

MINI PROJECT REPORT
ON
Colour Extraction & Image Histogram

as a part of
Project Based Learning

Submitted by
Dixit Vedant Haresh
Hingrajiya Shyam Niteshbhai
Shinde Aniket Rajendra
Nagare Anurag Ramesh

Guided by
Prof. Mariyam Maniyar

Master of Computer Applications
(Under Faculty of Engineering)

Department of MCA
K. K. Wagh Institute of Engineering Education & Research
Hirabai Haridas Vidyanagari, Amrutdham, Panchavati,
Nashik – 422 003

**K. K. WAGH INSTITUTE OF ENGINEERING EDUCATION &
RESEARCH
NASHIK - 422003**

CERTIFICATE

This is to certify that

**Dixit Vedant Haresh
Hingrajiya Shyam Niteshbhai
Shinde Aniket Rajendra
Nagare Anurag Ramesh**

have successfully completed the Mini Project on

Colour Extraction and Image Histogram

as a part of

Project Based Learning

during academic year 2022 – 23

Prof. Mariyam Maniyar
Project Guide

Prof. V. C. Bagal
I/c Head,
Dept. of MCA

Dr. K. N. Nandurkar
Principal

MINI-PROJECT ABSTRACT

PROJECT ID – 03

PROJECT TEAM MEMBERS

- | | |
|-------------------------------|------|
| • Shinde Aniket Rajendra | B_53 |
| • Nagare Anurag Ramesh | B_39 |
| • Hingrajiya Shyam Niteshbhai | B_25 |
| • Dixit Vedant Haresh | A_18 |

PROJECT AIM / OBJECTIVE

To develop an application to perform colour code extraction (RGB code) from a live image capture system and then to further use it in our colour palette for various design and development aspects.

FEATURES & USEFULNESS

1. User-friendly
2. Helps identify colours immediately
3. Pixel data in the form of RGB code
4. Helps understand colour values and pixel identification
5. Generates a histogram based on pixel density

MARKETING POTENTIAL / TARGET AUDIENCE

1. Website Designers
2. Image Processing System users
3. Digital Content Creators, YouTubers, Photo Editors
4. Data Analysts in Image Processing

TABLE OF CONTENTS

<i>Serial No.</i>	<i>Title</i>	<i>Page No.</i>
1)	Introduction	6-7
2)	Analysis	8-10
3)	Design & Modeling	11-17
4)	Methodology	18
5)	Results	19-22
6)	Testing	23-25
7)	Conclusion	26
8)	Future Enhancements	27
9)	Bibliography	28
10)	Acknowledgment	29

LIST OF FIGURES

FIGURE NO. 1 - ER DIAGRAM

Page No. 12

Determines the relationship between the entities involved within and around the project.

FIGURE NO. 2 - ACTIVITY DIAGRAM

Page No. 13

Determines the sequence of activities in the project with respect to the entities.

FIGURE NO. 3 - SEQUENCE DIAGRAM

Page No. 14

Determines the sequence and flow of instructions carried out.

FIGURE NO. 4 - USE CASE DIAGRAM

Page No. 15

Determines the roles the actors involved in the working of project perform.

FIGURE NO. 5 - DEPLOYMENT DIAGRAM

Page No. 16

Determines the systems used in the project and their working.

FIGURE NO.6 - COMPONENT DIAGRAM

Page No. 17

Determines the working and functioning of various modules or components with respect to the entities.

INTRODUCTION

Image recognition is bringing revolutionary changes to the ways in which we consume and process information online. Deeply integrated into web pages and apps, it allows us to make sense of visual data in small and large quantities alike as we've never been able to do before.

The applications of image recognition are diverse and empowering. Colour extraction is one of the most significant and game-changing capabilities offered by computer vision. The possibility to identify and analyze the colours in images gives numerous possibilities to businesses to better use their visual libraries, monetize them, and even increase sales of in-store products.

How does colour extraction through image recognition work? The colour API enables analysis of visuals in terms of the colours they contain. It determines the five most prominent colours that are present in an image. Then they can be exported as hex code, RGB triple, specific colour name, and parent colour name. This makes them easy to use for, say, keyword tagging and categorization.

- *The front-end or client-side i.e. GUI is developed using Python language by importing it's TKinter library for UI Design.*
- *The back-end or server-side i.e. the processing and functionality is organized using Python language by importing OpenCV CV2, Numpy, Pillow libraries.*

An image histogram is a type of histogram that acts as a graphical representation of the tonal distribution in a digital image. It plots the number of pixels for each tonal value. By looking at the histogram for a specific image a viewer will be able to judge the entire tonal distribution at a glance.

Image histograms are present on many modern services. Photographers can use them as an aid to show the distribution of tones captured, and whether image detail has been lost to blown-out highlights or blacked-out shadows. This is less useful when using a raw image format, as the dynamic range of the displayed image may only be an approximation to that in the raw file

The horizontal axis of the graph represents the tonal variations, while the vertical axis represents the total number of pixels in that particular tone

The left side of the horizontal axis represents the dark areas, the middle represents mid-tone values and the right hand side represents light areas. The vertical axis represents the size of the area (total number of pixels) that is captured in each one of these zones.

