C242-PS469

bangkit

Product-Based Capstone

OptiLens

See the Future Clearly

2024

C242-PS469

bangkit

Our Team

Mobile Development - Team

Sukma Wati

Machine Learning - Team

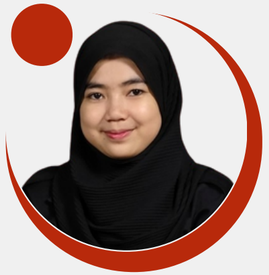
Abd Rahman Wahid

Salsabila Putri

Muhammad Hidayatul Fadillah

Cloud Computing - Team

Ahmad Faisal



Galbi Nadifah

OptiLens
See the Future Clearly

NEXT

C242-PS469

bangkit

Background

Cataracts and refractive errors (such as myopia, hyperopia, and astigmatism) are significant contributors to visual impairment worldwide. According to the 2019 **WHO report**, over 2.2 billion people are affected by vision problems, with cataracts being the leading cause of blindness. Despite advances in eye care, many individuals in underserved communities lack access to timely diagnosis and treatment due to the high cost and limited availability of healthcare facilities. Early detection and regular monitoring are crucial in preventing severe vision loss, yet these remain out of reach for many. With increasing **global awareness** of eye health, there is a pressing need for accessible, affordable, and accurate solutions that empower individuals to take control of their vision care.

OptiLens
See the Future Clearly

NEXT

C242-PS469

bangkit

Reason

The idea of creating OptiLens stems from the urgent need to address the challenges faced by individuals with limited access to ophthalmological resources. Inspired by technological advancements, we believe that machine learning and image processing can revolutionize eye health management. OptiLens intrigued us because it provides a practical solution to improve eye care accessibility while maintaining medical relevance.

By integrating cataract detection through advanced algorithms and manual refractive clarity assessments, our application bridges a critical gap in eye health diagnostics. Our goal is to make early diagnosis and vision monitoring accessible to vulnerable communities, ultimately reducing preventable blindness and empowering individuals to take proactive steps in managing their eye health.

Existing Result

C242-PS469

bangkit

Previous research and innovations have tried to utilize technologies such as Machine Learning and cloud-based applications to detect eye diseases, including cataracts. Some examples are:

Mobile Vision Health Apps

Several mobile apps have provided image-based diagnosis for early detection. However, many only focus on a single feature, such as cataract detection without providing additional insights, such as educational articles or manual refraction prediction.

OptiLens
See the Future Clearly

NEXT

C242-PS469

bangkit

Optimal Machine Learning Model

- Using CNN architecture with data augmentation enriches the variety of datasets, making the model more adaptive to different image conditions.
- The model is optimized using early stopping and hyperparameter tuning to achieve high accuracy without overfitting.

Cloud Computing

- Using CNN architecture with data augmentation enriches the variety of datasets, making the model more adaptive to different image conditions.
- The model is optimized using early stopping and hyperparameter tuning to achieve high accuracy without overfitting.

Mobile development

- The UI/UX prototype was designed using Figma to provide a simple and interactive user experience.
- The application development is based on Android Studio, supporting direct image upload from camera or gallery.

Why Choose a Specific Implementation/Improvement?

- Cost Efficiency
- Ease of Replication
- Transparency

OptiLens
See the Future Clearly

Implementation & Reasoning

NEXT

MACHINE LEARNING
Result

C242-PS469

bangkit

DATASET PREPARATION

- The dataset was categorized into three classes: Immature, Mature, and Normal. The data is divided into training, validation, and testing.
- Data augmentation was performed to enrich the variety of training data using ImageDataGenerator.

MODEL ARCHITECTURE

- Use TensorFlow to build Model
- Convolutional Neural Network (CNN)
- The model uses Sequential API with layers

MODEL TRAINING

- Optimizer: Adam.
- Loss Function: Categorical Crossentropy.
- Metrics: Accuracy.
- Callback: Early stopping is used to stop training when the model does not improve.

