DISSERTATION

ON

FilterTroy

(MCA – IV SEMESTER; MCA 2022-24)

Internal Guide

Dr. Parul Arora

(Asst. Prof., BVICAM, New Delhi)

Date of Presentation

Presentation by

Bhavesh Malhotra

(07511604422)

Contents

- Problem Statement
- Aim and Objectives
- Technology Used for Project Development
- My Role
- Deliverables
- Diagrams
- Screenshots
- Conclusion
- Future Scope
- Bibliography

Problem Statement

Existing photo editing applications offer pre-defined filters, limiting users' ability to apply unique visual effects to their images. Furthermore, there is a lack of platforms where users can easily create and share their own custom filters, inhibiting artistic expression and community interaction in the digital photography realm.

Aim

Empower users to personalize their photo editing experience by creating and applying custom filters through FilterTroy. Facilitate easy sharing and collaboration within a vibrant community to enhance artistic expression and foster engagement in digital photography.

Objectives

- Develop an intuitive user interface on FilterTroy for seamless photo uploading and filter application.
- Implement a robust compiler system to enable users to create and customize their own unique filters.
- Establish a secure platform that prioritizes user privacy and data protection.
- Cultivate an engaged community through features facilitating the sharing and exploration of custom filters.
- Continuously optimize performance and user experience based on feedback and usage analytics.

Technology Used

- Programming Language: Python, Javascript
- API Framework: Fast API
- Database: MySQL
- Image Processing: OpenCV, Numpy, SkImage
- UI Libraries: React, Redux, Radix, Axios, Sass
- Source Code Management: GIT,GITHUB
- Editor: VS code
- Containerization: Docker

Project Modules

- Project Modules:
    - Employee
    - HR
    - Administrator
- My Role in the Project:

Deliverables

Innovation and User Engagement:

Live Coding Environment: An interactive editor where users can write, test, and see the effects of their filters in real time.

Public Showcases: Regular events where users can present their most innovative filters and discuss their techniques with the community.

Collaborative Projects: Features that enable users to co-develop filters, merging different techniques and ideas from multiple creators.

Deliverables

Community and Collaboration:

Filter Gallery: A showcase for users to upload and display their custom filters.

Community Forums: Spaces for discussion, feedback, and collaborative projects.

User Profiles and Networking: Features allowing users to connect, follow, and interact with peers.

Deliverables

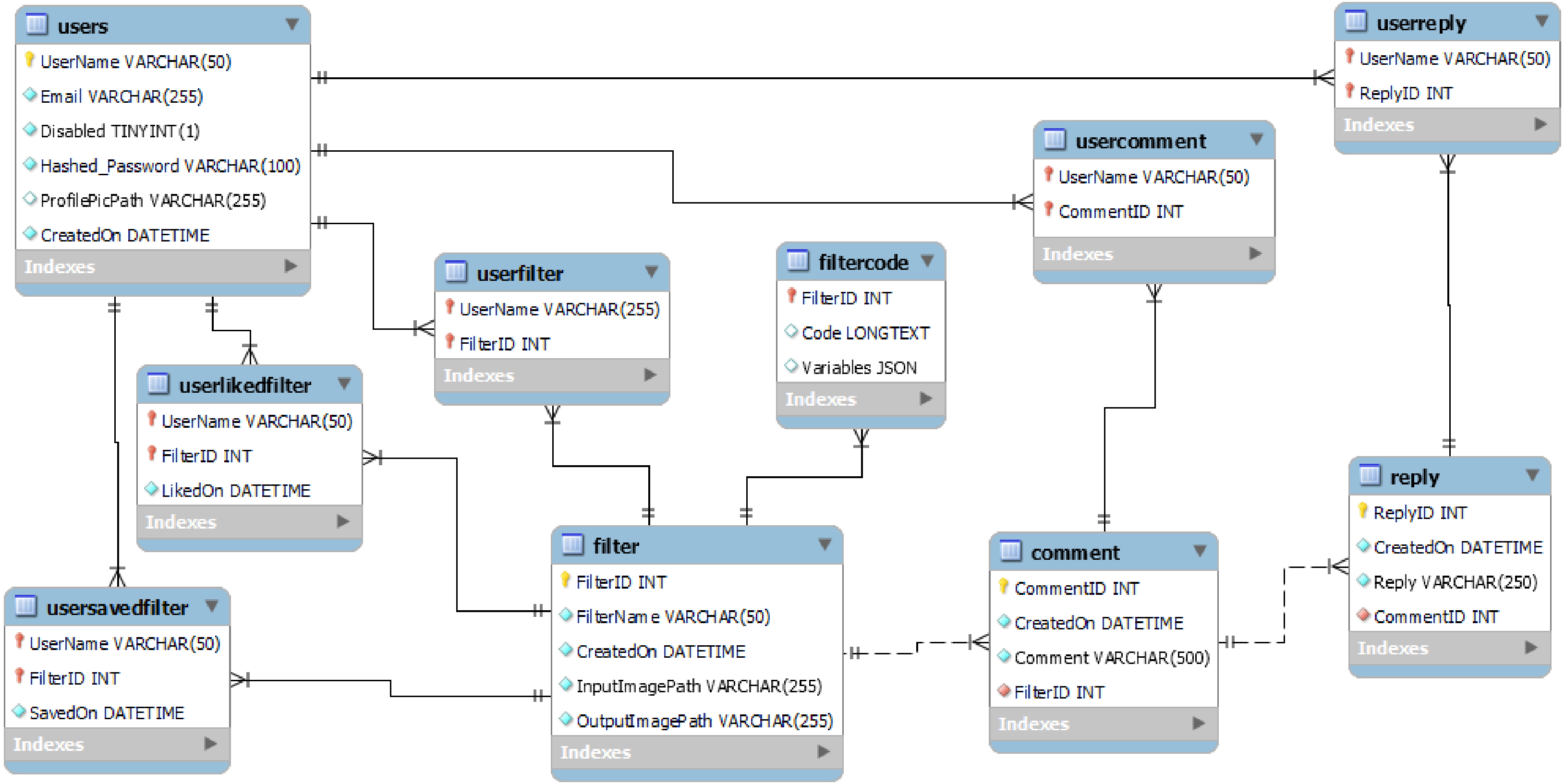
Advanced Features and Tools:

Real-Time Feedback System: An instant feedback mechanism allowing users to receive suggestions and optimizations for their filter codes.

Custom Libraries and APIs: Integration of advanced libraries and APIs that users can incorporate into their filters for enhanced functionality.

Performance Analytics: Tools for users to analyze the performance of their filters, including usage statistics and user engagement metrics.

Entity-Relationship (E-R) Diagram



users
- UserName VARCHAR(50)
- Email VARCHAR(255)
- Disabled TINYINT(1)
- Hashed_Password VARCHAR(100)
- ProfilePicPath VARCHAR(255)
- CreatedOn DATETIME
- Indexes

userreply
- UserName VARCHAR(50)
- ReplyID INT
- Indexes

usercomment
- UserName VARCHAR(50)
- CommentID INT
- Indexes

userfilter
- UserName VARCHAR(255)
- FilterID INT
- Indexes

filtercode
- FilterID INT
- Code LONGTEXT
- Variables JSON
- Indexes

userlikedfilter
- UserName VARCHAR(50)
- FilterID INT
- LikedOn DATETIME
- Indexes

usersavedfilter
- UserName VARCHAR(50)
- FilterID INT
- SavedOn DATETIME
- Indexes

filter
- FilterID INT
- FilterName VARCHAR(50)
- CreatedOn DATETIME
- InputImagePath VARCHAR(255)
- OutputImagePath VARCHAR(255)
- Indexes

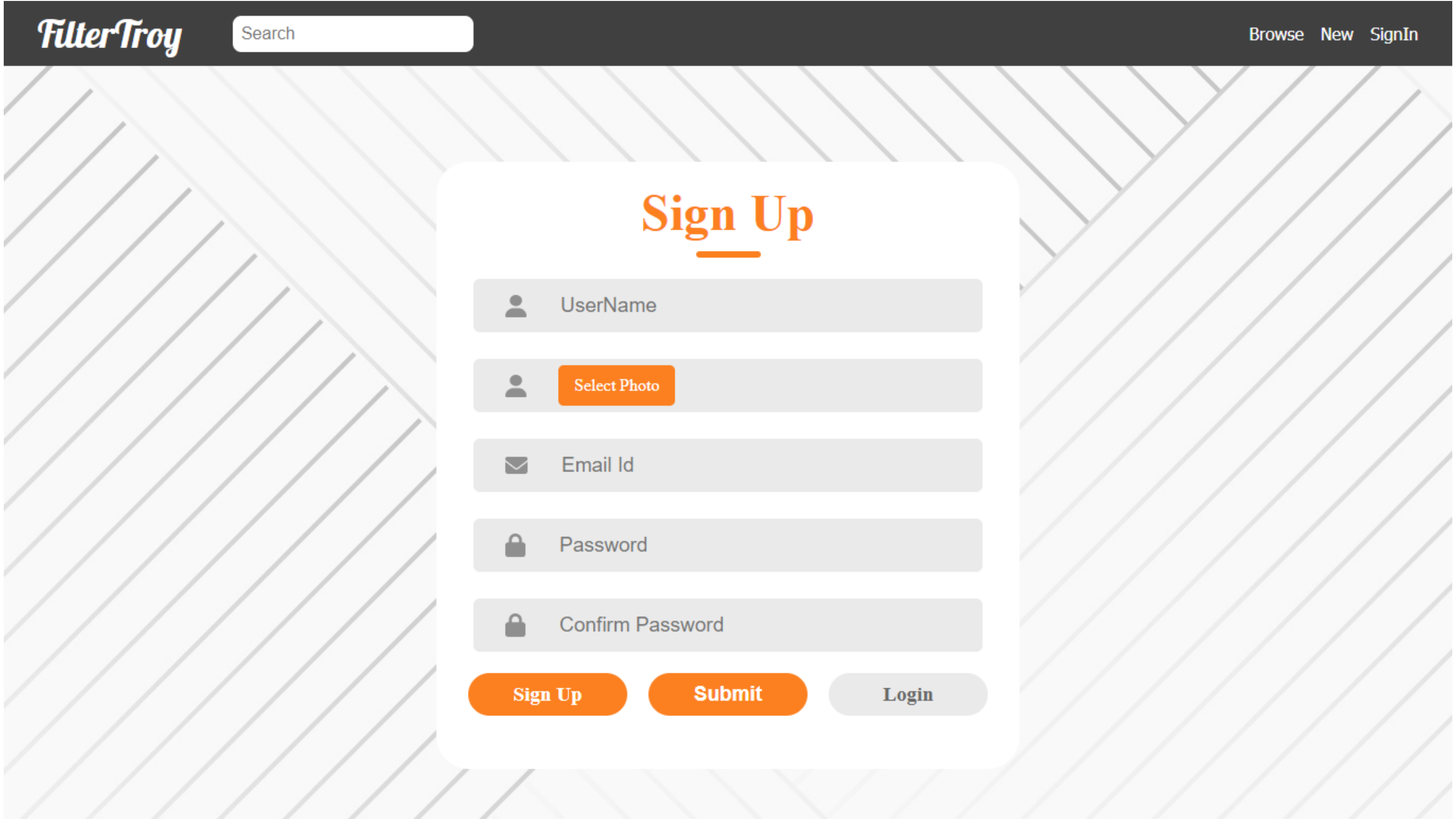
comment
- CommentID INT
- CreatedOn DATETIME
- Comment VARCHAR(500)
- FilterID INT
- Indexes

reply
- ReplyID INT
- CreatedOn DATETIME
- Reply VARCHAR(250)
- CommentID INT
- Indexes

Use Case

Screenshots

Screenshot – 1



FilterTroy

Search

Browse New SignIn

Sign Up

UserName

Select Photo

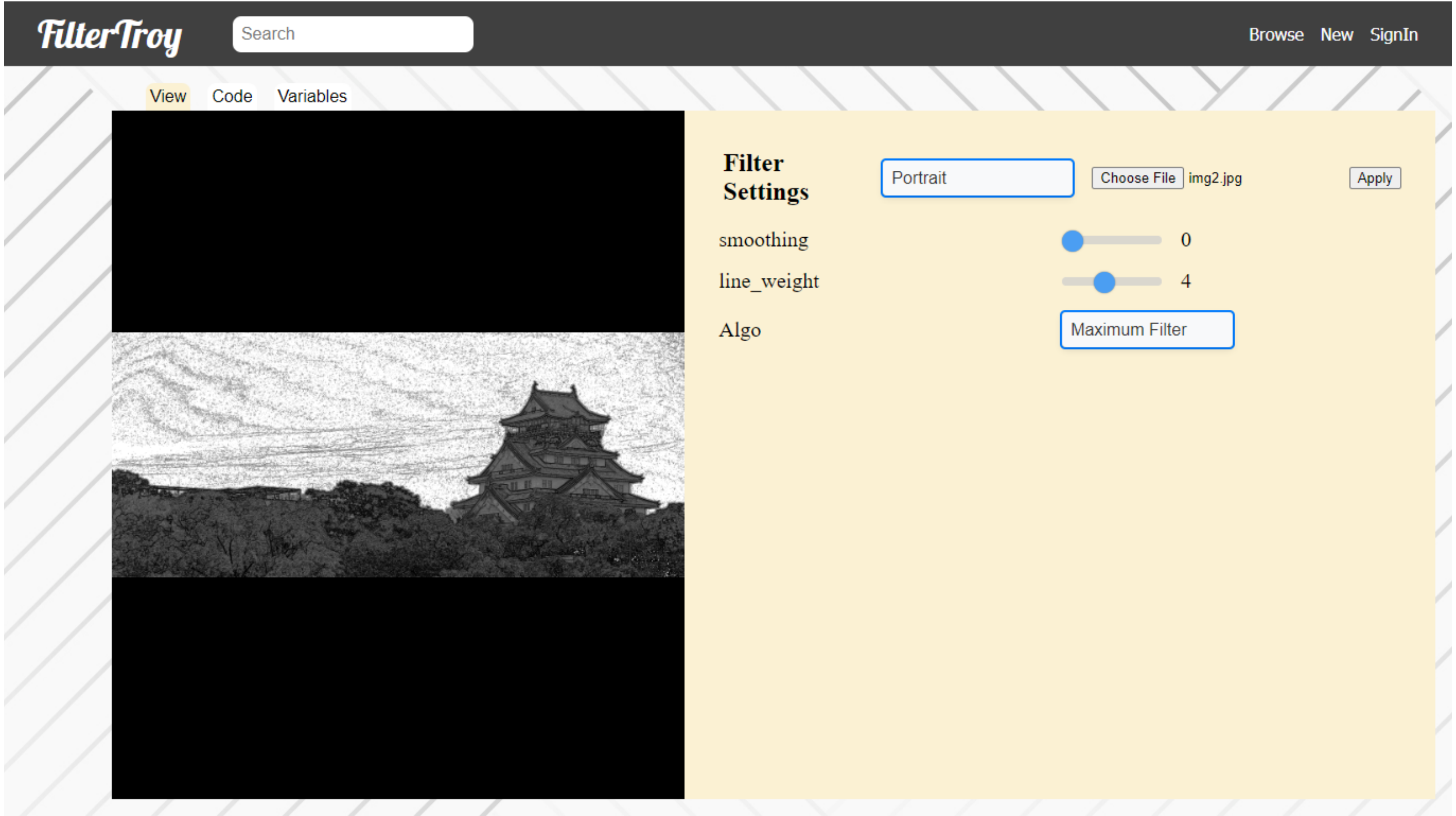
Email Id

Password

Confirm Password

Sign Up Submit Login

Screenshot - 2



Filter Settings

Portrait

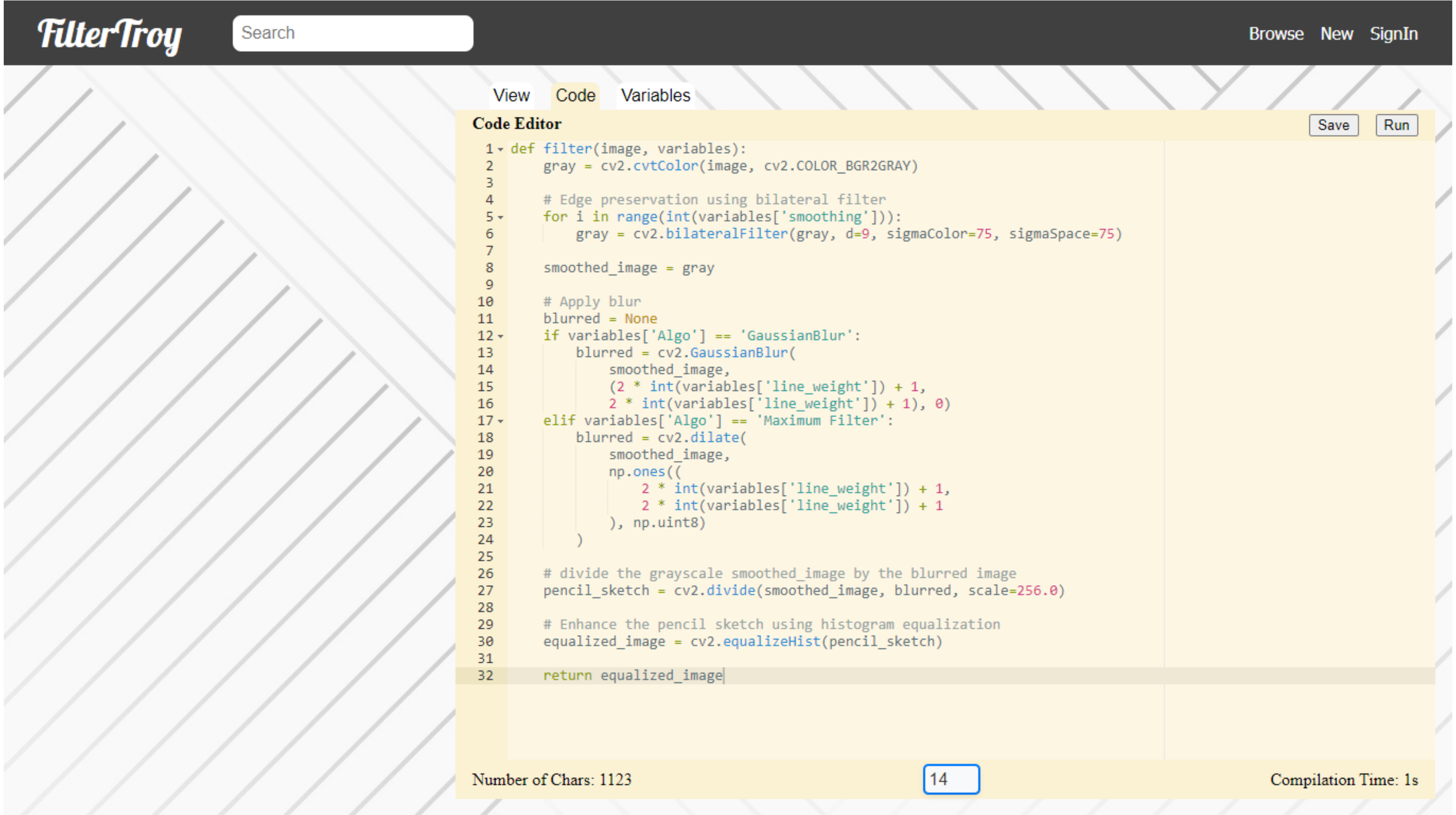
img2.jpg

smoothing 0

line_weight 4

Algo Maximum Filter

Screenshot – 3



Code Editor

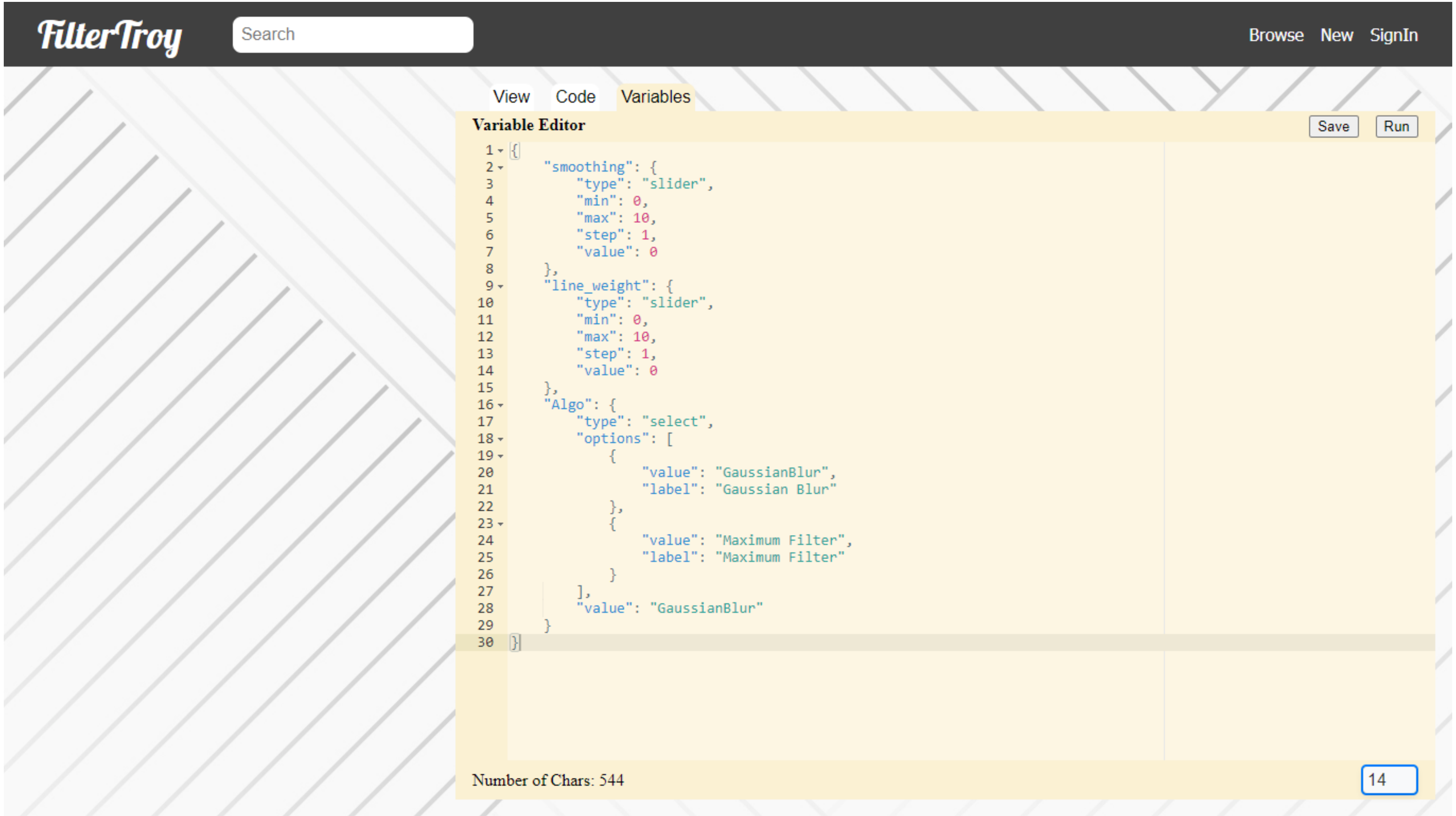
```python
def filter(image, variables):
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

    # Edge preservation using bilateral filter
    for i in range(int(variables['smoothing'])):
        gray = cv2.bilateralFilter(gray, d=9, sigmaColor=75, sigmaSpace=75)

    smoothed_image = gray

    # Apply blur
    blurred = None
    if variables['Algo'] == 'GaussianBlur':
        blurred = cv2.GaussianBlur(
            smoothed_image,
            (2 * int(variables['line_weight']) + 1,
            2 * int(variables['line_weight']) + 1), 0)
    elif variables['Algo'] == 'Maximum Filter':
        blurred = cv2.dilate(
            smoothed_image,
            np.ones((
                2 * int(variables['line_weight']) + 1,
                2 * int(variables['line_weight']) + 1
            ), np.uint8)
        )

    # divide the grayscale smoothed_image by the blurred image
    pencil_sketch = cv2.divide(smoothed_image, blurred, scale=256.0)

    # Enhance the pencil sketch using histogram equalization
    equalized_image = cv2.equalizeHist(pencil_sketch)

    return equalized_image
```

Number of Chars: 1123

14

Compilation Time: 1s

Screenshot – 4



Variable Editor

```
{
    "smoothing": {
        "type": "slider",
        "min": 0,
        "max": 10,
        "step": 1,
        "value": 0
    },
    "line_weight": {
        "type": "slider",
        "min": 0,
        "max": 10,
        "step": 1,
        "value": 0
    },
    "Algo": {
        "type": "select",
        "options": [
            {
                "value": "GaussianBlur",
                "label": "Gaussian Blur"
            },
            {
                "value": "Maximum Filter",
                "label": "Maximum Filter"
            }
        ],
        "value": "GaussianBlur"
    }
}
```

Number of Chars: 544

14

Conclusion

- **Empowerment through Customization:** FilterTroy provides users with the power to customize their photo editing experience through the creation and application of custom filters, fostering individuality and creativity.
- **Community-Centric Approach:** By facilitating collaboration, sharing, and discussion within a vibrant community, FilterTroy enriches the digital photography realm with diverse perspectives and innovative techniques.
- **Continuous Improvement:** Through feedback mechanisms, performance optimization, and the integration of advanced features, FilterTroy is committed to evolving to meet the evolving needs and expectations of its users.
- **Impact on Artistic Expression:** FilterTroy not only enhances the quality of edited photos but also serves as a catalyst for artistic expression, inspiring users to explore new styles and techniques in digital photography.

Future Scope

Mobile Integration and Social Media Connectivity:

Camera API Integration: Development of features that allow Filter Troy to integrate directly with smartphone cameras. Users will be able to capture photos, apply filters in real time, and preview effects before saving.

Instant Filter Application: Enhancements to allow seamless and quick application of filters to images taken with mobile devices, using optimized algorithms for speed and quality.

Social Media Sharing Tools: Built-in functionality for users to share their filtered photos directly to social media platforms like Instagram, Facebook, and Twitter. This includes easy-to-use sharing buttons and the ability to add captions and tags.

Future Scope

Technological Advancements:

AI-Driven Tools: Integration of AI technologies to assist users in creating more complex and refined filters automatically based on their preferences.

Augmented Reality Filters: Development of tools for creating AR-compatible filters that can be used in various AR applications and games.

Mobile Platform Development: Expansion of Filter Troy into mobile platforms, allowing users to create and share filters directly from their smartphones and tablets.

Bibliography

Bibliography:

- https://www.chat.openai.com
- www.google.com
- OpenCV (Open Source Computer Vision Library for computer vision, machine learning, and image processing)
- SkImage (a.k.a. skimage) is a collection of algorithms for image processing and computer vision
- Docker(platform for developing, shipping, and running applications using containerization technology)
- React(https://react.dev/learn)

Thank You!