home Screen, the remaining interface of I-Cuboid is divided into different panels. There are Query Formulation Panel, Grid View Panel, Browsing Panel, Navigation Panel and Image View Panel.

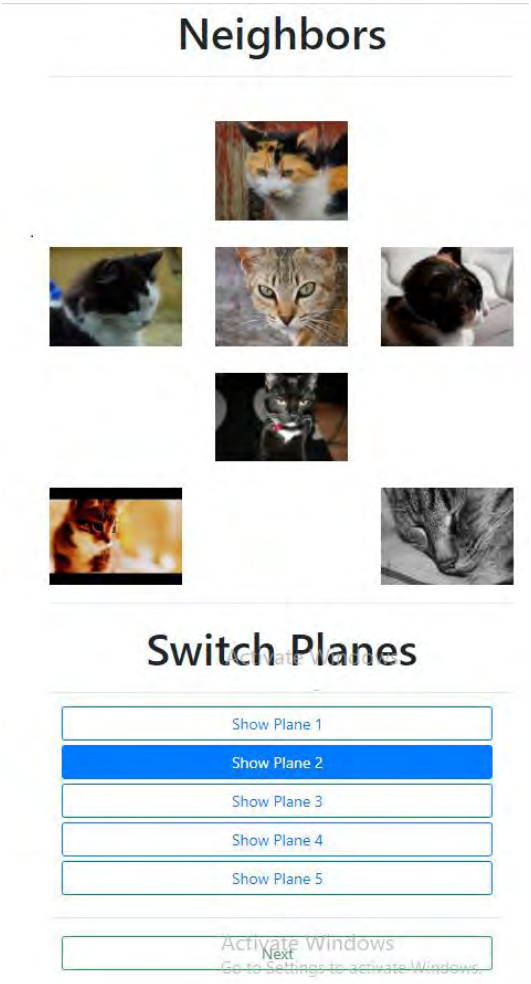
**Query Formulation Panel:** It contains logo (name) of the project and the search bar for querying images as shown below:


**Grid View Panel:** It contains images arranged and displayed in grid view shown in Figure 6.2.

**Browsing Panel:** It contains images arranged and displayed in Cube and enables in-depth browsing of images in cube as shown in Figure 6.3.

A switch option is provided which enable the user to switch from Grid View Panel to Browsing Panel and vice versa.

**Navigation Panel:** It contains buttons for the selection of planes in cube and display neighbor images of any image in cube as shown below.




All the navigation between planes of cube is done via navigation panel. Navigation panel lies within Browsing panel.

**Image View Panel:** It displays detailed view of any image whether selected from grid or from cube and also displays information about that image like image owner name, number of likes to that image, number of comments, number of downloads, date and time posted and the image format. This panel also provides the facility to download that image by providing download image option. When an image in opened, then Image View Panel appears as image slider.

Query Formulation panel and Image View panel are common to both Grid View and Browsing panels.

## Application Screenshots

Here are the screenshots of the application:

Home page of the I-Cuboid tool is shown in Figure 6.1.

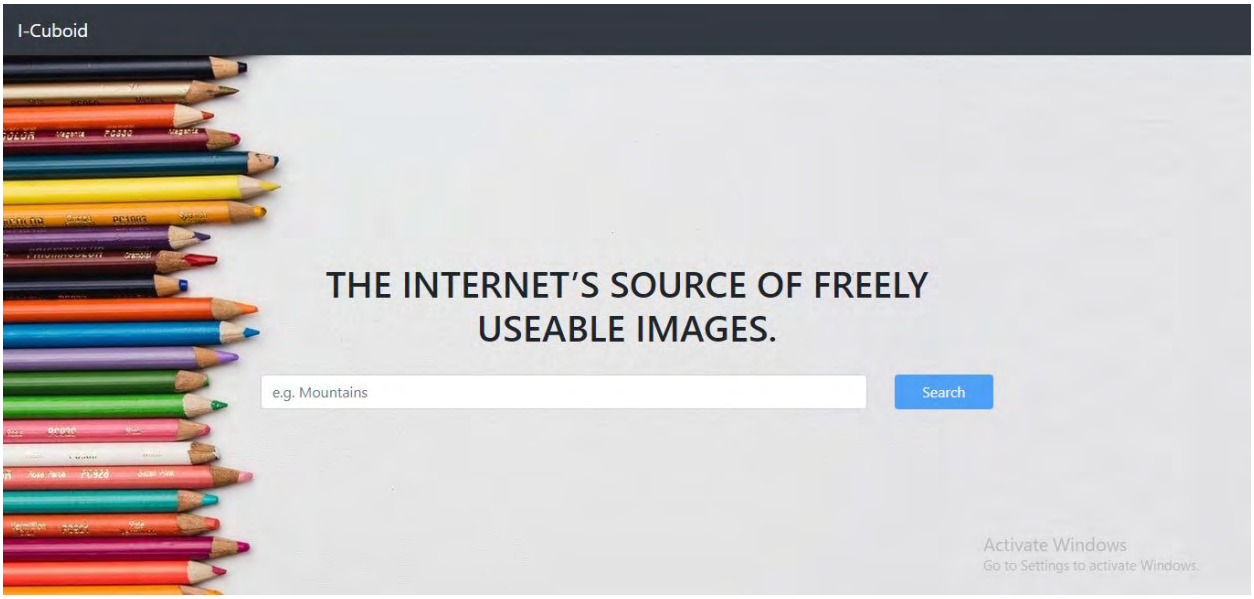



Figure 6.1 Home Page

Upon entering a query suppose “cats”, system is directed to grid view panel shown below in Figure 6.2.

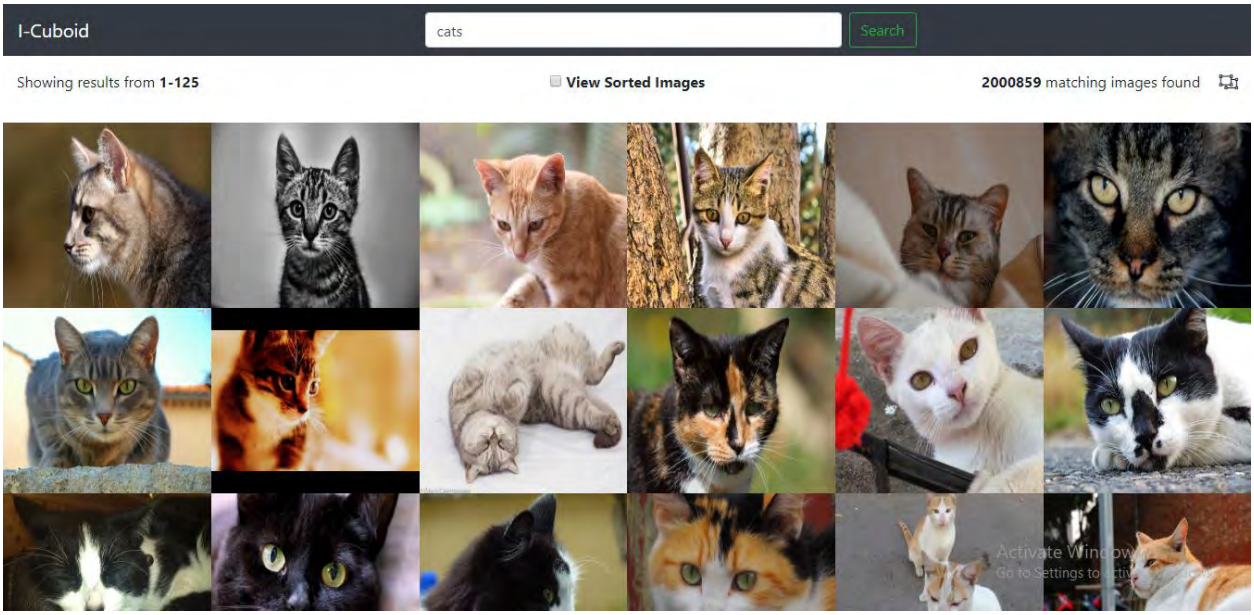



Figure 6.2 Grid View Panel

After switching from grid view panel, browsing panel appears along with navigation panel shown in Figure 6.3.

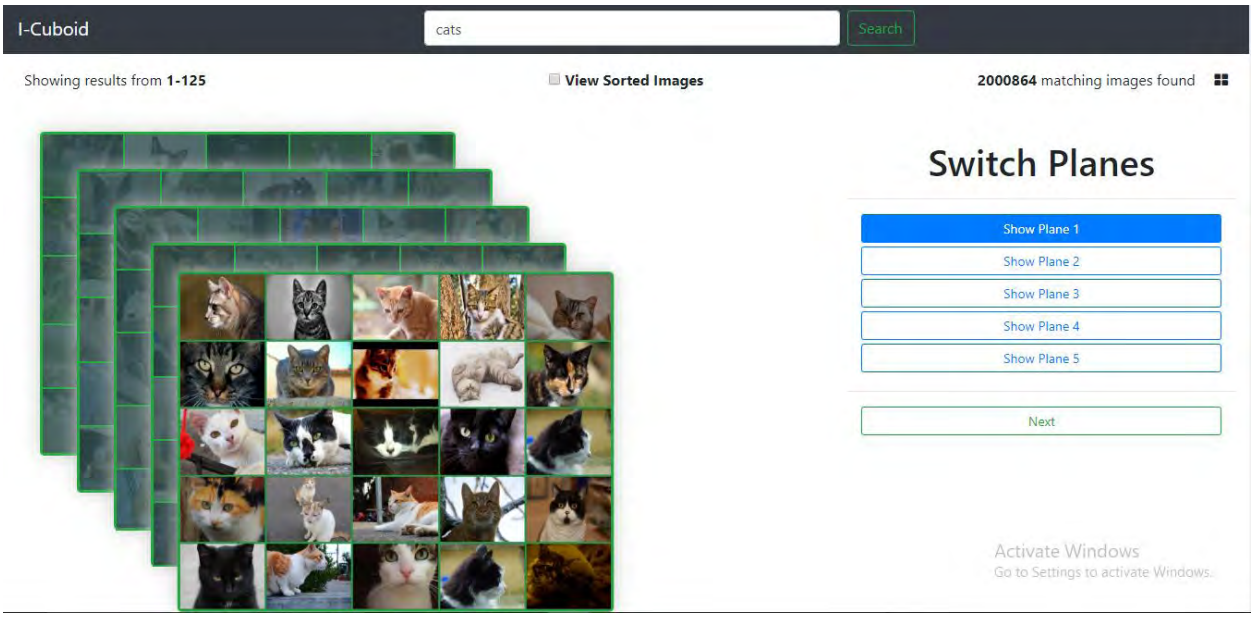



Figure 6 .3 Browsing Panel along with Navigation Panel

Opening any image from Grid view panel or Browsing panel leads to Image View Panel shown in Figures 6.4 and 6.5.