TRAINING RESULT

```
Epoch 1/20
/usr/local/lib/python3.10/dist-packages/keras/src/trainers/data_adapters/py_dataset_adapter.py:122: UserWarning: Your
  self._warn_if_super_not_called()
83/83 ━━━━━━━━━━━━━━━━━━━━ 118s 1s/step - accuracy: 0.5227 - loss: 0.9152 - val_accuracy: 0.9149 - val_loss: 0.1781
Epoch 2/20
83/83 ━━━━━━━━━━━━━━━━━━━━ 113s 1s/step - accuracy: 0.8841 - loss: 0.2865 - val_accuracy: 0.9868 - val_loss: 0.0612
Epoch 3/20
83/83 ━━━━━━━━━━━━━━━━━━━━ 0s 1s/step - accuracy: 0.9434 - loss: 0.1551
 Akurasi Validasi telah mencapai: 0.9905, menghentikan pelatihan.
83/83 ━━━━━━━━━━━━━━━━━━━━ 140s 1s/step - accuracy: 0.9434 - loss: 0.1552 - val_accuracy: 0.9905 - val_loss: 0.0461
```

OptiLens
See the Future Clearly

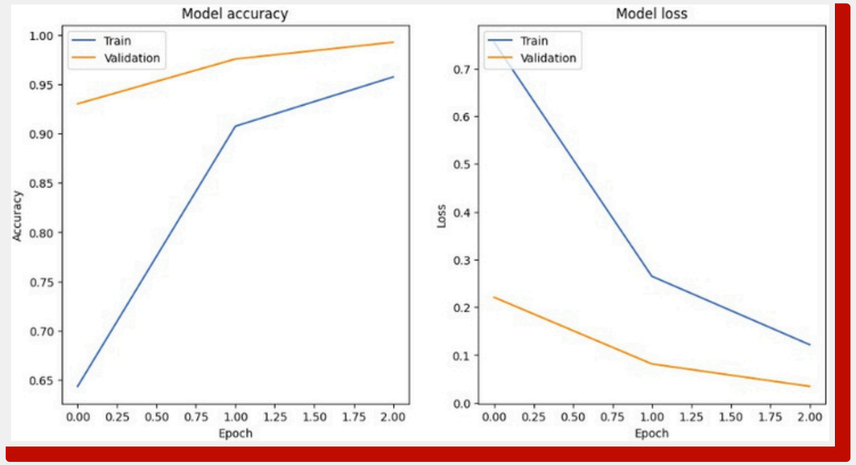
NEXT

VISUALIZATION OF RESULTS & MODEL SUMMARY

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 126, 126, 32)	896
max_pooling2d (MaxPooling2D)	(None, 63, 63, 32)	0
conv2d_1 (Conv2D)	(None, 61, 61, 64)	18,496
max_pooling2d_1 (MaxPooling2D)	(None, 30, 30, 64)	0
conv2d_2 (Conv2D)	(None, 28, 28, 128)	73,856
max_pooling2d_2 (MaxPooling2D)	(None, 14, 14, 128)	0
flatten (Flatten)	(None, 25088)	0
dense (Dense)	(None, 128)	3,211,392
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 64)	8,256
dropout_1 (Dropout)	(None, 64)	0
dense_2 (Dense)	(None, 3)	195

Total params: 9,939,275 (37.92 MB)
Trainable params: 3,313,091 (12.64 MB)
Non-trainable params: 0 (0.00 B)
Optimizer params: 6,626,184 (25.28 MB)

Model Architecture: CNN model structure with a total of 9.9 million parameters, consisting of convolution, pooling, and fully connected layers.



Loss graphs and accuracy: The graph shows an increase in accuracy and a decrease in loss for training and validation, indicating the model is learning well.

