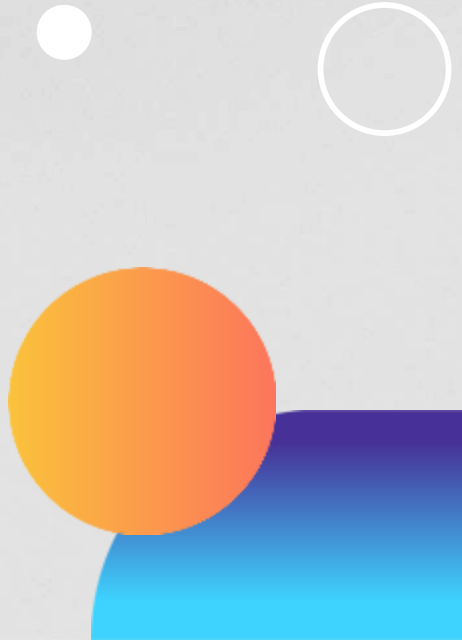
DevOps CA2

Lim Zhen Yang | 2214506 | DOAA CA2

What is DevOps?

DevOps is a methodology that integrates software development and IT operations through automation, collaboration, and communication to deliver high-quality software faster and more efficiently.

The utilization of code testing and RPA for continuous integration testing, coupled with a deploy hook for continuous deployment, embodies DevOps principles by automating and integrating key processes in the software development lifecycle.



Model Development

Saving models

Naming folders with timestamps for version tracking

Testing models

I did two forms of testing, single input and batch input to ascertain the reliability of the deployed models especially during continuous integration

Containerizing models

With Docker and TensorFlow serving, we can serve both models from the same container efficiently.

Deploying models

To ensure continuous availability of the model, any changes to the main branch is automatically deployed with a deploy hook. This is done with GitLab's CICD variables for security

Models deployed on: https://ca2-model-server-mfp2.onrender.com

Web Application Development

Developing backend routes

Use of Postman to check route json returns, status codes, error messages

Detailed testing

For each API route, I performed extensive testing such as successful calls and returns, as well as validating expected failures and their returns

Developing frontend

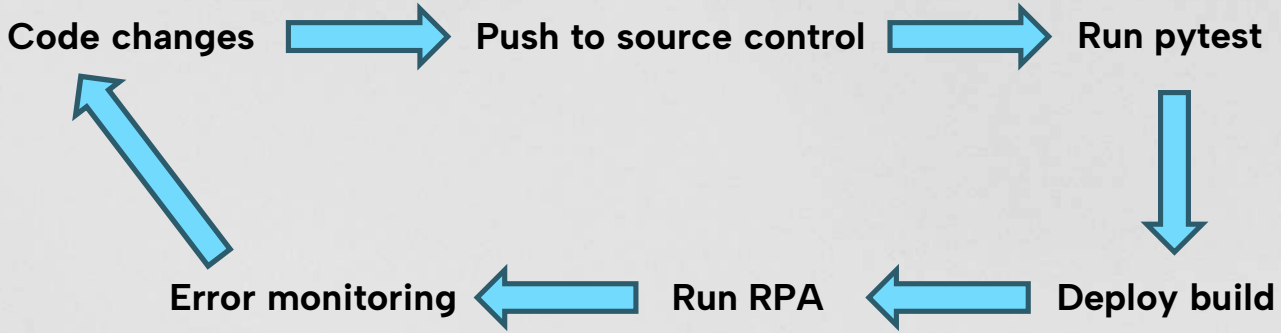
Vite + React for fast development and deployment. Automatic refreshing of pages during development for efficiency.

Deploying the application

Docker to containerize the web application. Built from the base image python 3.8.