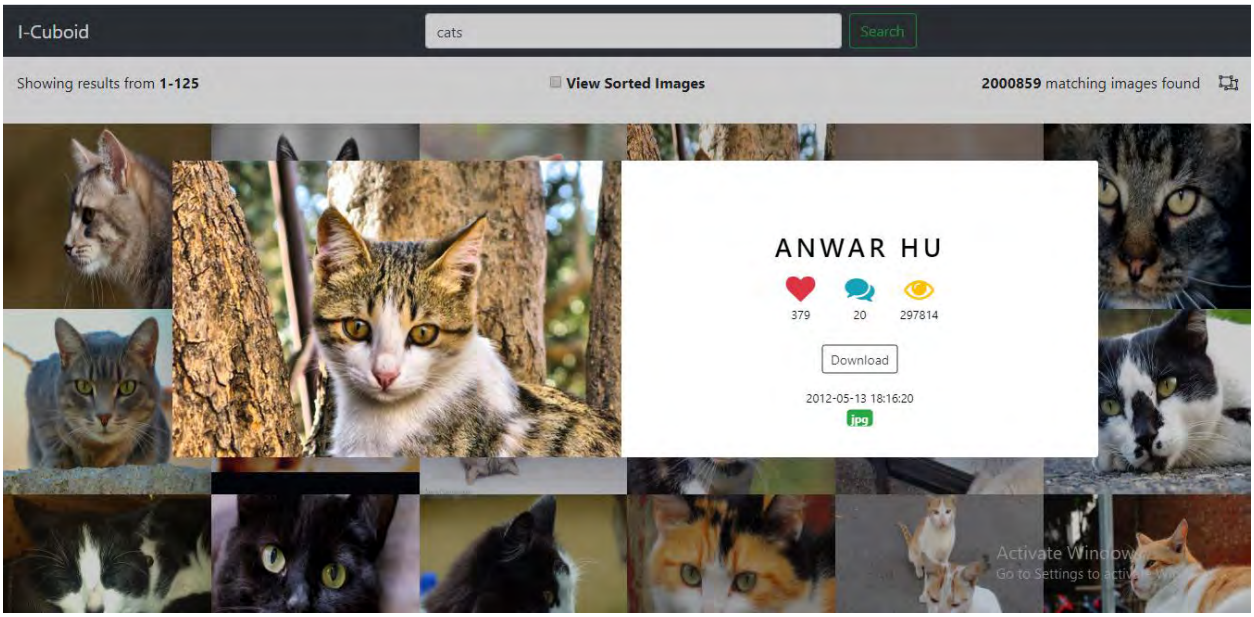



Figure 6 .4 Image View Panel in Grid

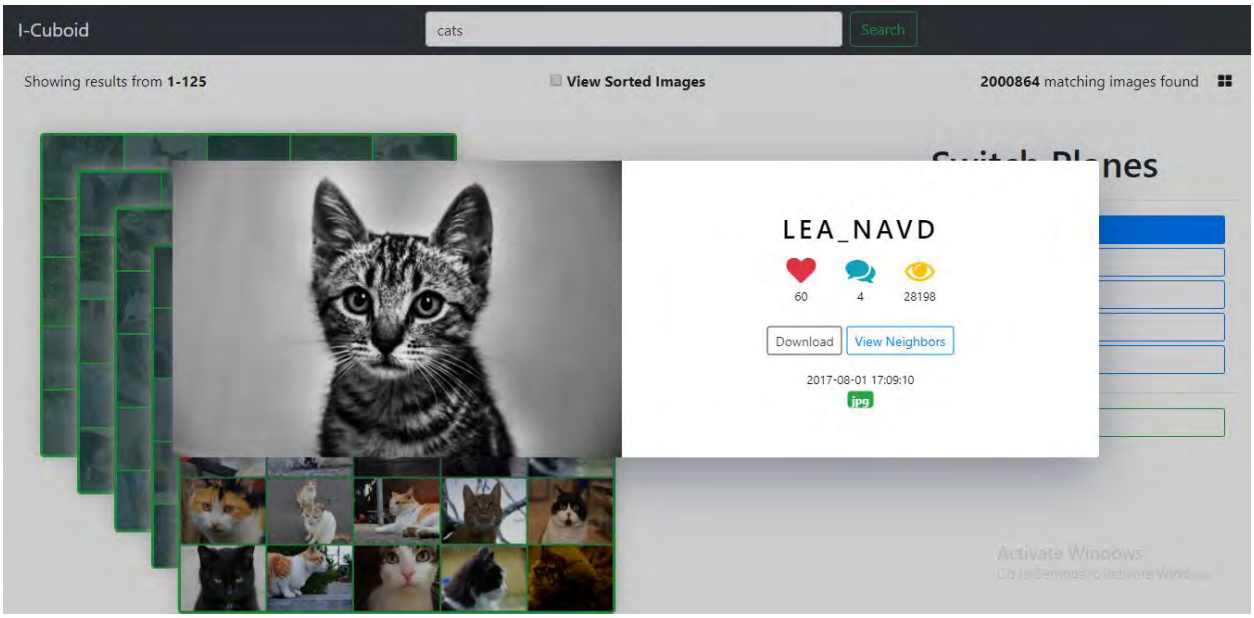



Figure 6 .5 Image View Panel in Cube

Case when user enters any invalid query, no images found error message appears, shown below in Figure 6.5.