C242-PS469

bangkit

OptiLens

Use Technology

TensorFlow

Google Colab

OpenCV

PIL (Python Imaging Library)

NestJS

PostgreSQL

TypeScript

Google Cloud

Docker

Figma

Android Studio

Kotlin

OptiLens
See the Future Clearly

NEXT

C242-PS469

bangkit

CLOUD COMPUTING

Plan

1. Build a modular and scalable backend architecture with NextJS and TypeScript-based Framework
2. Employing PostgreSQL for reliable and high-integrity data management
3. Leveraging Google Cloud Platform for deployment and scaling
4. Isolated application environtments and ensuring consistent with Docker
5. And Create some APIs for integration to Mobile Development

Budget & Expenses (GCP Cloud) :
$50

OptiLens
See the Future Clearly

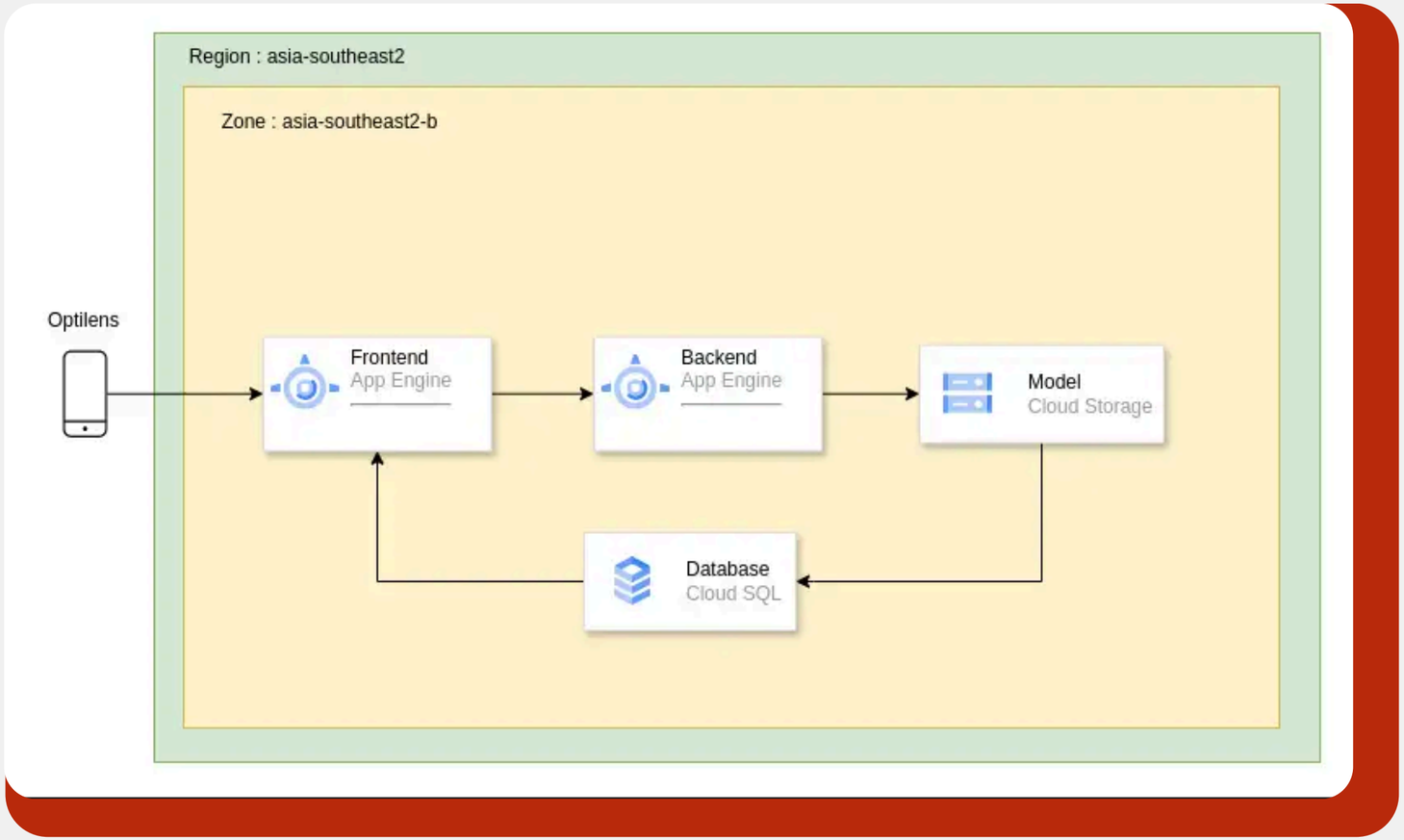
NEXT

C242-PS469

CLOUD COMPUTING

Cloud Architecture

bangkit



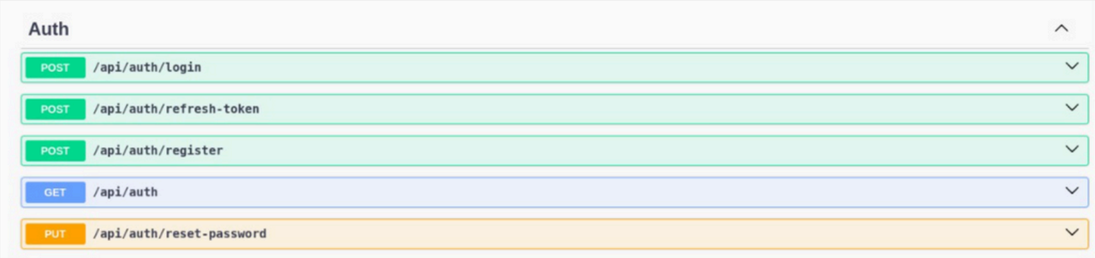
OptiLens
See the Future Clearly

NEXT

C242-PS469

