KLE Society's

KLE Technological University



**Structured Enquiry Assessment Report**

**On**

**" Harnessing C++ for Image Analysis”**

**Object Oriented Programming (20ECSC204)  
Object Oriented Programming Lab (20ECSP203)**

Submitted by

| **Name** | **Roll no** | **SRN** |
| --- | --- | --- |
| Harsh Nesari | 13 | 02FE22BCS040 |
| Mohd.Qadir Ternikar | 16 | 02FE22BCS053 |
| Mrunali Bennalkar | 17 | 02FE22BCS055 |
| Vanashree Nandaganve | 65 | 02FE22BCS171 |
| **Team Number:** A-16 | | |

Faculty In-charge:

**Prof . Vaishali Parab**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

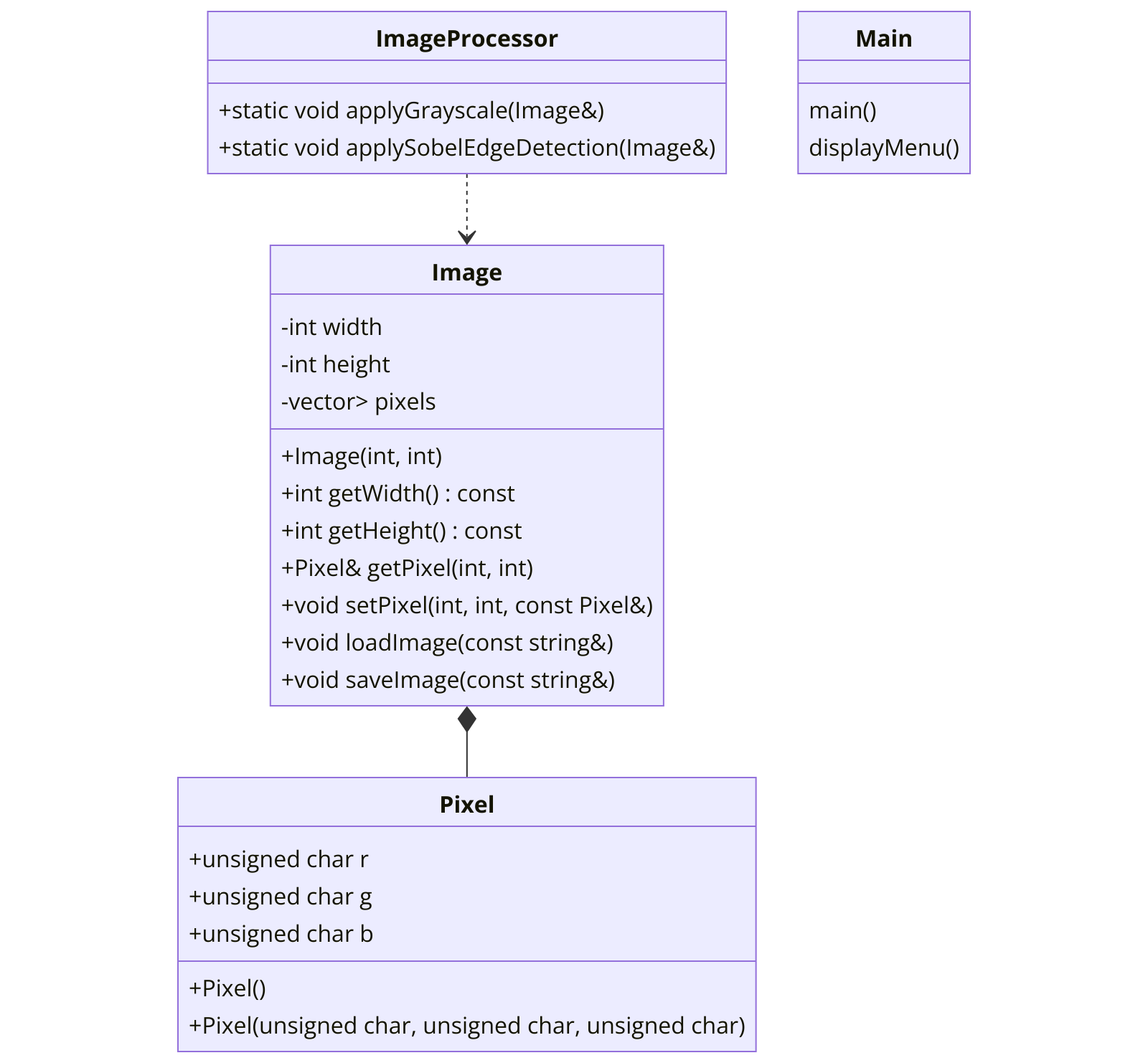
Academic year 2023-24

**CONTENTS**

| **1.** | **Introduction** | |
| --- | --- | --- |
|  | 1.1 | Overview of the problem statement |
|  | 1.2 | Features of Application |
| **2.** | **Design** | |
|  | 2.1 | Class Diagrams |
| **4** | **Implementation** | |
|  | 4.1 | Results |

| **1.** | **Introduction** | |
| --- | --- | --- |
|  | 1.1 | **Overview of the problem statement:**  In designing a program for processing and analyzing image data in C++, the implementation should encompass several key components: classes for image representation, algorithms for processing, and methods for manipulation and analysis. The foundation involves a class dedicated to storing pixel information, typically encapsulating properties such as width, height, and pixel data arrays. This class, often named Image, would provide essential methods for loading, saving, and accessing pixel values. On top of this, various image processing algorithms such as filtering, edge detection, and transformation functions would be implemented as member functions or external functions operating on instances of the Image class. |
|  | 1.2 | **Features of Application:**  The image processing application described provides several key features aimed at handling image data, applying filters, and performing basic image manipulations. Here are the features considered for implementation:   1. **Loading and Saving Images**:    * Ability to load images from files.    * Ability to save processed images back to files. 2. **Pixel Manipulation**:    * Setting and getting pixel values.    * Checking and handling pixel coordinates to avoid out-of-bound errors. 3. **Image Filters and Transformations**:    * Applying a grayscale filter to convert images to grayscale.    * Applying Sobel edge detection to highlight edges in images. 4. **User Interface**:    * Command-line menu for interacting with the application.    * Options to load, save, apply filters, and exit the application. |

UML DIAGRAM:



OUTPUT:

