

User Manual- Dog-A-Eye Photo Enhancement Web App

Introduction:

Welcome to Dog-A-Eye, a photo enhancement web app dedicated to assisting clients of our dog eye clinic in capturing high-quality photographs of their dogs' eyes for professional evaluation. This manual provides a step-by-step walkthrough to ensure you make the most out of the Dog-A-Eye app, ultimately aiding in the accurate and prompt assessment of your dog's eye health by our veterinary ophthalmologists.

1. Getting Started:

- Upon accessing [Dog-A-Eye](#), the landing page provides useful tips on how to capture a good photo of your dog's eye.
- It's advisable to read through these tips before proceeding.

2. Uploading a Photo:

Upload Button:

- Click on the main interaction button labeled "Upload an Image" to upload a picture of your dog's eye.
- Follow the prompts to select and upload the photo from your device.

3. Photo Quality Feedback:

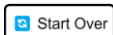
Good Quality Photo:

- If the photo is of good quality, you will be able to share it with the clinic via WhatsApp.

Poor Quality Photo:

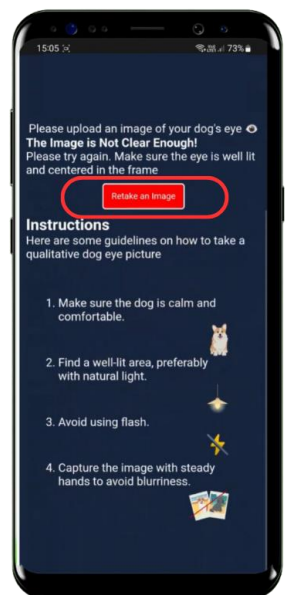
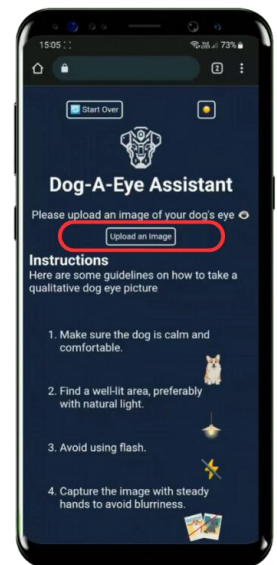
- If the photo is not of good quality, you will be asked to retake the picture.
- It is advised to read the instructions on the landing page again before retaking the photo.

4. Restarting the Process:

At any point, if you wish to restart the process, click on the  button visible on the screen.

5. Changing Theme:

To change the theme of the app, click on the "Change Theme" button. Choose from the available theme options to customize your viewing experience.

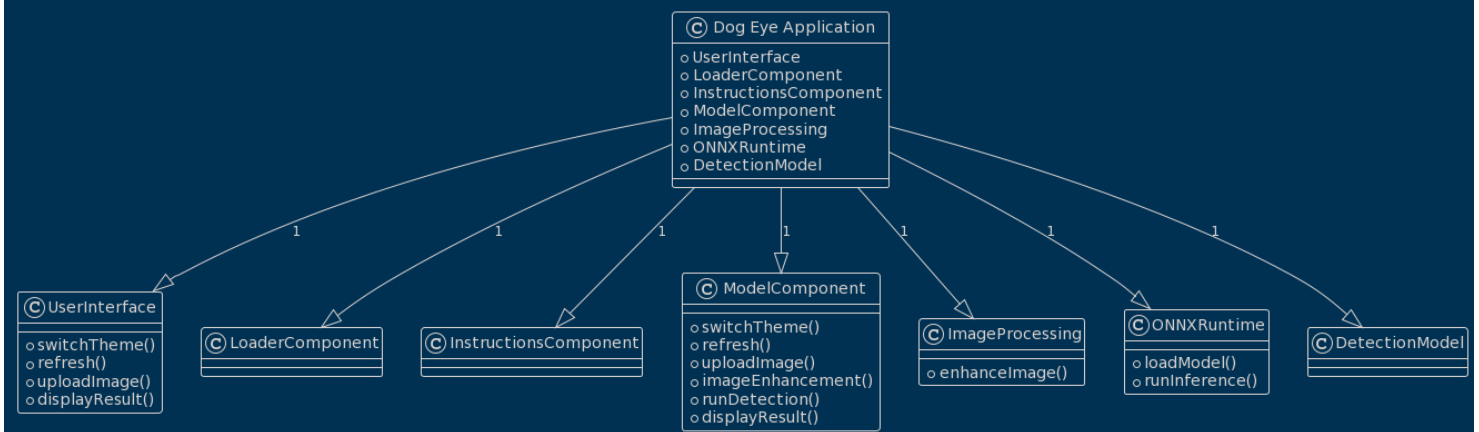


Developer Manual- Dog-A-Eye Photo Enhancement Web App

Introduction:

The "Dog Eye" project is an application designed to assist dog owners and veterinarians communicate better by ensuring good quality photos of dog's eyes for the vet to first evaluate the condition of the dog. This developer manual provides information on setting up the development environment and understanding the project's architecture.

Dog Eye Application - Class Diagram



1. App Components:

Dog Eye Application (App): This is the main class and serves as the central component of the system. It encapsulates the overall functionality and orchestrates the interactions between different parts of the application.

User Interface: The User Interface class represents the user interface of the application. It provides functions for switching themes, refreshing the interface, uploading images, and displaying results.

Loader Component: This component is responsible for handling data loading operations, although specific functionality details are not provided in the diagram.

Model Component: The Model Component is a crucial part of the application, responsible for a range of tasks including theme switching, refreshing, image uploading, image enhancement, running detection, and displaying results.

Image Processing: The Image Processing class encapsulates image processing capabilities, particularly image enhancement. It is connected to the Detection Model, indicating that image processing plays a role in the detection process.

ONNXRuntime: This class deals with loading machine learning models and running inference using the ONNX runtime. It is closely related to the application's functionality.

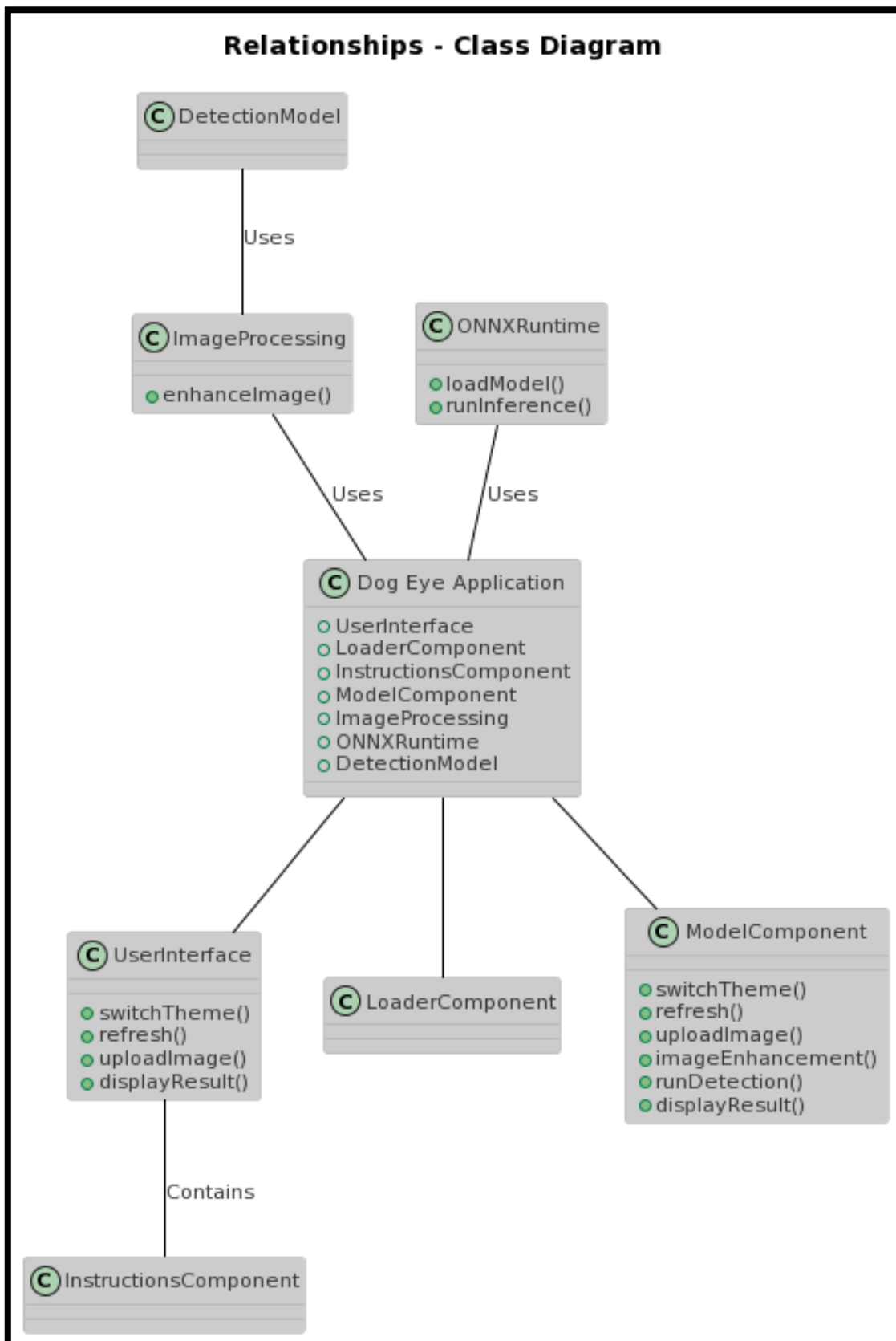
Detection Model: The Detection Model represents a machine learning model used for dog eye detection. It has a dependency on YOLO V8 open-source Detection model.

Image Processing: indicating that it utilizes image processing capabilities during the detection process.

Instructions Component: Instructions Component is part of the User Interface and provides instructional guidance to the user modifiable for future usage or changes.

In summary, the "Dog Eye Application" architecture comprises several interconnected components that work together to offer features such as user interface interaction, image loading and processing, machine learning-based detection, and result presentation. The use of classes and their relationships in the diagram helps illustrate how these components interact within the system.

2. Class Diagram:



2. Getting Started:

a. Prerequisites:

Before you can start working on the "Dog Eye" project, make sure you have the following prerequisites installed:

Node.js:

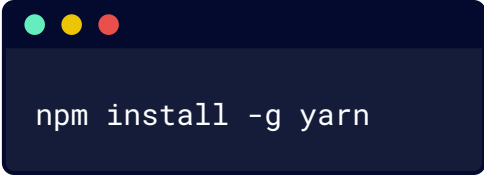
b. Installation:

To set up your development environment, follow these steps:

Install Node.js if you haven't already.

Install Yarn globally and set it to the latest version by running the following commands:

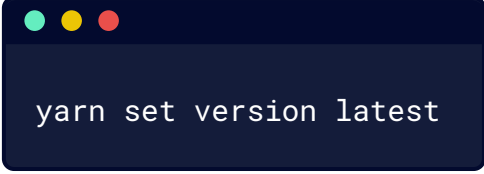
Bash (Copy code)



```
npm install -g yarn
```



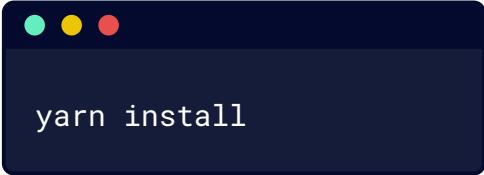
```
yarn set version berry
```



```
yarn set version latest
```

c. Install all the project dependencies by navigating to the project directory and running:

Bash (Copy code)




```
yarn install
```

d. Running the Application

To run the application locally for development and testing purposes, execute the following command:

Bash (Copy code)



```
yarn dev
```

The application will be accessible at <http://localhost:3000>.

3. Overview:

Overview:

The "Dog Eye" project follows a structured directory layout to maintain code organization and clarity.

Directory Structure

src: Contains the application's source code.

components: React components.

styles: Stylesheets and CSS files.

utils: Utility functions.

public: Static assets and HTML templates.

LICENSE.md: The project's license file.

Key Files

src/components: This directory contains the React components used to build the user interface of the application.