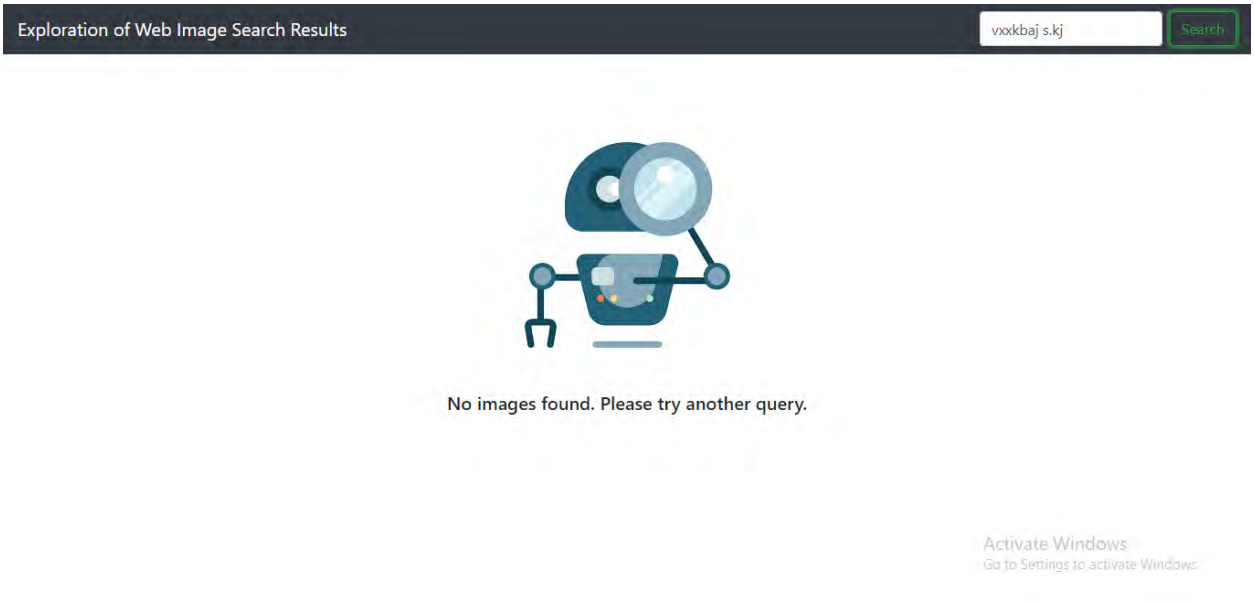



Figure 6.6 for Error Message (Invalid Query)

# CHAPTER # 07

# PROJECT EVALUATION

# INTRODUCTION

This chapter describes evaluation of I-Cuboid tool. It includes evaluation methodology, apparatus, instruments, procedure, results, discussion and conclusion.

# TYPES OF EVALUATION

Two types of evaluation has been done.

1. Usability Evaluation
2. Comparison

### Usability Evaluation

Usability testing is a technique used in user-centered interaction design to evaluate a product by testing it on users. It describes how easy it is for the user to use the software.

### Comparison

It describes comparison of I-Cuboid tool with Flicker search engine.

# EVALUATION METHODOLOGY

## Respondents

Respondents are those who participated in project evaluation. Total eighteen respondents took part in evaluation. They are categorized on the basis of gender, age and occupation.

### Gender

On the basis of gender, ten respondents were male and eight were females.

| **Males** | **Females** |
|---|---|
| 10 | 8 |

### Age

Minimum age of participant was twenty one and maximum age was forty five.

| **Minimum Age (years)** | **Maximum Age (years)** | **Mean Age (years)** |
|---|---|---|

| | | |
|---|---|---|
| 21 | 45 | 28 |

**Occupation**

Among eighteen participants, two were housewives, six were students and ten were employees working in different fields.

| **Housewives** | **Students** | **Working Body** |
|---|---|---|
| 2 | Under Graduate = 4<br>Post Graduate = 2 | Employees = 9<br>Faculty = 1 |

## Tasks

Four image exploration tasks were designed for respondents. Out of four tasks, two tasks were guided tasks where complete step by step guidance was provided to user in order to explore a particular image and two tasks were unguided where description of a particular image is provided to user and he/she had to explore that image by himself/herself. One pair of guided and unguided tasks was for the evaluation of grid view panel and the other pair of guided and unguided tasks was for the evaluation of browsing panel or cubic panel.

Tasks were performed by users by giving them predefined query terms. Four predefined query terms used for evaluation are:

1. Gold Fish in Aquarium
2. Bullet Trains in Japan
3. Fighter Aircraft in Pakistan
4. Indian Tiger

## Apparatus

I-Cuboid tool was installed on workstation equipped with Windows 10 Pro operating system, 10 GB RAM and 3 GHz dual core processor. Workstation was connected with wired mouse and keyboard. A screen recorded software was also installed on workstation to record users interaction with the tool.

## Instruments

There are many instruments available to evaluate usability e.g. Perceived Usefulness and Ease of Use (PUEU), Questionnaire for User Interface Satisfaction (QUIS), Computer System Usability Questionnaire (CSUQ), After Scenario Questionnaire (ASQ), System Usability Scale (SUS) etc.

The selected instruments for I-Cuboid tool are SUS and QUIS. Reason for selecting them is SUS provides overall system satisfaction while QUIS provides overall usability evaluation as well as

evaluation of other aspects for example terminology, system capabilities etc. Total number of questions in SUS are ten while QUIS has twenty seven questions.

