Project Brief - Bangkit 2023

Baksara

Product Capstone

Team ID : **C23-PS209**

Selected Themes/Case : Education ▾

Mentor Name :
1. **Sayed Khaidir Ali (Cloud Computing),** Mentoring done on May 26, 2023 9:00 AM and Jun 1, 2023 10:00 PM
2. **Achmad Ilham (Mobile Development),** Mentoring done on May 29, 2023 5:00 PM

Member Name :

Member ID	Member Name	University	Path	Status
M351DKX3883	Theofilus Arifin	Universitas Surabaya	ML	Active
M038DSX1525	Achmad Nashruddin Riskynanda	Institut Teknologi Sepuluh Nopember	ML	Active
A351DKX3904	Hans Wirjawan	Universitas Surabaya	MD	Active
A351DKX3903	Rony Hartono Irawan	Universitas Surabaya	MD	Active
C227DSX0761	Glenn Steven Santoso	Universitas Kristen Petra	CC	Active
C038DSX0839	Okyan Awang Ramadhana	Institut Teknologi Sepuluh Nopember	CC	Active

Backgrounder:

1. Machine Learning:
This machine learning system that can recognize and transliterate ancient Javanese script (aksara) from images was developed with these main steps. First step is collecting a dataset of 28 classes that consist of 4067 images of aksara from various sources and formats. After collecting the dataset, we built a machine learning model using transfer learning with MobileNetV2 architecture, a lightweight and efficient

neural network for image classification. After developing the model, we trained and evaluated the machine learning model to obtain the best result on testing accuracy, 94.7%. Not only classification, we also built a segmentation pipeline for segmenting each aksara from the input image, using techniques such as adaptive thresholding, contour detection, and bounding box extraction since a few aksara has its characteristic. After this step is done, we built a transliteration pipeline to transliterate the results from the classification model to the correct Latin script, using a mapping table and rules based on the Javanese script system.

2. Mobile Development: We create the application with MVVM architecture and XML layout. The application features and UI/UX is based on our Figma Project. We are using the Minimum Viable Product concept for implementing the features. The MVP features are Gamified Learning Class, consisting of 3 learning steps (read Aksara, writing Aksara using canvas, and multiple choice), Aksara Document Scanner, and Aksara Library. For our Gamified Concept we added a badge feature when users complete our learning module. We have completed the MVP features. In addition, we create more features like Aksara challenges, articles, problem reports in our application and profile information.

3. Cloud Computing: We have developed multiple API services. The first API service, built with Node.js featuring a GraphQL interface, serves as the main service connected to SQL instance. It handles user data and various functionalities within the application. The second API, built with Express.js, provides translation services from Latin to Aksara, allowing users to convert text into desired script. Lastly, we have an API that acts as a container for the Machine Learning Model, managing Aksara Jawa classification and image recognition. These APIs collectively power the Baksara Application. We deployed our application in Google Cloud Platform using Docker image with Cloud Run. We also integrated the CI / CD pipeline using Cloud Build with push triggers on the main branch of the repository. Any detected changes to a valid commit version will immediately make revisions.

Project Status:

100% Completed based on Project Plan

Screenshots/Demo Video:
https://youtu.be/LRupPblr0nk

Dataset Link:
https://github.com/baksara-id/baksara_dataset

Deployed Link:
1. Transliteration API → An API service that provides Latin to Aksara translation. It is deployed on Cloud Run and can be accessed using the following URL:

Project Brief - Bangkit 2023

https://aksara-api-v4-qohpfhypea-et.a.run.app
2. Core Data System API → This API caters to the database-related requirements of the application.
https://graphql-api-qohpfhypea-an.a.run.app
3. Machine Learning Model API → This API is utilized for Aksara Jawa classification and image recognition purposes.
https://python-api-qohpfhypea-et.a.run.app
4. Connect the Core Data System API with Apollo Sandbox:
https://studio.apollographql.com/sandbox/explorer/

Github Repo Link:
https://github.com/orgs/Baksara-id/repositories

10-Min Video Presentation Link:
https://youtu.be/gO9JOdMmbdQ

Slide Presentation Link(s):
https://s.id/baksara-final-presentation

Mentoring Remark(s), if any:
1. API documentation should be completed before you start developing.
2. Make sure the core business is working properly if you want to enhance by adding some supporting features