**MAKERERE UNIVERSITY**

COLLEGE OF COMPUTING AND INFORMATION SCIENCES

DEPARTMENT OF NETWORKS

BACHELOR OF SCIENCE IN SOFTWARE ENGINEERING (YEAR 2)

RECESS TERM 2 (BSE 2301)

SOFTWARE REQUIREMENTS SPECIFICATION DOCUMENT FOR:

**MULTI SALIENT OBJECT ANALYSIS PROJECT**

**PROJECT MEMBERS (GROUP 12)**

|  |  |  |  |
| --- | --- | --- | --- |
| **NAME** | **REG No.** | **STD No.** | **SIGNATURE** |
| ISABIRYE TAIBU | 16/U/20495 | 216022278 |  |
| NABUUFU ERETH | 16/U/8227/EVE | 216009341 |  |
| NAKIRANDA PROSCOVIA | 16/U/8785/PS | 216002865 |  |
| KANDOLE LAILA | 12/U/6171/PS | 212011932 |  |

**Introduction.**

**Purpose.**

The purpose of this Software Requirements Specification (SRS) document is to provide a detailed description of the functionalities of the proposed Multi Salient Objects Analysis system. This document will cover each of the system’s intended features, as well as offer a preliminary glimpse of the system’s user interface and system constraints. It will cover hardware, software, and various other technical dependencies and how the model will react to external stimuli

The document is intended to describe accurately the capabilities that the proposed system should provide to its end-users and also to specify all the non-functional requirements that the application should implement, regarding subjects like: performance, availability, reliability,usability,maintainability, security, etc.

**Document Conventions.**

This document has 1.5 spacing and all the text is justified. The body is 12point Times New Roman font. Major headers are bold and 18points Times New Roman font. Sub headers use bolded 12point Times New Roman font.

This document features some terminology and abbreviations which readers may be unfamiliar with. See Appendix A (Glossary) for a list of these terms and their definitions.

**Intended Audience and Reading Suggestions.**

This Software Requirements document is intended for:

* Developers of the system will use this document to get the details of the intended project capabilities and more easily understand where their efforts should be targeted in implementing the system features.
* Project testers will use this document as a base for their testing strategy as some bugs are easier to find using a requirements document. This way testers will be able to do the validation and verification tests to make sure the developed system satisfies the user requirements.
* End users of this application who wish to read and get more details about what the developed system can do.
* Project supervisors will use this document to assess the quality of the implemented system

The document contains six main sections describing the project. For readers who may want to skip some parts of the document, below is the organization of the document.

Section 1: Introduction.

This section gives a brief introduction to the project. It contains subsections describing the purpose of the project, the document conventions and the intended audience for the document.

Section 2: Overall Description

This section gives a detailed description of the project. It contains the following subsections.

* Product perspective.
* Product functions.
* User classes and classifications.
* Operating environment.
* Design and implementation constraints.
* User documentation.
* Assumptions and dependencies.

**Section 3: External Interface requirements**

This gives details of the system interfaces. It contains the follow sections.

* User Interfaces.
* Hardware Interfaces.
* Communication Interfaces.

**Section 4: System features.**

This section gives a detailed description of the major system features.

**Section 5: Other nonfunctional requirements.**

This section discusses the nonfunctional requirements of the system including the following.

* Performance requirements.
* Safety requirements.
* Security requirements.

**Section 6: Other requirements.**

This describes requirements that are not discussed in any other section in the document.

**Product Scope.**

The Multi Salient Object detection system, is unified for all types of images including PNG, JPG and TIFF. The system will allow the user to input an image through a graphical user interface, the image them will be displayed in a browser window. The number of salient objects in the image/scene are then displayed to the user. The system in addition will allow the user to apply changes to the images e.g. cropping, rotating, flipping among other. These will not affect the original image.

**References:**

[1] K.F Keren Enhancement of Salient Image Region for Visual Object Detection. Chalmers

Reproservice, Goteborg, Sweden, November 2014.

[2] J Hunaizu, Y. Zejian and S Li, Salient Object Detection: A Discriminative Regional

Feature Integration Approach.,2011.

[3] A. Borgi, M.M. Cheng, Salient Object detection: A survey. In CVPR, 2017.

**Overall Description.**

**Product Perspective.**

The multi salient object detection system is a new self-contained product with no previous version. The product after implementation will be the first release.