**SUS Questionnaire**

| | The System Usability Scale<br>Standard Version | Strongly Disagree | 1 | 2 | 3 | 4 | 5 Strongly Agree |
|---|---|---|---|---|---|---|---|
| 1 | I think that I would like to use this system frequently. | | O | O | O | O | O |
| 2 | I found the system unnecessarily complex. | | O | O | O | O | O |
| 3 | I thought the system was easy to use. | | O | O | O | O | O |
| 4 | I think that I would need the support of a technical person to be able to use this system. | | O | O | O | O | O |
| 5 | I found the various functions in this system were well integrated. | | O | O | O | O | O |
| 6 | I thought there was too much inconsistency in this system. | | O | O | O | O | O |
| 7 | I would imagine that most people would learn to use this system very quickly. | | O | O | O | O | O |
| 8 | I found the system very awkward to use. | | O | O | O | O | O |
| 9 | I felt very confident using the system. | | O | O | O | O | O |
| 10 | I needed to learn a lot of things before I could get going with this system. | | O | O | O | O | O |

Figure 7. 1 SUS Questionnaire

**QUIS Questionnaire**

| OVERALL REACTION TO THE SOFTWARE | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | | NA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1. | terrible | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | wonderful | ○ |
| 2. | difficult | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | easy | ○ |
| 3. | frustrating | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | satisfying | ○ |
| 4. | inadequate power | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | adequate power | ○ |
| 5. | dull | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | stimulating | ○ |
| 6. | rigid | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | flexible | ○ |
| **SCREEN** | | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | | **NA** |
| 7. Reading characters on the screen | hard | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | easy | ○ |
| 8. Highlighting simplifies task | not at all | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | very much | ○ |
| 9. Organization of information | confusing | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | very clear | ○ |
| 10. Sequence of screens | confusing | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | very clear | ○ |
| **TERMINOLOGY AND SYSTEM INFORMATION** | | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | | **NA** |
| 11. Use of terms throughout system | inconsistent | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | consistent | ○ |
| 12. Terminology related to task | never | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | always | ○ |
| 13. Position of messages on screen | inconsistent | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | consistent | ○ |
| 14. Prompts for input | confusing | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | clear | ○ |
| 15. Computer informs about its progress | never | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | always | ○ |
| 16. Error messages | unhelpful | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | helpful | ○ |
| **LEARNING** | | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | | **NA** |
| 17. Learning to operate the system | difficult | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | easy | ○ |
| 18. Exploring new features by trial and error | difficult | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | easy | ○ |
| 19. Remembering names and use of commands | difficult | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | easy | ○ |
| 20. Performing tasks is straightforward | never | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | always | ○ |
| 21. Help messages on the screen | unhelpful | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | helpful | ○ |
| 22. Supplemental reference materials | confusing | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | clear | ○ |
| **SYSTEM CAPABILITIES** | | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | | **NA** |
| 23. System speed | too slow | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | fast enough | ○ |
| 24. System reliability | unreliable | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | reliable | ○ |
| 25. System tends to be | noisy | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | quiet | ○ |
| 26. Correcting your mistakes | difficult | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | easy | ○ |
| 27. Designed for all levels of users | never | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | always | ○ |
| | | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | | **NA** |

Figure 7. 2 QUIS Questionnaire

## Procedure

Evaluation procedure is given as:

1. Brief introduction of five to ten minutes in provided to respondents.
2. Respondents watched video simulation of I-Cuboid tool.
3. Respondents selected one of the four topic of their interest.
4. Respondents performed all the four tasks on the selected topic.
5. Questionnaires were filled by respondents.

## Results and Discussion

## Usability Evaluation

Usability evaluation was done with SUS and QUIS questionnaires.

## SUS Grid Results

SUS has total ten questions. SUS grid results are summed up as:

Table 7. 1 SUS Grid Results

| User/Q.No | Q1 | Q2 | Q3 | Q4 | Q5 | Q6 | Q7 | Q8 | Q9 | Q10 | Sum | Score Value |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 4 | 2 | 4 | 2 | 4 | 2 | 4 | 2 | 4 | 3 | 29 | 72.5 |
| 2 | 5 | 2 |  | 3 | 5 | 3 | 5 | 1 | 3 | 1 | 28 | 70 |
| 3 | 1 | 2 | 3 | 2 | 3 | 3 | 1 | 1 | 4 | 3 | 21 | 52.5 |
| 4 | 3 | 3 | 3 | 2 | 3 | 2 | 2 | 2 | 4 | 2 | 24 | 60 |
| 5 | 3 | 2 | 4 | 1 | 2 | 2 | 4 | 2 | 5 | 1 | 30 | 75 |
| 6 | 4 | 1 | 4 | 3 | 5 | 1 | 5 | 1 | 5 | 4 | 33 | 82.5 |
| 7 | 5 | 1 | 5 | 1 | 5 | 1 | 5 | 1 | 5 | 1 | 40 | 100 |
| 8 | 5 | 1 | 5 | 1 | 4 | 1 | 5 | 1 | 4 | 1 | 38 | 95 |
| 9 | 4 | 1 | 5 | 3 | 4 | 2 | 5 | 1 | 5 | 2 | 34 | 85 |
| 10 | 4 | 2 | 5 | 1 | 4 | 3 | 4 | 2 | 5 | 1 | 33 | 82.5 |
| 11 | 4 | 2 | 1 | 2 | 2 | 2 | 1 | 2 | 4 | 2 | 22 | 55 |
| 12 | 5 | 4 | 5 | 5 | 4 | 3 | 3 | 1 | 5 | 1 | 28 | 70 |
| 13 | 4 | 1 | 5 | 2 | 5 | 2 | 5 | 1 | 4 | 1 | 36 | 90 |
| 14 | 4 | 1 | 5 | 1 | 5 | 1 | 5 | 1 | 5 | 1 | 39 | 97.5 |
| 15 | 4 | 3 | 4 | 1 | 4 | 1 | 3 | 2 | 4 | 2 | 30 | 75 |
| 16 | 5 | 1 | 5 | 1 | 5 | 1 | 5 | 1 | 5 | 1 | 40 | 100 |
| 17 | 5 | 1 | 5 | 1 | 5 | 1 | 5 | 1 | 5 | 2 | 39 | 97.5 |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 18 | 4 | 1 | 5 | 1 | 5 | 2 | 5 | 1 | 5 | 1 | 38 | 95 |
| Average Score | | | | | | | | | | | | 80.8 |

SUS score for grid part comes out to be 80.8. The figure below shows that the score is in acceptable range, near to excellent.