Thus, the histogram for a very dark image will have most of its data points on the left side and center of the graph.

SYSTEM REQUIREMENTS

Software Requirements

- Python IDLE or any commonly used Python Editor like PyCharm, Jupyter, etc.
- OpenCV Libraries for Python Capture and Image Processing
- TKinter Libraries for Python UI
- Operating System – Windows, Linux

Hardware Requirements

- CPU – Dual Core or better
- RAM – 2GB or better

ANALYSIS

Analysis of a new system is very crucial to conduct before making it actual. To study the system, the analyst needs to collect facts and all relevant information. The facts when expressed in quantative forms are termed as data. The success of any projects depends upon the accuracy of the available data. Accurate information can be collected with the help of certain methods/techniques. It is to minimize the risk factor.

These fact finding methods/techniques include:

- *Observation*
 - *Interview*
 - *Questionnaires*
 - *Record Review*
-
- **OBSERVATION**

In this technique, the system analyst participates in the organization, studies the flow of documents, applies the existing system, and interacts with the users. Observation can be a useful technique when the system analyst has a user point of view. A sampling technique called work sampling is useful for observation. By using this technique, system analyst can know how employees spend their days.
 - **INTERVIEW**

An interview is the most commonly used technique to collect information from the face to face interviews. The purpose of the interview is to find, verify, clarify facts, motivate end-users involved, identify requirements, and gather ideas and opinions. The role of the interview includes interviewer who is a system analyst and interviewee who are a system owner or user. The interviewing technique needs good communication skills for interaction between system analysts and users.

There are two types of interviews:

- ✓ **UNSTRUCTURED INTERVIEWS**

An interview that is conducted with only a general goal or subject in mind and with few, if any, specific questions. Open-ended questions type is used in an unstructured interview that allows the user to answer freely in an appropriate way.

- ✓ **STRUCTURED INTERVIEWS**

A structured interview is an interview that contains a predefined set of questions. In a structured interview, close-ended questions type is used to limit answers to specific choices, short and direct responses from the interviewees.

- **QUESTIONNAIRES**

Questionnaires are also one of the useful fact-finding techniques to collect information from a large number of users. Users fill up the questions which are given by the system analyst and then give the answers back to the system analyst. Questionnaires can save time because the system analyst does not need to interview each of the users and if the time of the interview is short, questionnaires are more useful. To fulfill the requirements of the system objective, a system analyst should have the ability to clearly define the design and frame of questionnaires.

There are two types of questionnaires:

- ✓ **FREE-FORMAT QUESTIONNAIRES**

In free format questionnaires, users are allowed to answer questions freely without an immediate response. The results are also useful in learning about the feelings, opinions, and experiences of the respondents.

✓ **FIXED-FORMAT QUESTIONNAIRES**

The purpose of fixed-format questionnaires is to gather information from the predefined format of questions. Users are allowed to choose the result from the given answers. There are three types of fixed-format questions: multiple-choice questions (Yes or No type), rating questions (Strongly Agree, Agree, No opinion, Disagree, Strongly disagree), ranking questions.

▪ **RECORD REVIEW**

The information related to the system is published in the sources like newspapers, magazines, journals, documents etc. This record review helps the analyst to get valuable information about the system and the organization.

DESIGN & MODELING

UML, short for Unified Modeling Language, is a standardized modeling language consisting of an integrated set of diagrams, developed to help system and software developers for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML is a very important part of developing object-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software. In this article, we will give you detailed ideas about what is UML, the history of UML and a description of each UML diagram type, along with UML examples.

An Entity-Relationship Diagram and UML Diagrams relevant to the project are given as follows:

FIGURE NO. 1 - ENTITY-RELATIONSHIP DIAGRAM

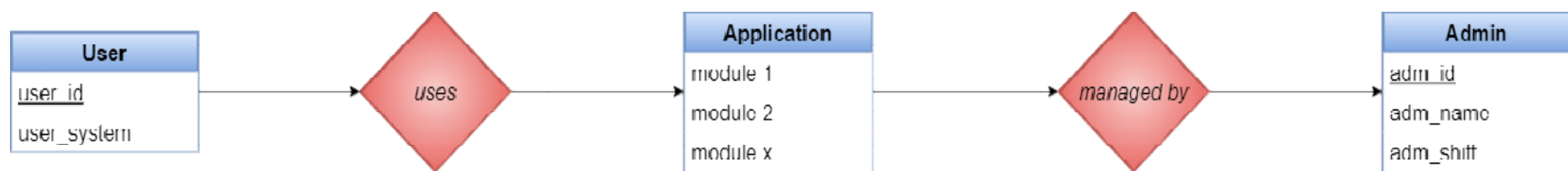


FIGURE NO. 2 - ACTIVITY DIAGRAM

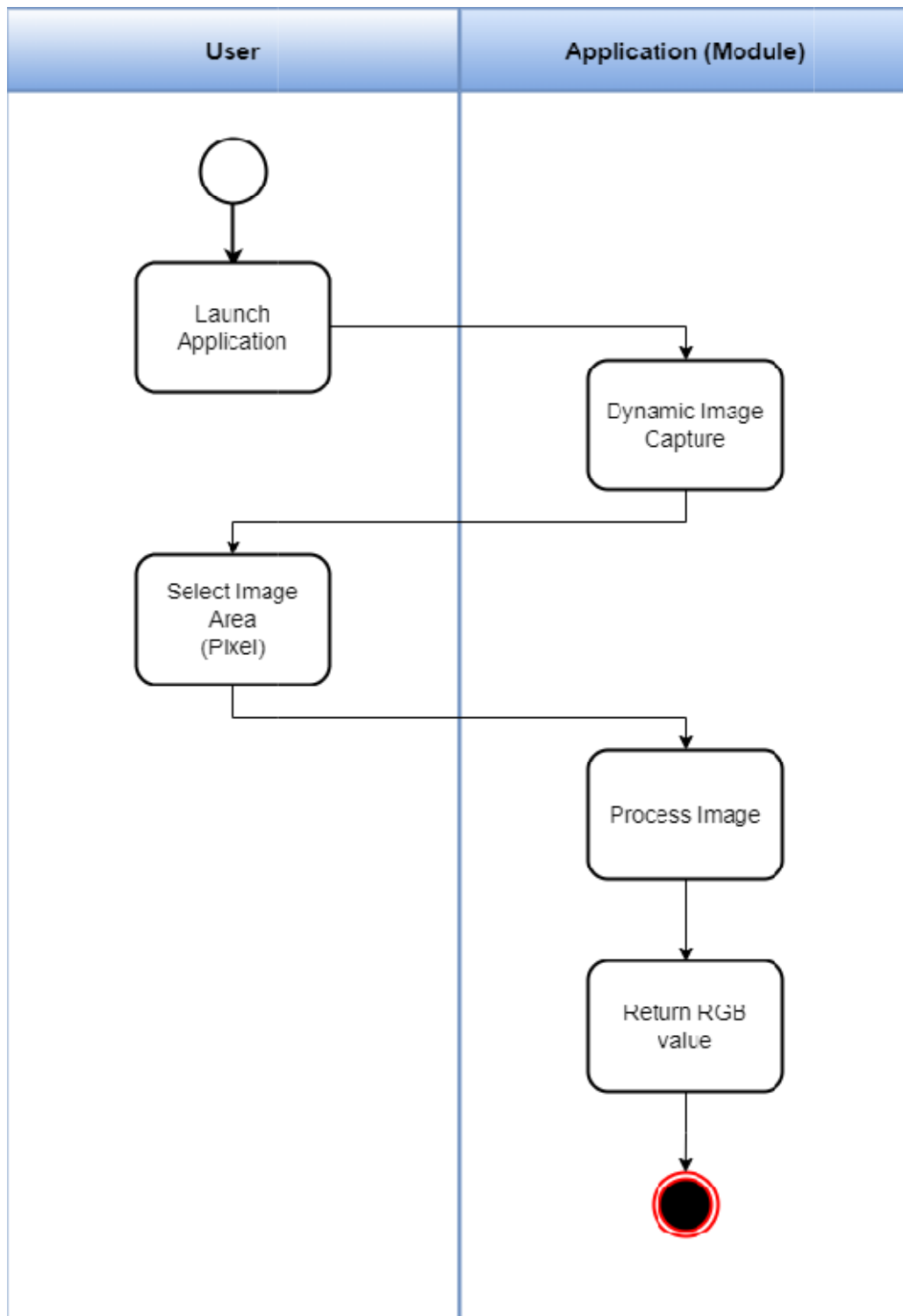


FIGURE NO. 3 - SEQUENCE DIAGRAM

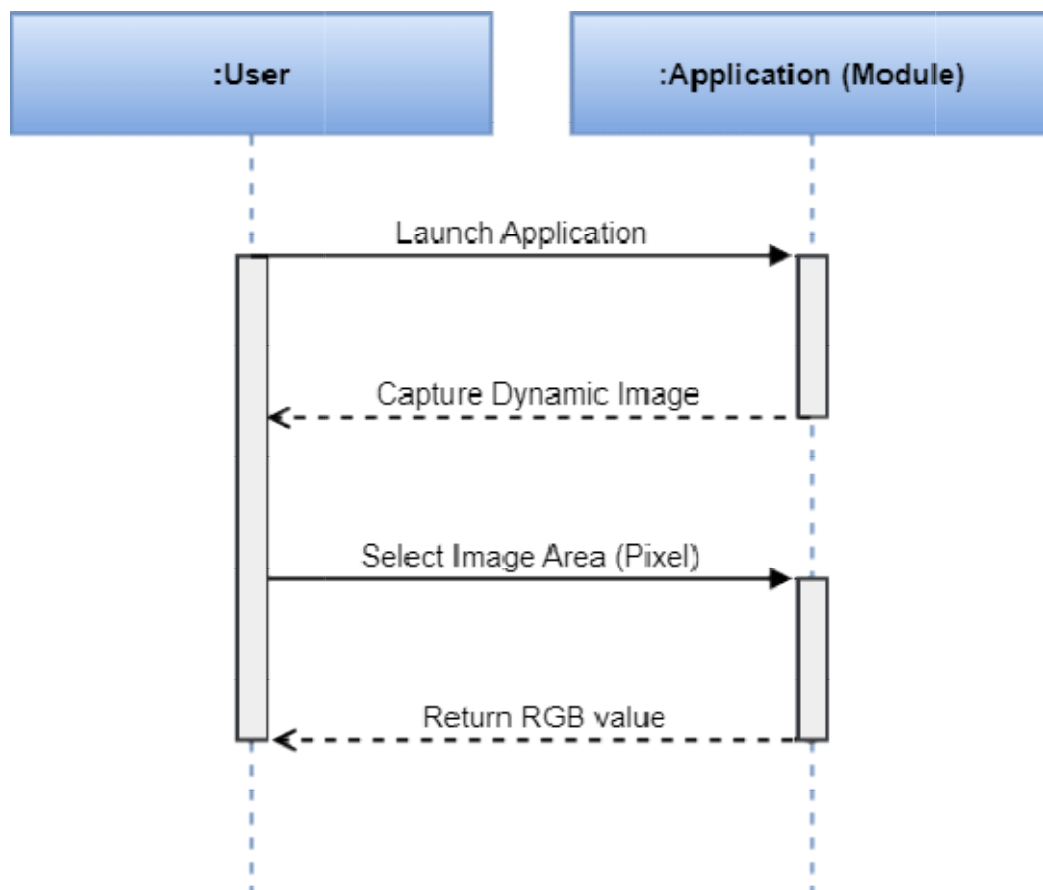


FIGURE NO. 4 - USE CASE DIAGRAM

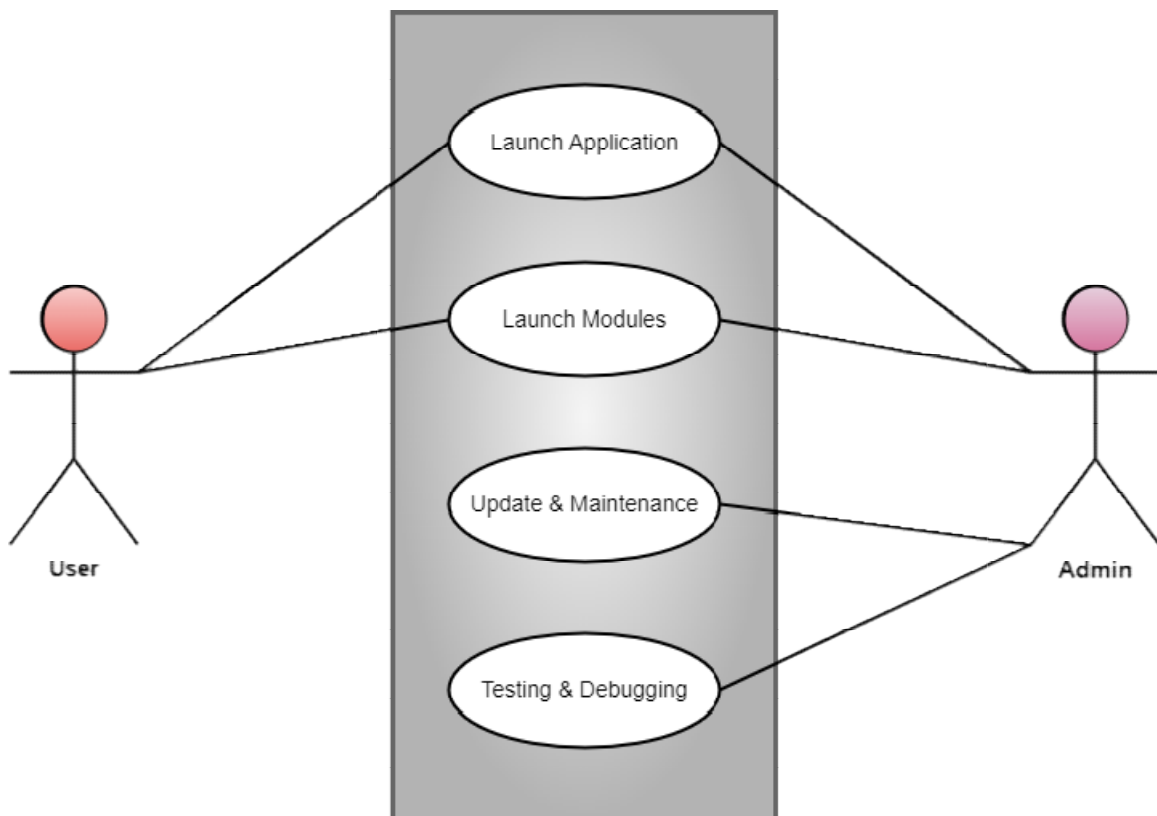


FIGURE NO. 5 - DEPLOYMENT DIAGRAM

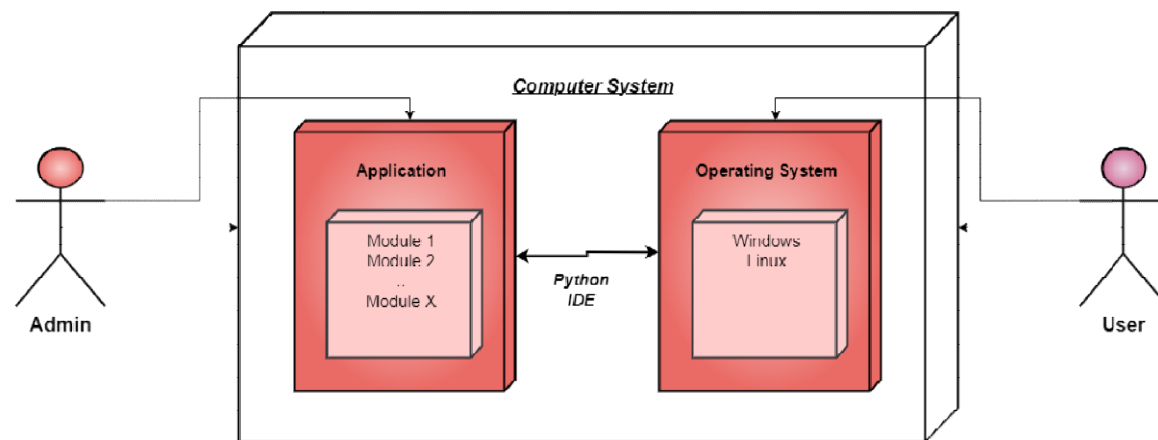
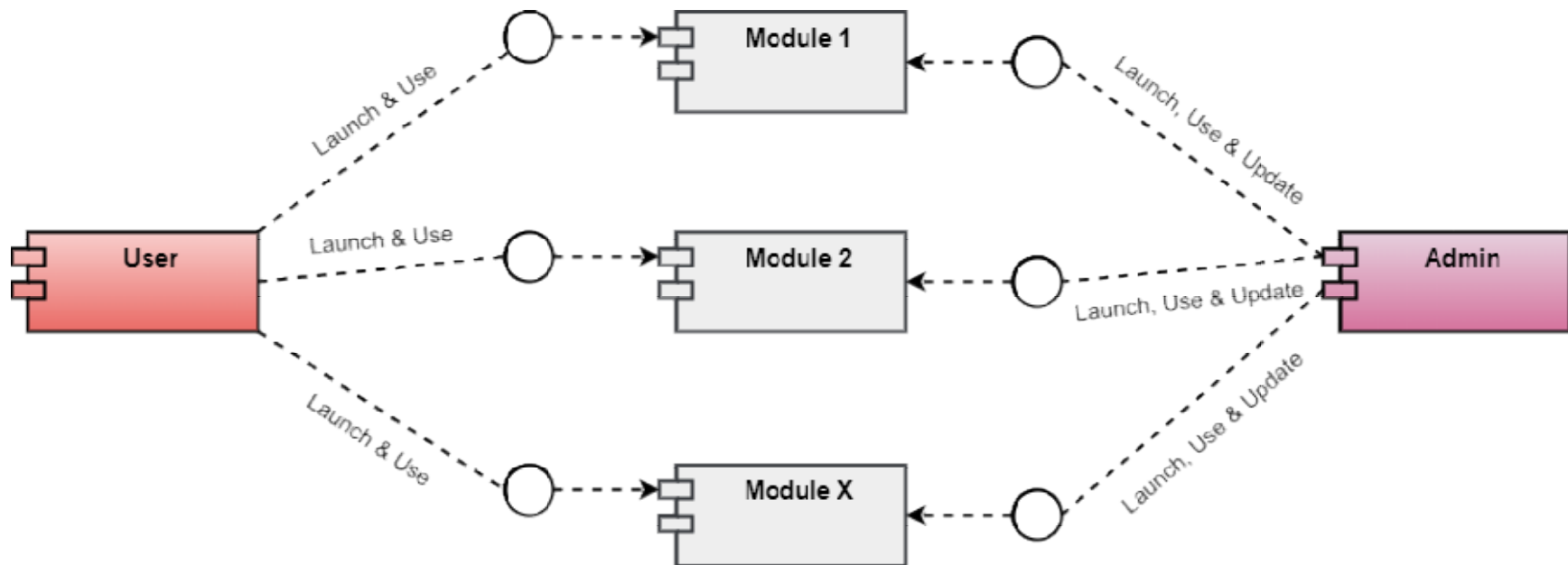


FIGURE NO. 6 - COMPONENT DIAGRAM



METHODOLOGY

A traditional approach involves a series of consecutive stages in the project management process. It is a step-by-step sequence to design, develop and deliver a product or service. It entails achieving the succession in the implementation process and provides the benefits of milestone-based planning and team building. In IT and software development, this methodology type is called “*Waterfall*” – one portion of work follows after another in linear sequence.