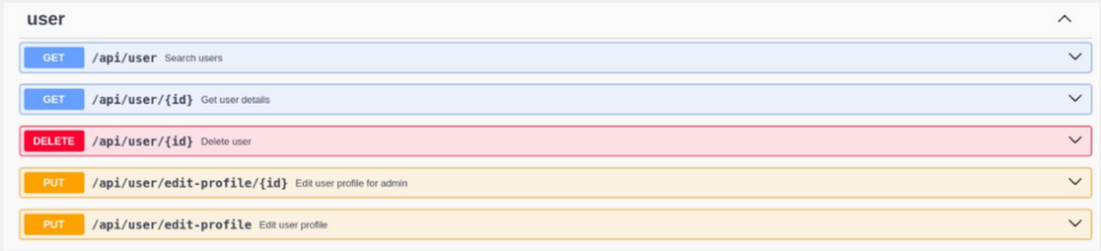
bangkit

Auth

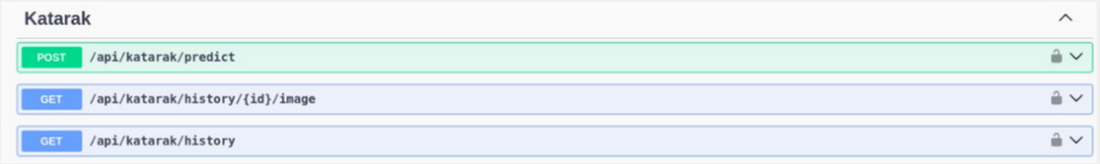


Artikel

User



Katarak



API from CLOUD COMPUTING

OptiLens
See the Future Clearly

NEXT

C242-PS469

bangkit

PROTOTYPING THE UI/UX

CREATE AND SET-UP THE PROJECT ON GITHUB

IMPLEMENTING THE PROTOTYPE INTO THE PROJECT

DESIGNING ARCHITECTURE COMPONENTS

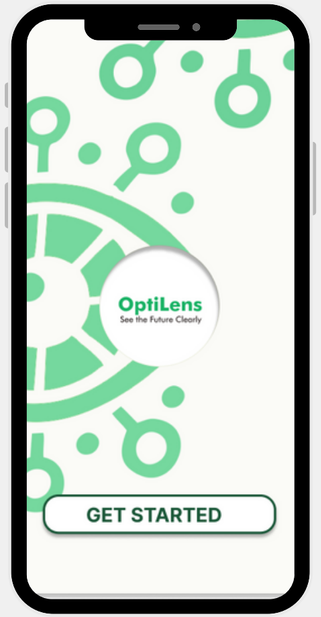
APPLY INSTRUMENTAL AND UNIT TESTING, BUG FIXING FOR FINAL APP

MOBILE DEVELOPMENT

Plan

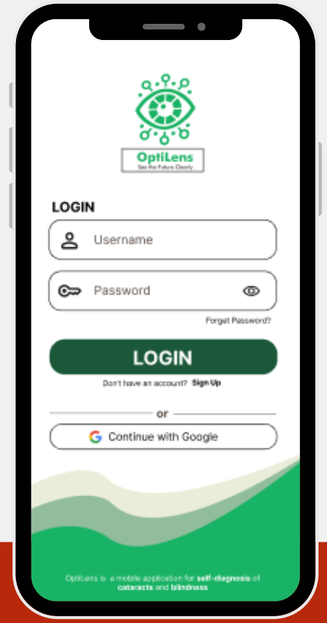
OptiLens
See the Future Clearly

NEXT

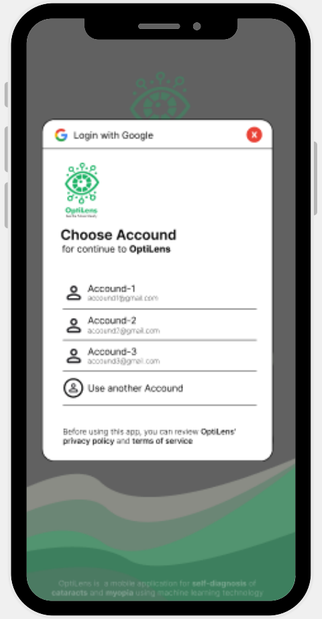


Welcome Guide

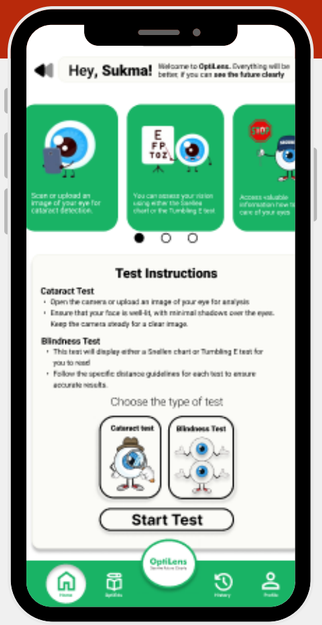
Login



OAuth



SignUp



Main Menu

OptiLens
See the Future Clearly

NEXT

Mobile Apps Design

C242-PS469

Mobile Apps Design

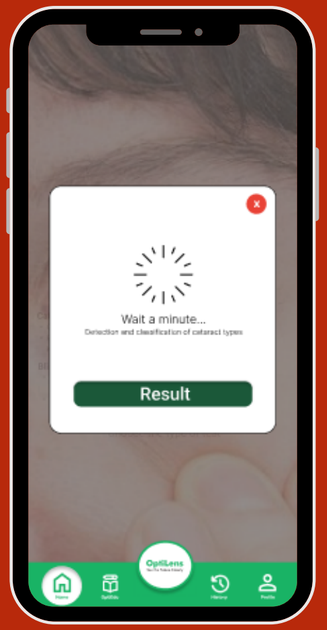
bangkit

Cataract Select

Upload or take image

Take Image

Detecting