Deployed on: https://greens-ai.onrender.com

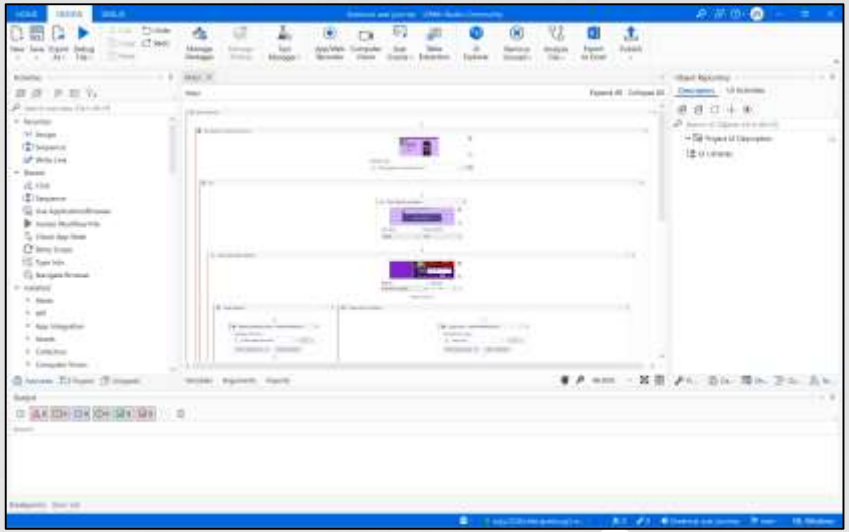
Development Cycle



RPA (UiPath)

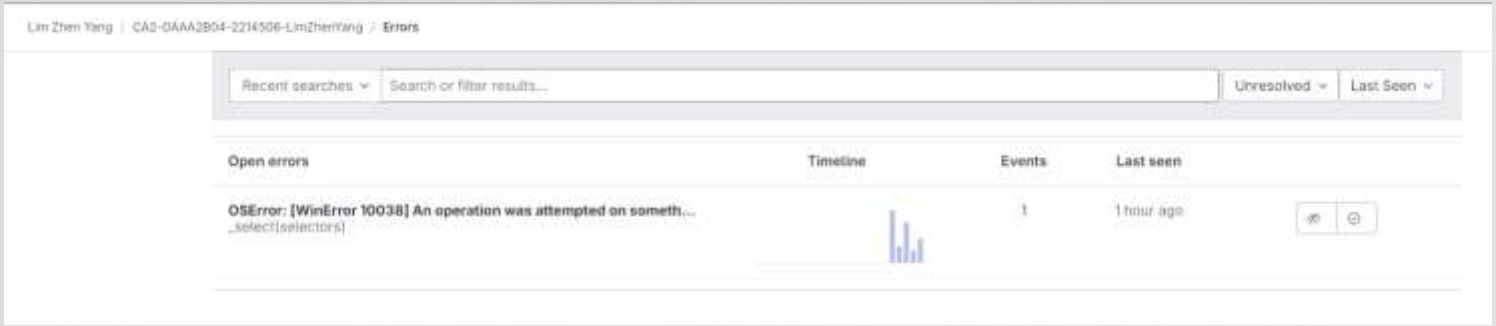
Simulate an entire user journey on the website

- Login
- Sign Up
- Make predictions
- Delete predictions
- Searching predictions

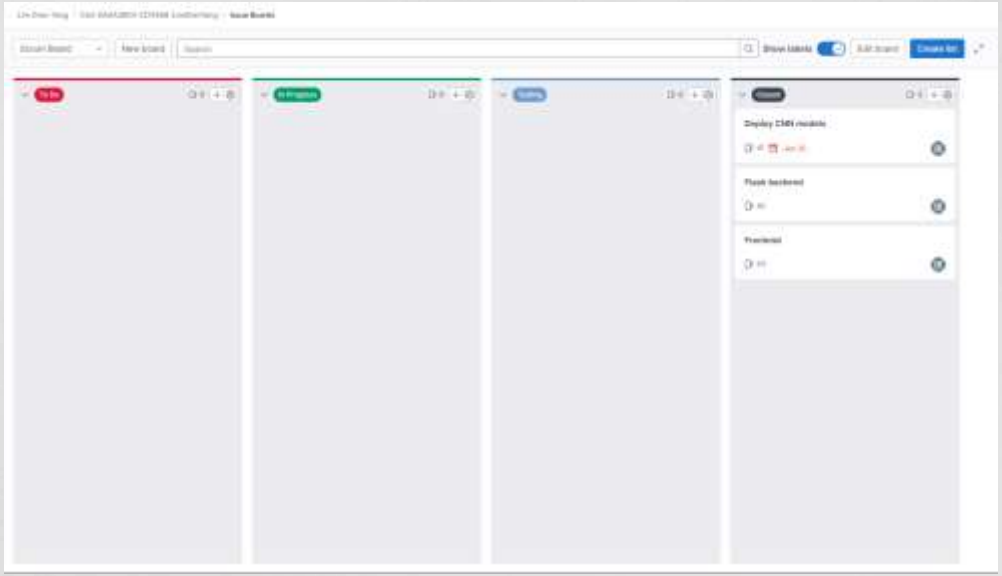


Error Monitoring (Sentry)

To continuously keep track of errors after deployment.
By integrating Sentry with GitLab, we can easily see, track, and resolve errors



Scrum Board



A Scrum board provides a visual representation of the tasks in a project, facilitating transparency, collaboration, and accountability among team members. It helps teams organize, prioritize, and track their work efficiently, leading to increased productivity and adaptability in Agile development processes.

Branches

I have created branches for each new feature I implemented but the main ones are

1. Main
2. Client
3. Flask-backend
4. CICD

Main

Protected branch, no code breaking changes should occur here

Client

Frontend development

Flask-backend

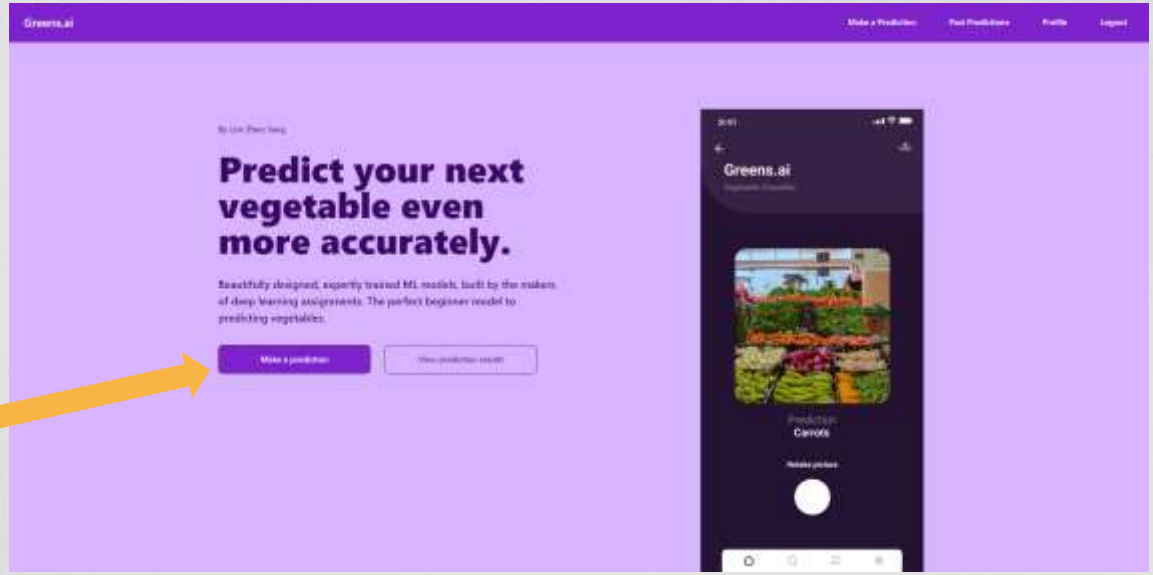
Backend development

CICD

Creating the CI/CD pipeline for GitLab to execute

Home Page

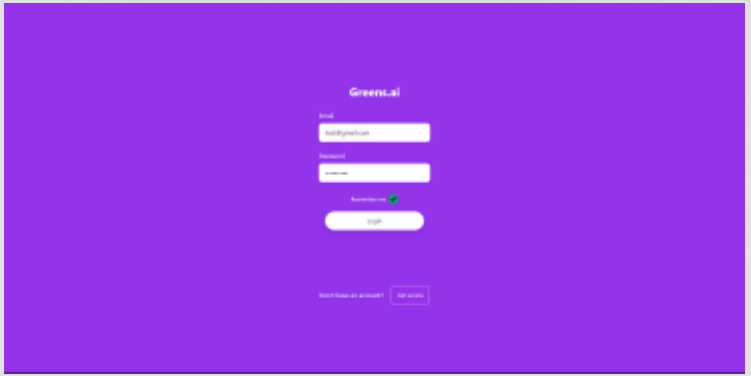
Link to prediction page



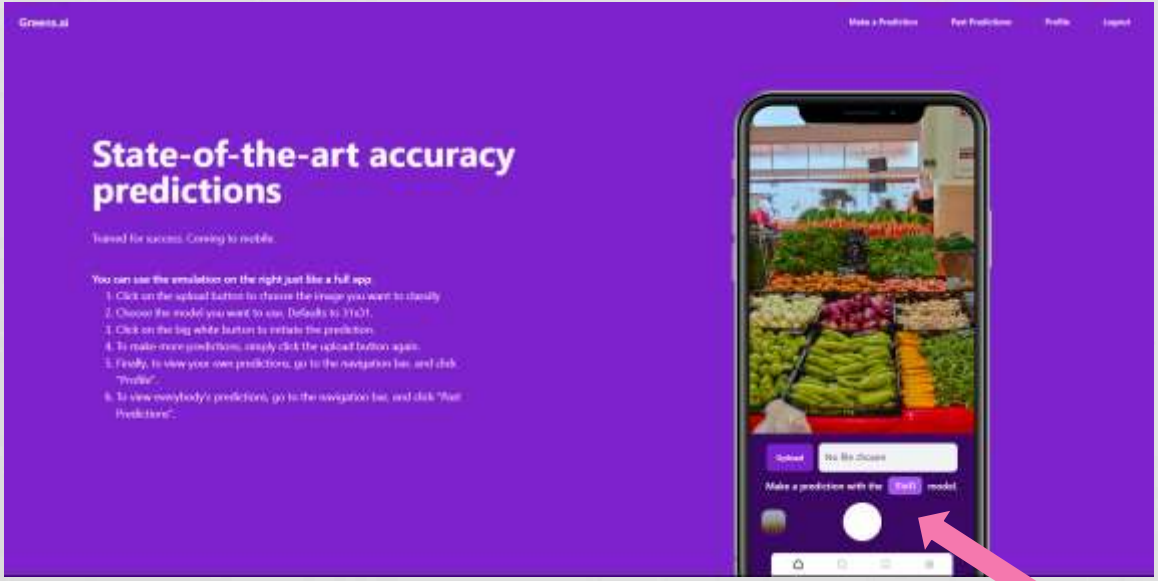
Sign Up and Log In

Sign Up

Log In



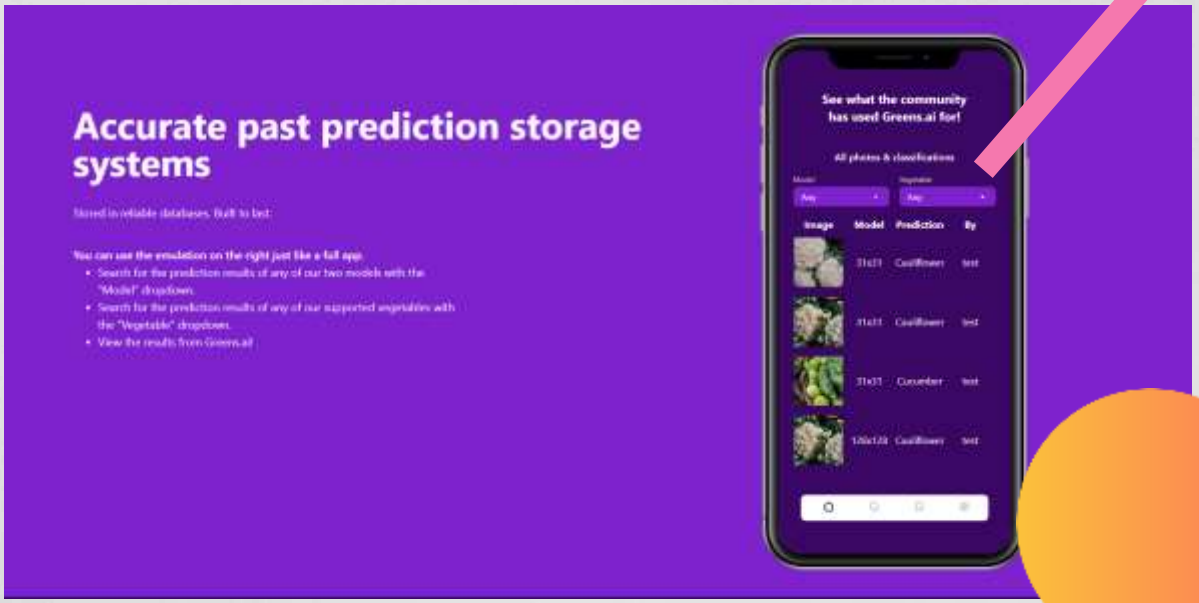
Prediction Page



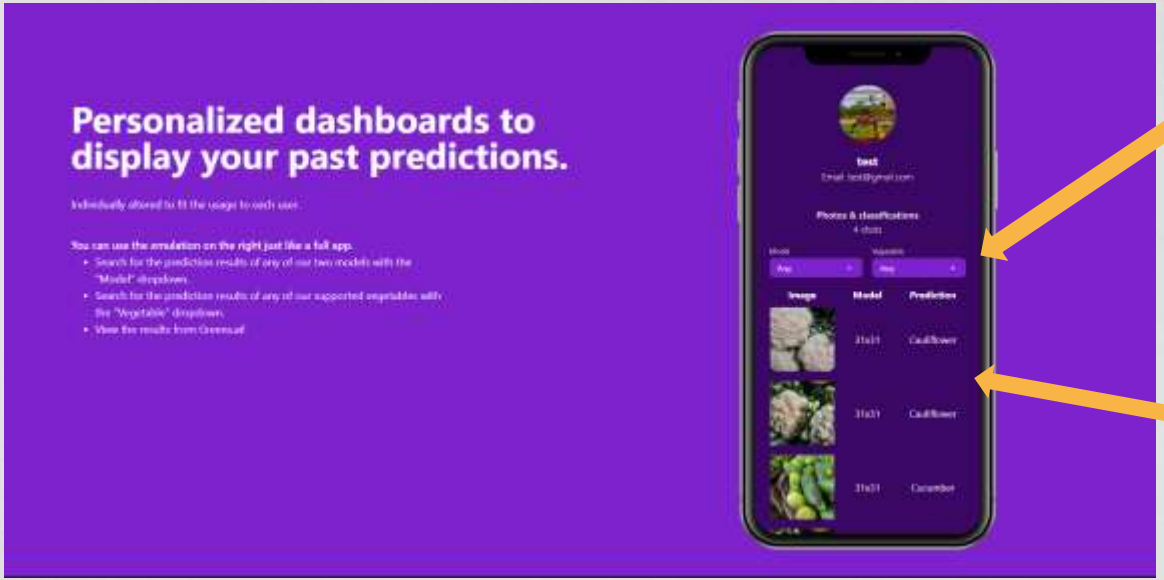
Create prediction from this phone emulator

Past predictions page

Prediction history of all users



Profile page

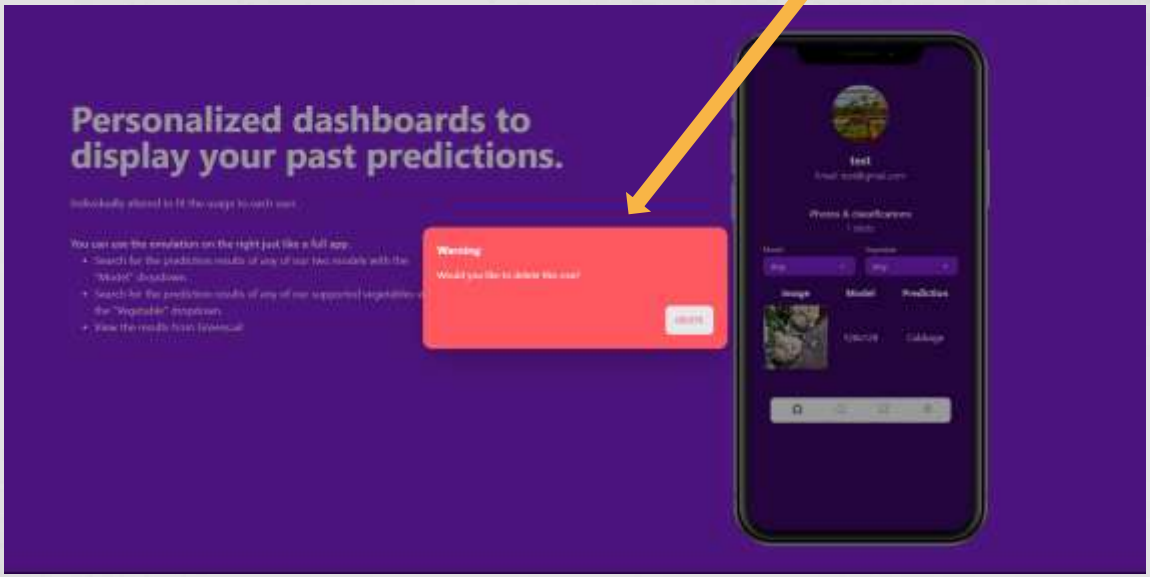


Filter and search past predictions

Display past predictions

Profile page

Clicking on a row opens this pop up



Next steps

Model Development

From the continuous collection of more images, we can train our model to improve it.

Any issues such as low model accuracy can be a detected by user feedback.

WebApp Development

Errors arising after deployment detected by Sentry can be quickly tracked by development.

Thanks!