The following stages are included the traditional project management methodology:

- *Initiation (requirements specification)*
- *Planning and Design*
- *Execution (construction and coding)*
- *Control and Integration*
- *Validation (testing and debugging)*
- *Closure (installation and maintenance)*

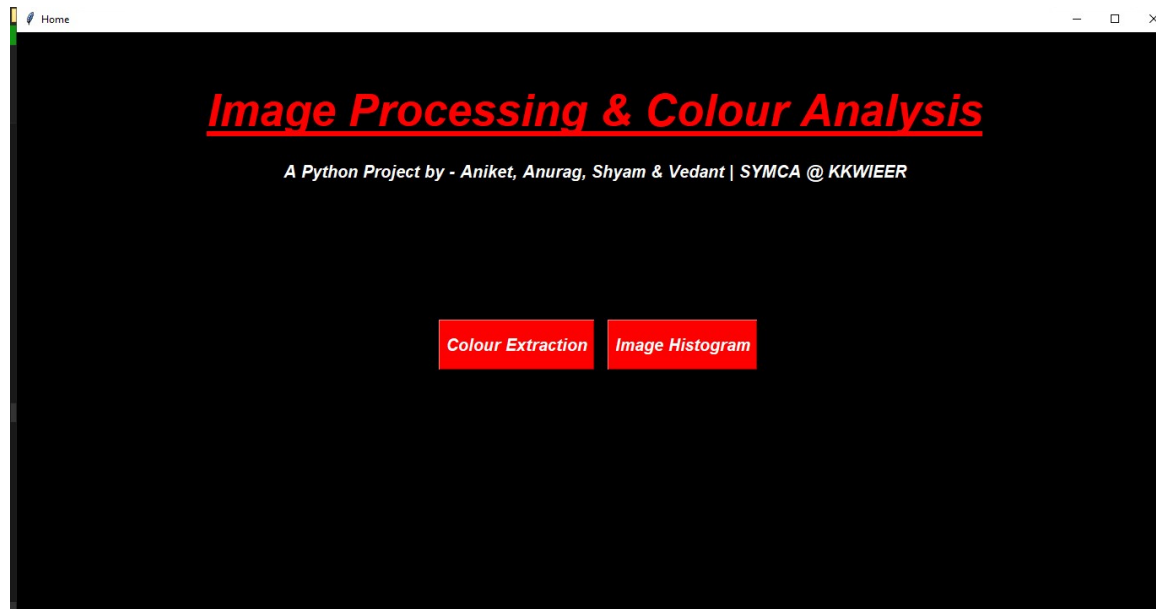
RESULTS

- 1) 'Colour Extraction and Image Histogram' an image processing and data analysis application which involves a dynamic image capture for colour extraction and a local image processing for generating a histogram was created successfully.
- 2) The application is a simple and user-friendly interface currently with two base modules which can be used for understanding Python programming libraries, Image Processing as well as Data Analysis using Histogram charts
- 3) 'Colour Extraction and Image Histogram' application proved to be an exciting project for our team to work on. It included an extensive research, a well-structured design approach and rapid implementation.
- 4) The application is highly flexible and extensible as number of relevant modules can further be embedded into the system and a software with vast libraries and functionalities can thus be developed.

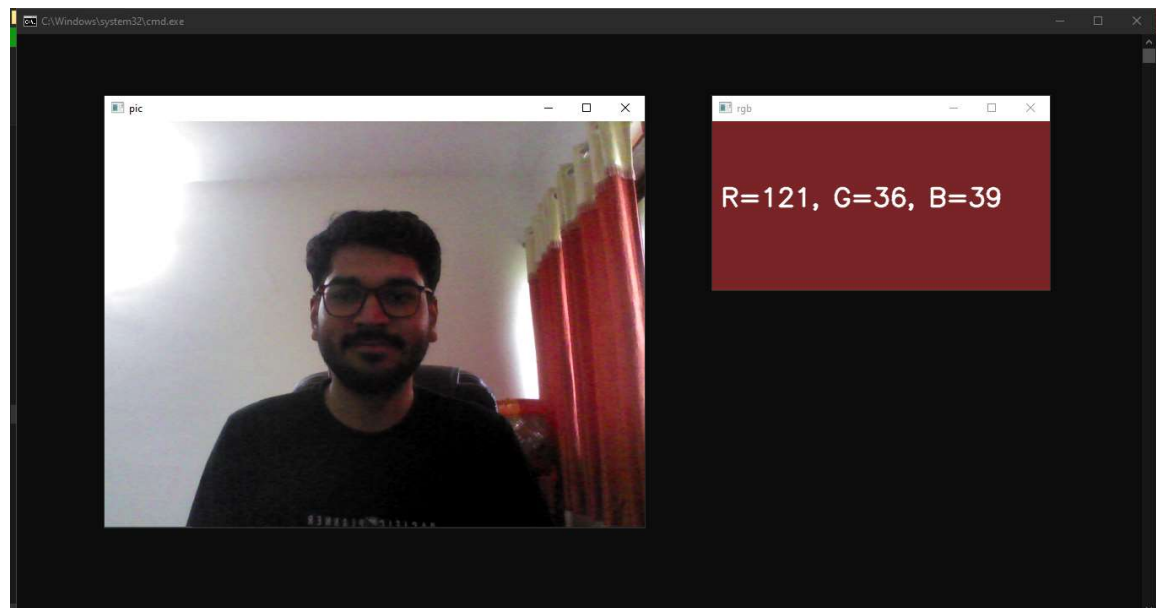
▪ **OUTPUT–**

Following are the snapshots of the application windows with respective modules as –