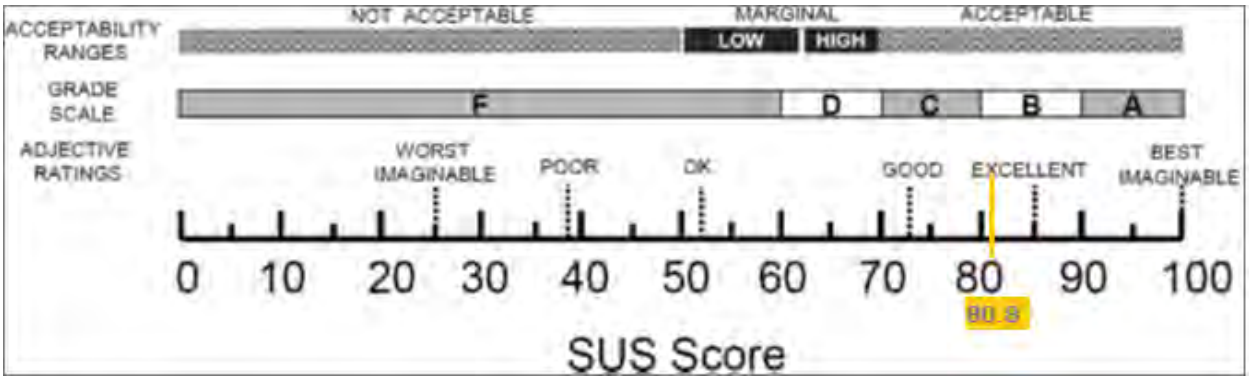



Figure 7. 3 SUS Score for Grid

**SUS Cube Results**

SUS cube results are summed up as:

Table 7. 2 SUS Cube Results

| User/Q.No | Q1 | Q2 | Q3 | Q4 | Q5 | Q6 | Q7 | Q8 | Q9 | Q10 | Sum | Score Value |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 4 | 2 | 4 | 2 | 4 | 2 | 4 | 2 | 4 | 3 | 29 | 72.5 |
| 2 | 4 | 3 | 5 | 4 | 4 | 3 | 4 | 2 | 3 | 3 | 25 | 62.5 |
| 3 | 5 | 1 | 1 | 5 | 5 | 1 | 5 | 2 | 5 | 3 | 29 | 72.5 |
| 4 | 3 | 2 | 4 | 2 | 4 | 2 | 4 | 2 | 4 | 2 | 29 | 72.5 |
| 5 | 5 | 1 | 5 | 1 | 5 | 1 | 5 | 1 | 5 | 1 | 40 | 100 |
| 6 | 2 | 4 | 2 | 1 | 2 | 4 | 3 | 2 | 4 | 4 | 18 | 45 |
| 7 | 5 | 1 | 5 | 1 | 5 | 1 | 5 | 1 | 5 | 1 | 40 | 100 |
| 8 | 5 | 1 | 5 | 1 | 5 | 1 | 4 | 1 | 4 | 1 | 38 | 95 |
| 9 | 4 | 5 | 3 | 4 | 3 | 4 | 5 | 2 | 5 | 2 | 23 | 57.5 |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 10 | 5 | | 5 | 2 | 4 | 3 | 4 | 3 | 5 | 2 | 33 | 82.5 |
| 11 | 5 | 1 | 5 | 3 | 5 | 1 | 4 | 1 | 5 | 2 | 36 | 90 |
| 12 | 5 | 4 | 4 | 4 | 5 | 3 | 3 | 3 | 5 | 2 | 26 | 65 |
| 13 | 4 | 3 | 4 | 3 | 2 | 2 | 4 | 4 | 3 | 4 | 21 | 52.5 |
| 14 | 3 | 2 | 3 | 1 | 4 | 1 | 5 | 2 | 4 | 2 | 31 | 77.5 |
| 15 | 4 | 2 | 4 | 2 | 3 | 2 | 3 | 2 | 4 | 2 | 28 | 70 |
| 16 | 5 | 1 | 5 | 1 | 5 | 1 | 5 | 1 | 5 | 1 | 40 | 100 |
| 17 | 5 | 1 | 5 | 2 | 5 | 1 | 5 | 1 | 5 | 1 | 39 | 97.5 |
| 18 | 4 | 1 | 5 | 1 | 5 | 2 | 5 | 1 | 5 | 1 | 38 | 95 |
| Avg Score | | | | | | | | | | | | 78.2 |

SUS score for cube part comes out to be 78.2. The figure below shows that the score is in acceptable range, in between good and excellent.