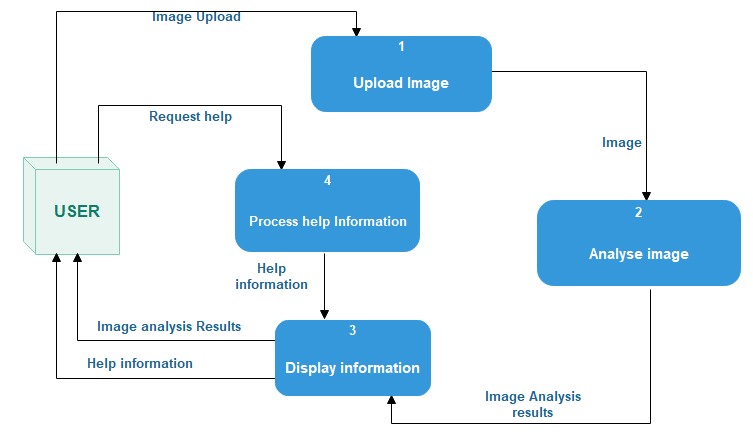
**Product Functions.**

* Users must be able to upload images
* Users must be able to see image analysis results
* Users must be able to easily get help on how to use the system.
* \* search for specific number of salient object\*

The data flow diagram below illustrates these functions.

**prediction of salient object**

MSO analysis system level 1 Data Flow Diagram.



**User Classes and Classifications.**

There are no user class constraints on the MSO system and its will be used by anyone who wants to analyze images for salient objects.

The users are not expected to have a high educational and proficiency level or technical expertise. Hence, the user interfaces in available in the most popular official and international language, English.

**Operating Environment.**

The main component of the Multi salient object detection system is a cross platform software system which will be not be limited to any single operating environment and it has no practical hardware constraints. It will be able to operate in any environment including windows, Mac OS, Linux and the family of Unix based systems where R is installed.

The system will rely on several functionalities built into the R Application Programming Interface (API), so ensuring appropriate usage of the required R packages will be a major concern.

Beyond that, the application is a self-contained unit and will not rely on any environment, software or hardware components.

**Design and Implementation Constraints.**

MSO system is platform independent and will be written in R. Its user interface will be written with R shiny package functions, so anyone who wishes to work on further development of the system has to know this programming language.

The memory requirements of Multi Salient Object detection system will be small since it is a light weight application.

MSO is meant to be quick and responsive, even when dealing with large number

and transactions, so each feature must be designed and implemented with efficiency in mind.

**User Documentation.**

Though, the system will be designed to be as simple to use as possible, users may still require some supplementary information about each component of the system. For this reason, the system will contain a feature that offer this. The Help menu will be implemented as part of the system so that users can get help readily when they encounter difficulties while using the system.

The Help menu is a collection of topics covering each of the application’s functionality and its features. At any time, the user can navigate to the Help menu and select any of these topics to obtain more information.

**Assumptions and Dependencies**

One assumption about the product is that it will always be used on platforms were R can be installed and that the system has enough performance. It is assumed that the users have working and well configured R environment version 1.6 or above is on their machine. This is because most of the features of the system depend on R environment packages and will work as intended only if these packages are installed.

It is also assumed that the users have a browser on their computer since MSO has a web based interface.

\* 80% of the dataset is used for training and 20% for testing the model\*

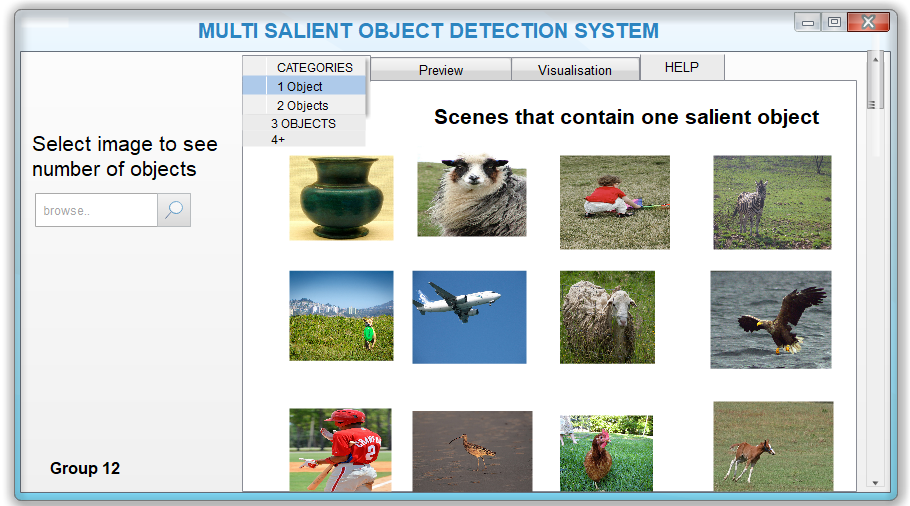
Beyond this, no other assumptions and dependencies are necessary to run the system.