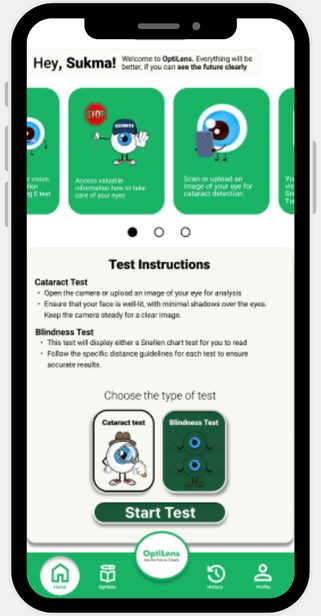
Result

OptiLens
See the Future Clearly

NEXT

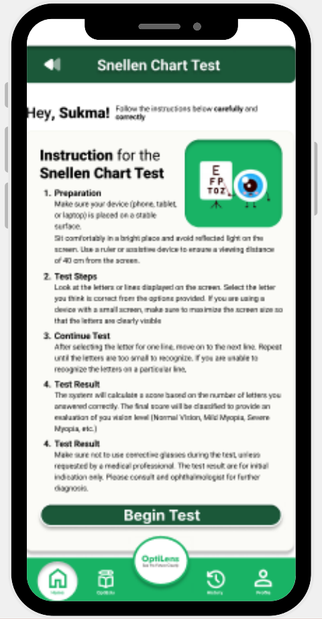
C242-PS469

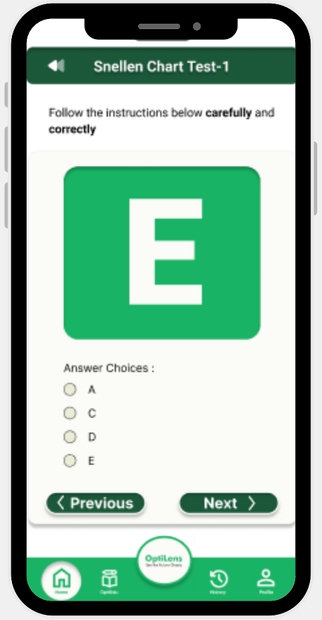
bangkit



Blindness Select

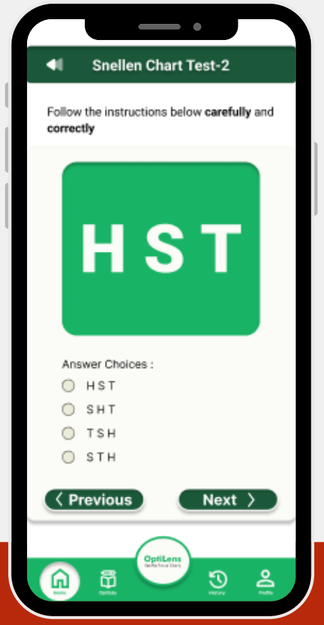
Instruction

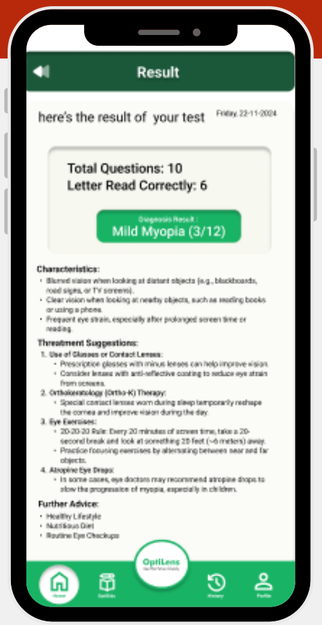




Test-1

Test-2...N





Result

OptiLens
See the Future Clearly

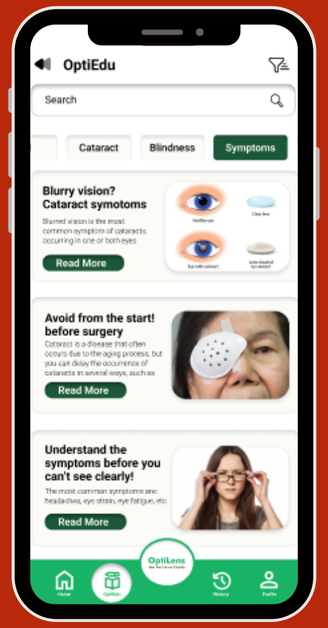
Mobile Apps Design

NEXT

C242-PS469

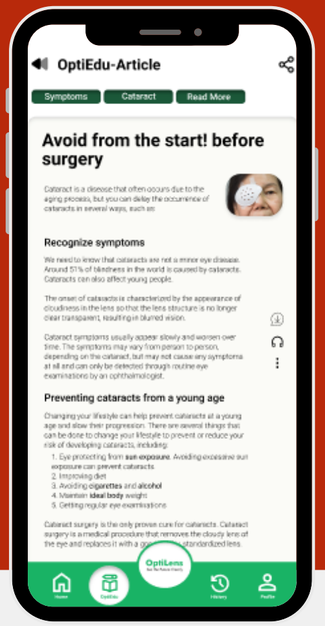
Mobile Apps Design

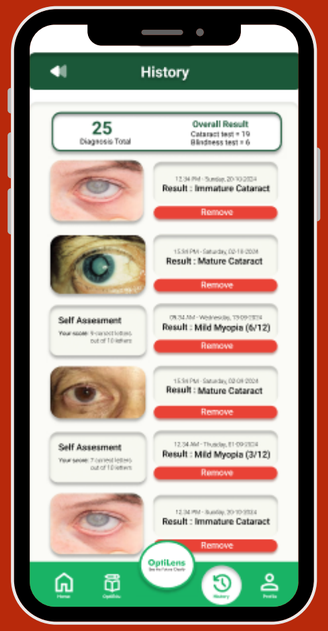
bangkit



OptiEdu

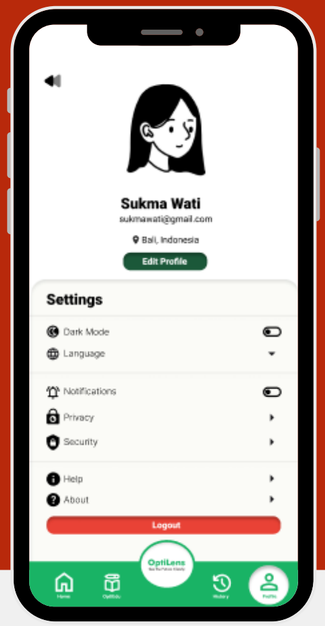
Select Article





User History

User profile



Setting Profile

OptiLens
See the Future Clearly

NEXT

C242-PS469

Person-Hours

bangkit

Machine Learning - Team

Abd Rahman Wahid

68 Hours

Salsabila Putri

68 Hours

Muhammad Hidayatul Fadillah

50 Hours