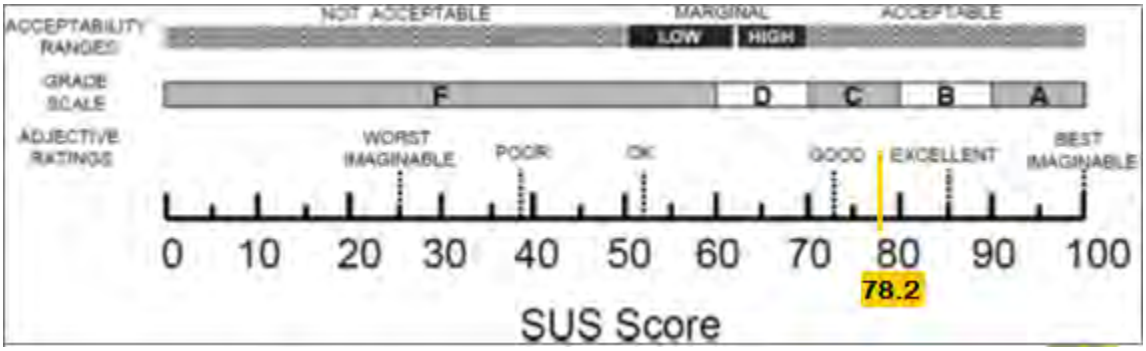



Figure 7. 4 SUS Score for Cube

**QUIS Grid Results**

QUIS has total twenty seven questions and it evaluates five different aspects of a software system which are overall reaction to the software, screen, terminology and system information, learning and system capabilities. QUIS grid results are summed up as:

Table 7. 3 QUIS Grid Results

| User/Q.No | Overall | Screen | Sys. Info. & Ter. | Learning | Sys. Cap. | Average | Score Value |
|---|---|---|---|---|---|---|---|
| 1 | 8 | 8 | 8 | 7.67 | 8 | 7.93 | 88.11 |
| 2 | 8.4 | 8.3 | 7 | 8.2 | 6 | 7.58 | 84.22 |
| 3 | 5.5 | 3.8 | 4.83 | 4 | 4.4 | 4.51 | 50.11 |
| 4 | 6.67 | 6.3 | 6.67 | 6 | 4.2 | 5.97 | 66.33 |
| 5 | 6.83 | 5.5 | 6.17 | 7.67 | 7.75 | 6.78 | 75.33 |
| 6 | 9 | 8.8 | 9 | 9 | 9 | 8.96 | 99.56 |
| 7 | 9 | 9 | 9 | 9 | 9 | 9 | 100 |
| 8 | 8.33 | 9 | 7.33 | 8.83 | 9 | 8.5 | 94.44 |
| 9 | 8.67 | 8 | 5.67 | 8 | 8 | 7.67 | 85.22 |
| 10 | 8.33 | 8.3 | 7.83 | 8.67 | 8.8 | 8.39 | 93.22 |
| 11 | 1.2 | 2.8 | 2.33 | 3.33 | 1 | 2.13 | 23.67 |
| 12 | 7.67 | 6.3 | 8 | 8.5 | 6 | 7.29 | 81 |
| 13 | 8.67 | 8 | 8 | 8.5 | 8.25 | 8.28 | 92 |
| 14 | 6.75 | 9 | 9 | 9 | 9 | 8.55 | 95 |
| 15 | 7 | 7 | 7.5 | 7.33 | 7.4 | 7.25 | 80.56 |
| 16 | 9 | 9 | 8 | 8 | 8 | 8.4 | 93.33 |
| 17 | 8.8 | 8.5 | 8.83 | 8.83 | 8.8 | 8.75 | 97.22 |
| 18 | 8.5 | 8.8 | 7.75 | 9 | 8.2 | 8.45 | 93.89 |
| Average | 7.57 | 7.47 | 7.27 | 7.75 | 7.27 | | |
| Score Value | 84.11 | 83 | 80.78 | 86.11 | 80.78 | | |

QUIS score values for all the five aspects for grid are given below:

Overall System = 84.11%

Screens = 83%

System Information and Terminology = 80.78%

Learning = 86.11%

System Capability = 78.2%

**QUIS Cube Results**

QUIS cube results are summed up as:

Table 7. 4 QUIS Cube Results

| User/Q.No | Overall | Screen | Sys. Info. & Ter. | Learning | Sys. Cap. | Average | Score Value |
|---|---|---|---|---|---|---|---|
| 1 | 6.83 | 7.25 | 7 | 7.5 | 8 | 7.32 | 81.33 |
| 2 | 7.4 | 8 | 6.33 | 6.75 | 5.4 | 6.78 | 75.33 |
| 3 | 5.5 | 7.25 | 8.33 | 8 | 7.6 | 7.34 | 81.56 |
| 4 | 7.17 | 7.5 | 7.6 | 6.75 | 6 | 7 | 77.78 |
| 5 | 8.5 | 8.5 | 8.2 | 9 | 9 | 8.64 | 96 |
| 6 | 1.33 | 2.25 | 2.33 | 2 | 5.8 | 2.74 | 30.44 |
| 7 | 9 | 9 | 9 | 9 | 9 | 9 | 100 |
| 8 | 8.67 | 9 | 7.6 | 9 | 9 | 8.65 | 96.11 |
| 9 | 6 | 7.67 | 8.5 | 4 | 2 | 5.63 | 62.56 |
| 10 | 8.33 | 8.25 | 8.17 | 8.5 | 8.4 | 8.33 | 92.56 |
| 11 | 9 | 9 | 9 | 9 | 9 | 9 | 100 |
| 12 | 7.33 | 7.25 | 7.67 | 8 | 6.75 | 7.4 | 82.22 |
| 13 | 6.67 | 5 | 6 | 5.17 | 7 | 5.97 | 66.33 |
| 14 | 7 | 6.67 | 9 | 8.75 | 9 | 8.08 | 89.78 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 15 | 7.67 | 7.25 | 7.33 | 6.67 | 7.2 | 7.22 | 80.22 |
| 16 | 9 | 9 | 8 | 8 | 8 | 8.4 | 93.33 |
| 17 | 8.8 | 9 | 8.83 | 8.67 | 8.8 | 8.82 | 98 |
| 18 | 8.5 | 8.75 | 8 | 9 | 8.2 | 8.49 | 94.33 |
| Average | 7.37 | 7.59 | 7.61 | 7.43 | 7.45 | | |
| Score Value | 81.89 | 84.33 | 84.56 | 82.56 | 82.78 | | |

QUIS score values for all the five aspects for cube are given below:

Overall System = 81.89%

Screens = 84.33%

System Information and Terminology = 84.56%

Learning = 82.56%

System Capability = 82.78%

**Comparison**

**SUS Results**

As SUS results are computed for both Grid and Cube, SUS score for grid is 80.8 while SUS score for cube is 78.2.