**External Interface Requirements**

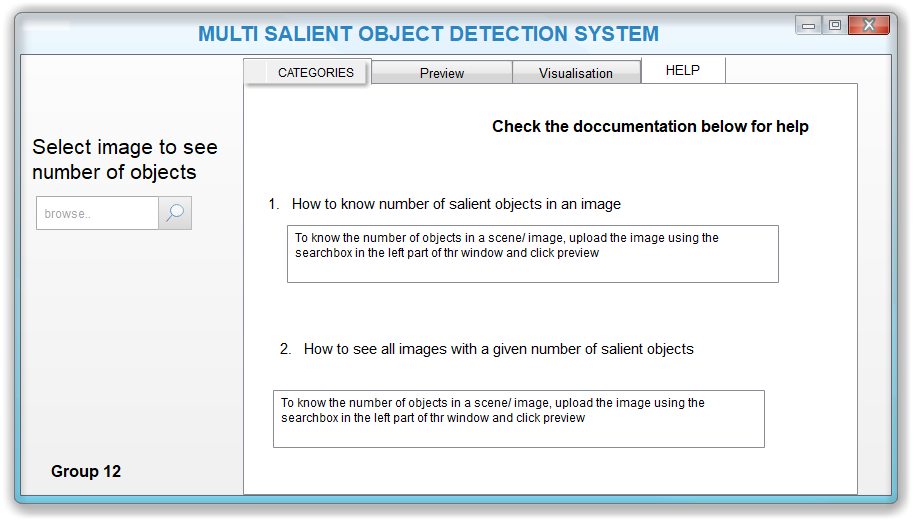
**User Interfaces.**

Below are some of the user interfaces in the multi salient object analysis system

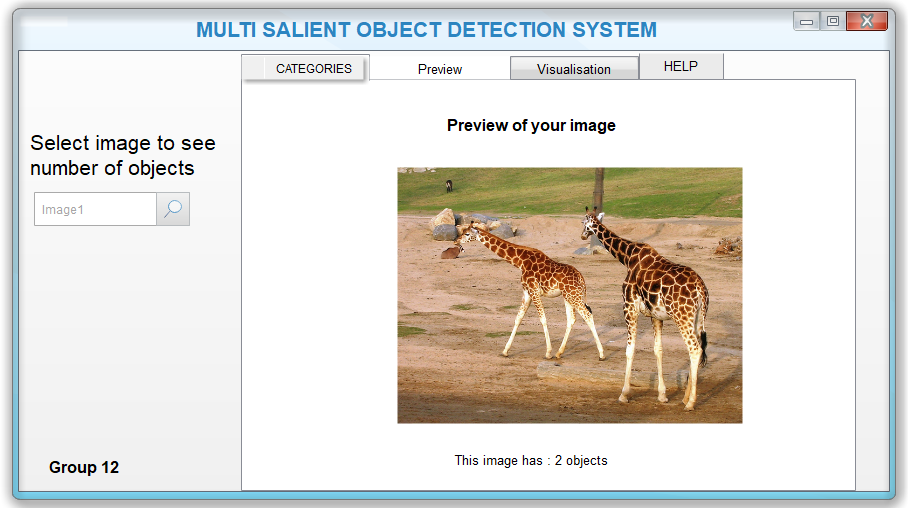
**Page containing scenes with one salient object**



**Help page containing system documentation**



**Page where user can preview a scene and see the number of salient objects it contains**



**Hardware Interfaces.**

MSO is intended as a web based application for any platform and hence is solely

supported on any hardware. Messages, updates, and data exchanged

are transmitted to and handled by the Shiny server where the system is hosted using the Http protocol.