Home Screen UI, Colour Extraction UI and Image Histogram UI



HOME SCREEN



COLOUR EXTRACTION (DYNAMIC CAPTURE)

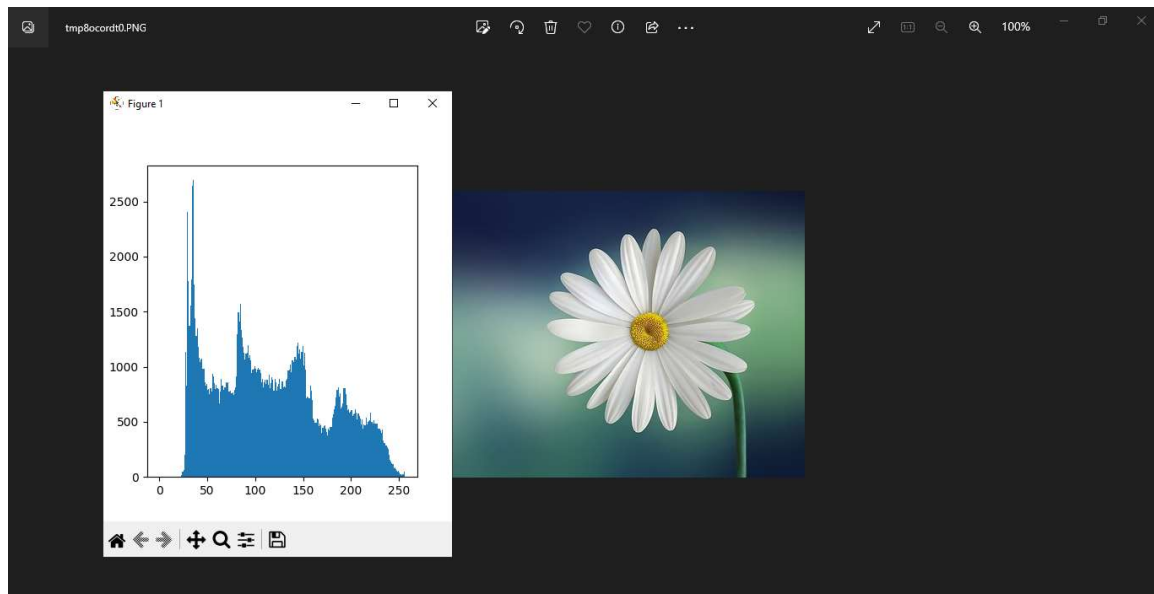


IMAGE HISTOGRAM (LOCAL IMAGE)

TESTING

The ultimate goal of functional testing is to ensure that software works according to specifications and in line with user expectations. While the objective sounds simple, the task involves many functional testing types, some of which might be preferred or prioritized over others depending on the nature of the application and organization.

For example, functional testing types like component, integration and system testing validate the ability to work in isolation or with other components, so the testing strategy might depend on how interoperable the app must be. Similarly, if an organization performs system testing, it might not perform acceptance testing, as that work might be considered redundant.

Another example: alpha, beta and production testing all describe builds that are in a production environment. Your choice between these functional testing types — or how the scale of people involved in each stage — might depend on the stability of the app, the scope of the changes, or timelines set by the business.

Types of testing performed on the project include:

- **UNIT TESTING**

Before you can test an entire software program, make sure the individual parts work properly on their own. Unit testing validates the function of a unit, ensuring that the inputs (one to a few) result in the lone desired output. This testing type provides the foundation for more complex integrated

software. When done right, unit testing drives higher quality application code and speeds up the development process. Developers often execute unit tests through test automation.

- **COMPONENT TESTING**

Also called module testing, component testing checks individual parts of an application. Similar to unit testing, component testing assesses a part of the software in isolation from the broader system.

- **REGRESSION TESTING**

Just because functional tests pass once doesn't mean they'll always pass. When developers commit new code or change a feature, you run regression tests to make sure the software still functions as expected. Regression testing helps maintain a stable product while changes are made to it. Regression tests are often automated.

- **UI TESTING**

With UI testing, professionals interact with the graphical interface of a software program. This includes testing of UI controls like buttons, menus and text input to ensure that the experience flow and features chosen are optimal for the user experience.

- **SYSTEM TESTING**

With system testing, professionals test the software in its entirety, as a complete product. With this type of functional testing, testers validate

the complete and integrated software package to make sure it meets requirements. Where necessary, testers can provide feedback on the functionality and performance of the app or website without prior knowledge of how it was programmed. This helps teams develop test cases to be used moving forward. System testing is also referred to as end-to-end testing.

TEST CASES

<u>Test</u>	<u>Description</u>	<u>Expected Result</u>	<u>Actual Result</u>	<u>Status</u>
1. Functional Testing	<i>To check whether the modules are functioning as intended.</i>	<i>Modules should function as intended</i>	<i>Modules are functioning as intended</i>	<i>Pass</i>
	<i>To check whether the given components are performing their specified actions.</i>	<i>Components should perform specified actions</i>	<i>Components perform specified actions</i>	<i>Pass</i>
	<i>To check whether modules can be launched properly</i>	<i>Modules should launch on demand</i>	<i>Modules launch on demand</i>	<i>Pass</i>
	<i>To check whether image capture is functioning as intended</i>	<i>Image capture should function as intended</i>	<i>Image capture is functioning as intended</i>	<i>Pass</i>

2. UI Testing	<i>To check whether the UI elements are clearly visible</i>	<i>UI elements should be visible</i>	<i>UI elements are visible</i>	<i>Pass</i>
	<i>To check whether UI elements denote their respective functionalities clearly</i>	<i>UI elements should denote their respective functionalities</i>	<i>UI elements denote their respective functionalities</i>	<i>Pass</i>
3. Compatibility Testing	<i>To check whether the website application is running on latest operating systems e.g. Windows 10 and Linux</i>	<i>Application should run on latest operating systems</i>	<i>Application runs on latest operating systems</i>	<i>Pass</i>
	<i>To check whether web camera is accessible and compatible with the application</i>	<i>Web Camera access must be compatible with the application</i>	<i>Web Camera access is compatible with the application</i>	<i>Pass</i>

CONCLUSION

- 5) 'Colour Extraction and Image Histogram' an image processing and data analysis application which involves a dynamic image capture for colour extraction and a local image processing for generating a histogram was created successfully.
- 6) The application is a simple and user-friendly interface currently with two base modules which can be used for understanding Python programming libraries, Image Processing as well as Data Analysis using Histogram charts
- 7) 'Colour Extraction and Image Histogram' application proved to be an exciting project for our team to work on. It included an extensive research, a well-structured design approach and rapid implementation.
- 8) The application is highly flexible and extensible as number of relevant modules can further be embedded into the system and a software with vast libraries and functionalities can thus be developed.

FUTURE ENHANCEMENTS

- **WEB-BASED APPLICATION SUPPORT**

The application can be made compatible for web-based support so that users need not install and update the software for a number of times.

- **CLOUD-BASED APPLICATION SUPPORT**

The application can be made suitable for cloud-based support so that users will be able to use the application remotely and also save their personal data on their own personal cloud storage system.

- **INTERACTIVE UI**

The application can include of a more expressive user interface design with animations and characterizations as an eye candy as well as to provide the users an immersive experience while using the application.

- **AUGMENTED REALITY / VIRTUAL REALITY INTEGRATION**

The application can synergize with latest visual tools and technologies such as AR, VR and MR to dive into a new world of image and graphics processing and 3D rendering.

- **ADDITIONAL IMAGE PROCESSING MODULES**

The application can involve other image processing or formatting module such as:

- Local image upload module
- Image filters module
- Image transformation module
- Global pattern recognition module

BIBLIOGRAPHY

Reference Books :

- *Python Crash Course: A Hands-On, Project-Based Introduction to Programming by Eric Matthes*
- *Learn Python 3 The Hard Way, by Zed A. Shaw*

Research Papers :

- *Color Detection of RGB Images Using Python and OpenCv P. Raguraman^{1*}, A. Meghana², Y. Navya³, Sk. Karishma⁴, S. Iswarya⁵*

Programming Documentations :

- OpenCV - <https://docs.opencv.org/4.x/>
- Numpy - <https://numpy.org/doc/stable/>
- Tkinter - <https://docs.python.org/3/library/tkinter.html>
- Pillow - <https://pillow.readthedocs.io/en/stable/>

ACKNOWLEDGEMENT

We would like to express our special thanks of gratitude to our Project Guide **Prof. Mariyam Maniyar**, Dept. Head **Prof. Dr. V. C. Bagal** and Principal **Dr. K. N. Nandurkar** who gave us this opportunity to work on this project of the topic '**Colour Extraction and Image Histogram**'. It motivated us in conducting good research and we came to light about a lot of educational concepts in fields of studies like Colour Recognition, Machine Learning, Data Analysis, Image Processing and also about Python libraries such as OpenCV and TKinter.

Furthermore, we would also like to thank our parents and our friends who helped us during the development of our project by keeping us encouraged and determined.

- 'Colour Extraction and Image Histogram' Project Team

Dixit Vedant Haresh

Hingrajiya Shyam Niteshbhai

Shinde Aniket Rajendra

Nagare Anurag Ramesh