POC Details

Participants – Gianmarco, Felix

Deadline – 4 weeks

Budget – 8800 USD

1. Detailed timeline for each use case story

1) AI part(Gianmarco)

- Classification model pre-train
  Data collection & preprocessing – 1 day
  Model training – 1 day
  Feedback & fix – 1 day
  Total – 3 days
- Input file preprocessing and automatic classification
  Input file preprocessing – 1 day
  Automatic classification & Feedback – 1 day
  Total – 2 days
- Input data vector embedding & save
  Input data embedding – 1 day
  Vector data setup and management – 1 day
  Feedback & fix – 1 day
  Total – 3 days
- Model automatic update for new category
  Data preparation - 1 day
  Model automatic updata – 1 day
  Feedback & fix – 1 day
  Total – 3 days
- Model fine tune with sample document
  Fine-tuning – 1 day
  Validation & feedback – 1 day
  Total – 2 days
- API implementation on Cloud(AWS)
  Infrastructure – 1 day
  Deploy - 1 day
  Feedback & fix – 1 day
  Total – 3 days

2) Fullstack part(Felix)

- Design
  Design Demo – 1 day
  Feedback and fix – 1 day
  Total – 2 days

- Upload and Result display(story 1-3)
  Upload & Result UI – 2 days
  Amazon S3 bucket uploading, MongoDB setting Up using Next.js API – 2 days
  Feedback and fix – 1 day
  Total – 5 days
- Input New Category(story 4)
  Input new category – 1 day, it will include input new category and upload file for AI model
- Management(story 5-7)
  UI – 2 days, it will include showing all lists with metadata and preview function, open in new tab, download and delete function, Meta Data View UI
  Fetching Preview data from Amazon S3 and implementing functions – 2 days
  Feedback and fix – 1 day
  Total – 5 days

3) Integration(Gianmarco & Felix)

- Integration AI and Fullstack
  Total - 2 days
- Feedback and fix
  Total – 2 days

2. Technology Stack and Architecture foundation

- Framework - Tensorflow/Keras
- Cloud – AWS
- Fullstack - Next.js, MongoDB
- Hosting – Vercel
- File Service - Amazon S3 bucket
- LLMOps - Akira AI
- Model Eval - Langchain or Akira AI
- Foundation Model - Open AI
- Database - ChromaDB / Pinecone
- Runtime – Langchain
  • Your current project is Demo version, so we have to consider scailability.
  Because Langchain has full function for LLM agent, I suggest to use Langchain.
  Of course, we can use pre-trained LLM classification models in Hugging Face, OctoML so on. But in this case, we will get difficulty to expand it.
  • Langchain support ChromaDB as default database and can use free, that's why I suggest to use ChromaDB as vector database.
  Of course if you want, we can use Pinecone for vector database because Langchain supports Pinecone also.

3. Price Quote for each use case with a timeline breakdown

1) AI part(Gianmarco)

- Classification model pre-train – 800
- Input file preprocessing and automatic classification - 600
- Input data vector embedding - 800
- Model automatic update for new category - 800

- Model fine tune with sample document - 600
- API implementation on Cloud(AWS) – 800
- Total Budget - 4400

2) Fullstack part(Felix)

- Design – 500
- Upload and Result display - 1300
- Input New Category – 300
- Management – 1300
- Total Budget – 3400

3) Integration(Gianmarco & Felix)

- Integration AI and Fullstack
- Feedback and fix
- Budget
  1000(Gianmarco – 500, Felix - 500)

4) Total Budget

- Gianamrco - 4900
- Felix – 3900
- Total - 8800