Since the system does not have any designated hardware, it does not have any direct hardware interfaces. The hardware is managed by the underlying operating system on the user’s computer

**Software Interfaces.**

MSO system is compatible with every system supports a R and JPG, TIFF and PNG files. That means that it is independent from the operating system of the computer system on which it runs.

\* MSO uses the following libraries

shiny package used to develop the interfaces

tensorflowR and keras for training the neural networks

EBImage provide tools for working with images\*

**Communications Interfaces.**

The web based UI is the only means of communication between the user and the system. The system is accessible through all popular web browsers.

\*web browser .this will use Http to load the MSO system from Rstudio through the shiny package \*

**System Features**

**Upload Image(s)**

Description and priority

This feature is one of the core features of the system. It will enable the user of the MSO system to select and upload an image for analysis.

**Priority** This is a high priority requirement

**Response sequence**

User clicks on the search icon on the search box.

Device displays a file explorer window.

User browse, select the image and click ok.

**Functional requirements**

The system shall provide the option to submit image.

The system shall store the image in temporary memory.

The system shall verify image and display an error message if the input is not an image.

The system shall display the currently viewed image after verification.

Help

Description and priority

This feature is one of the core features of the system. It will enable the user of the MSO system to get help on how to use the other system features.

Priority This is a medium priority requirement

Response sequence

User clicks on the help tab.

Device displays help information.

User scrolls to the desired topic.

**Functional requirements**

The system shall provide the help menu tab.

Displays a list of topics covering the different components of the MSO system.

**Image Analysis**

Description and priority

This feature is the most core features of the system. It involves the system segmenting the image and analyzing each segment

Priority This is a high priority requirement

Response sequence

User submits image.

System breaks the image into segments.

System analyses images.

**Functional requirements**

The system shall mechanism of dividing image into segments.

System stores analysis results

**Display Analysis Results**

Description and priority

This feature will enable the user of the MSO system to get the results of the analysis of the input image

Priority This is high priority requirement

Response sequence

User clicks on the preview tab.

Device displays analysis results.

**Functional requirements**

The system shall retrieve analysis information.

Displays the information

**Other Nonfunctional Requirements**

**Performance Requirements.**

MSO is a light weight system that needs very few system resources in order to work. It is designed not to delay the system from other key processes and the response time of the program is direct.

Furthermore, the system will undergo various improvements and system updates will always be available. These updates will be more efficient in performance than the previous versions. The user is encouraged to always have the latest versions for the best performance experience.

\*the system must resolve salient object detection issues and handle concurrent use of the model in a 24/7 basis. It must be interactive and delays involved must be less or must be minimal

so in every detection process the delay is based on the number of salient objects available and so there is a probability that there, will be a delay of less than 20 seconds.

**Safety Requirements.**

To ensure quality image processing the system is designed in a way that it does not make any modifications e.g. compression to the input data. The system does not affect any other program running on the user’s computer

In case of error the user is encouraged to use documentation information in the help menu tab on how to properly use the system.

**Security Requirements.**

The system does not restrict any user and there is no user authentication required. This is because the system does not affect the quality and state of the dataset thus any user can use the system with no threat posed onto the state of the data and other applications on the computer.

**Software Quality Attriblutes.**

\* usability\* The MSO system provides a pleasant and user friendly graphical interface with relatively simple functions. Any user should be able to use the system without any specific knowledge or experience by reading the user documentation in the help menu. Users only need to provide some image inputs to the system and with just a few clicks they can perform any action.

\* availability .if some data is lost from the dataset I.e incase of deletion during input process, the data should be sent back for verification.

Flexibility .the system should be able to operate on different platforms(both hardware and software)

maintainability. The system will be easy to maintain by updating data and changing code to suit other visualisations not previously implemented.\*

**Other Requirements**

**Portability of service**

The Service shall be useable on any modern PC running any current, supported version of Windows, Mac OS.

**Appendix A: Glossary**

**MSO -** Multi salient object

**OS** - Operating system

**Http** - Hypertext transfer protocol

**UI** - User interface.

JPG-Joint Photographic Experts. A commonly used method of compression for digital images

TIFF-Tagged Image File Format. A computer file format for storing graphics images, popular among graphic artists, the publishing

PNG-Portable Network Graphics- A raster graphics file format that supports lossless data compression.

API- Application Programming Interface

R – Popular programming and data analysis language used to implement the system

**Appendix C: To Be Determined List**

Presently there are no remaining to be determined list. All TBDs have been tracked to their closure.