- **Optimal Data Preparation:** We structured the dataset into three categories (Immature, Mature, Normal) and applied data augmentation to enhance data variety and improve model robustness against overfitting.
- **Efficient Model Architecture:** Our CNN architecture, featuring convolutional, pooling, and dropout layers, effectively extracted features while minimizing overfitting risks.
- **Focused Training and Evaluation:** Using the Adam optimizer and categorical crossentropy loss, we achieved efficient convergence. Callbacks, like early stopping, ensured training stopped at optimal performance. Visualization of accuracy and loss trends facilitated overfitting detection.
- **Ease of Inference:** The model processes uploaded or webcam-captured images with high accuracy, making predictions seamlessly.
- **Cloud Integration:** Converting the trained model to .h5 format allowed smooth deployment on cloud servers.

OptiLens
See the Future Clearly

NEXT

C242-PS469

Person-Hours

bangkit

Cloud Computing - Team

Ahmad Faisal

60 Hours

Galbi Nadifah

60 Hours

On the backend, we used the NestJS framework written in TypeScript to build a modular and extensible application architecture. The database used is PostgreSQL, which is reliable for high-integrity data management. To support deployment and scaling management, we utilize Google Cloud Platform and use Docker to package the application in an isolated and consistent environment.

Mobile Development

Ahmad Faisal

40 Hours

The app development process starts with designing the UI/UX prototype in Figma, translating it into functional components in Android Studio using Kotlin. The project is set up on GitHub for version control, using modern architectures like MVVM for scalability.

OptiLens
See the Future Clearly

NEXT

C242-PS469

bangkit

Thank You

OptiLens
See the Future Clearly

2024