After comparing SUS results of both the presentations, it is concluded that grid is preferred more by users as it scores higher than cube.

The reason why grid is preferred more by users is because users mostly use grid presentation for image searching and browsing so they are familiar with it.

**QUIS Results**

QUIS results are compared as:

Table 7. 5 QUIS Results Comparison

| | **Overall** | **Screen** | **System Info. & Terminology** | **Learning** | **System Capabilities** |
|---|---|---|---|---|---|
| **Grid Score Value** | 84.11 | 83 | 80.78 | 86.11 | 80.78 |
| **Cube Score Value** | 81.89 | 84.33 | 84.56 | 82.56 | 82.78 |
| **Comparison** | Grid score higher than cube | Cube score higher than grid | Cube score higher than grid | Grid score higher than cube | Cube score higher than grid |

Overall cube score higher than grid in three aspects while grid score higher than cube in two aspects.

# CHAPTER # 08

# CONCLUSION AND FUTURE ENHANCEMENTS

# INTRODUCTION

This document describes the project conclusions and future enhancements – what type of new features can be added with time.

## Summary

This project allows user to search and explore image results in an interactive way. This tool allows user to browse images horizontally, vertically and in-depth.

## Conclusions

- ✓ We are now able to search image results based on query formulated by simple keywords.
- ✓ We are able to browse retrieved image results arranged in grid and in cube.
- ✓ We are able to browse retrieved, ranked image results arranged in grid and in cube.
- ✓ We are able to see detailed-view of any image arranged in cube.
- ✓ We are able to download any image from grid or cube.
- ✓ We are able to browse image results in-depth, arranged in cube.

## Future Enhancements

In future software product can be enhanced as:

- Instead of calling Image Search API of any search engine for image retrieval, this tool can be made as full-fledge search engine with its own web crawler that can search documents over web.
- We have implemented a fixed sized cube of 5x5x5. Instead, a cube of dynamic size can be implemented and its size will be determined by the number of images returned or number of images after ranking.
- In this project, images ranking is done by using textual features of images (textual meta data). Images can be ranked by using visual features of images for example Image colors, texture etc.
- Classification of images can be done in cube where same kind of images can be placed in one plane. For example, for query “cats”, white cats can be placed in one plane, black cats in another plane, brown cats in some other plane and so on.

# REFERENCES


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

[2] Guidelines for Extend Relationship

http://sce.uhcl.edu/helm/rationalunifiedprocess/process/modguide/md_exrel.htm

[3] The Include and Extend Relationships in Use Case Model

http://www.karonaconsulting.com/downloads/usecases_includesandextends.pdf

[4] https://www.wordstream.com/seo-keyword, last accessed 29, 06, 2019

[5] https://www.techopedia.com/definition/12708/search-engine-world-wide-web, last accessed 29,6,2019

[6] https://en.wikipedia.org/wiki/Web_search_query, last accessed 29, 06, 2019

[7] https://www.researchgate.net/post/Does_a_non-text_based_search_engine_exist, last accessed 29, 06, 2019

[8] https://www.igi-global.com/dictionary/ant-colony-optimization-use-content/13836, last accessed 29, 06, 2019

[9] https://en.wikipedia.org/wiki/Ranking, last accessed 29, 06, 2019

[10] https://medium.com/@perrysetgo/what-exactly-is-an-api-69f36968a41f, last accessed 29, 06, 2019

[11] https://en.wikipedia.org/wiki/Metadata, last accessed 29, 06, 2019

[12] Hyper Text Markup Language, HTML5 Toturial

https://www.w3schools.com/html/, last accessed 20, 01, 2020

[13] Cascading Style Sheets

https://en.wikipedia.org/wiki/Cascading_Style_Sheets, last accessed 20, 01, 2020

[14] Bootstrap4 Introduction, Bootstrap4 Tutorial

https://www.w3schools.com/bootstrap4/, last accessed 20, 01, 2020

https://getbootstrap.com/, last accessed 20, 01, 2020

[15] HyperText Preprocessor

https://www.javatpoint.com/php-full-form, last accessed 30,6,2019

[16] Information Retrieval and Ranking

https://en.wikipedia.org/wiki/Ranking_(information_retrieval), last accessed 30, 06, 2019

[17] Usability Testing

https://en.wikipedia.org/wiki/Usability_testing, last accessed 30, 06, 2019

[18] Flicker Image Search API (flicker.photos.search)

https://www.flickr.com/services/api/flickr.photos.search.html, last accessed 20, 01, 2020

[19] Visual Studio Code

https://code.visualstudio.com/, last accessed 20, 01, 2020

[20] Sequence Diagram If-Else Condition/Guard Condition

https://www.researchgate.net/figure/UML-Sequence-diagram-of-nested-if-else-Fig-6-represents-the-nested-if-else-message_fig6_307902652, last accessed 21, 01, 2020

https://circle.visual-paradigm.com/selection-loops-combination/, last accessed 21, 01, 2020

[21] Jaccard Index

https://en.wikipedia.org/wiki/Jaccard_index