ARCHITECTURE DESIGN

Siddharth Agarwal

Shreyansh

Garvit Gupta

Priet Ukani

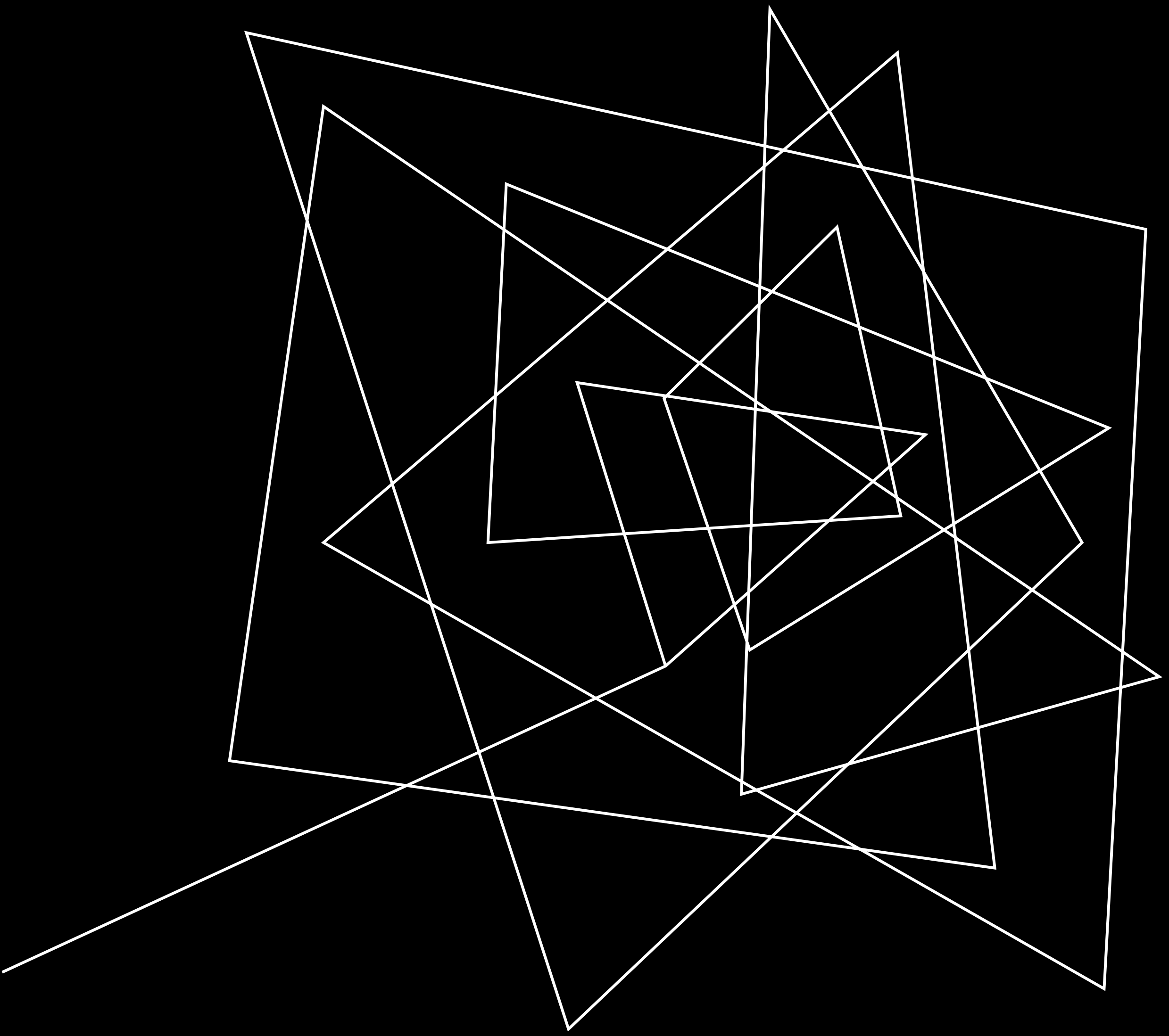
AGENDA

Cloud Models

Timeline

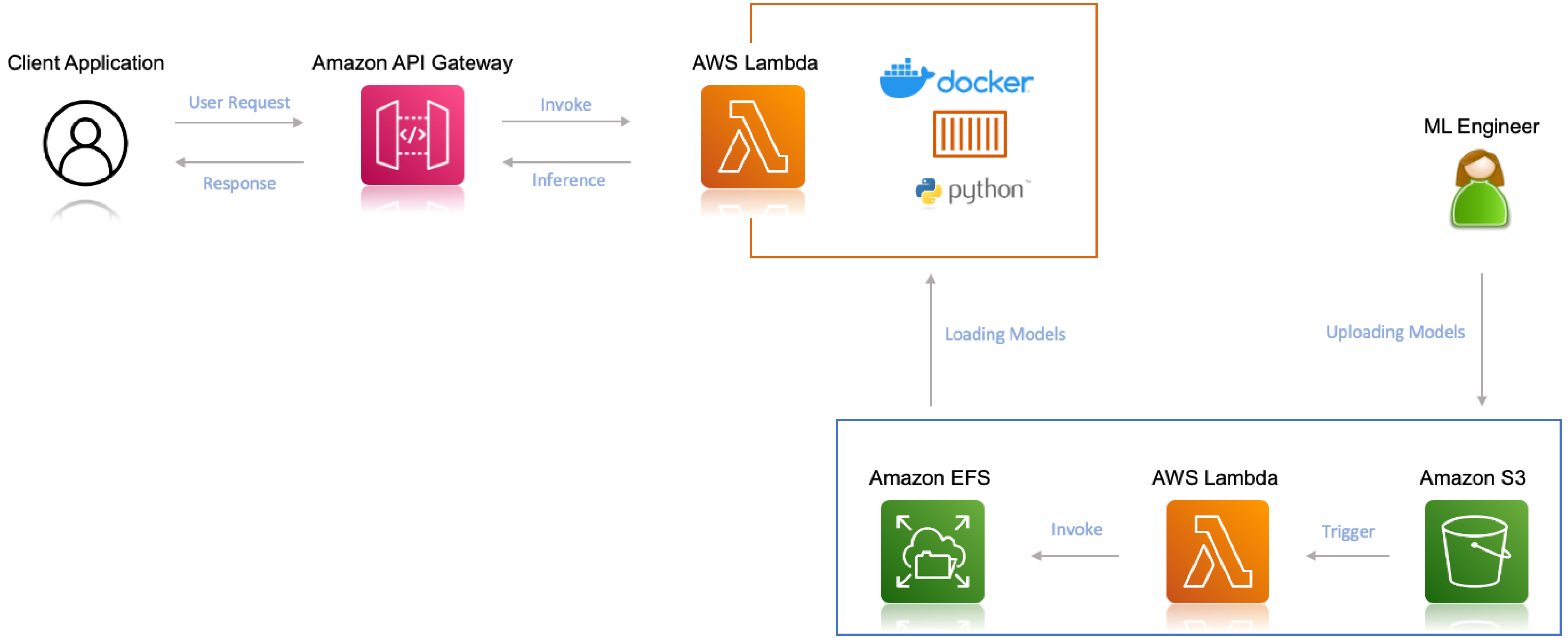
INTRODUCTION

We plan to figure out a cost effective, easy to use cloud service provider for deploying the ML model and provide an approximate timeline of the project.



CLOUD MODELS

AMAZON LAMBDA + S3 + EFS



AMAZON LAMBDA + S3 + EFS

- Simple and cost-effective. Charges will be applicable as per inferences.

- Due to the complexity of deploying a pre-trained model on Sagemaker and its high cost, we would prefer to deploy on lambda + S3 now, and when monitoring and more scaling is required, we can add Sagemaker to our architecture.

- Pros: Simple & best suitable for our current situation, cost-efficient (as per inference) & easy to handle. (We can switch to Sagemaker whenever required)

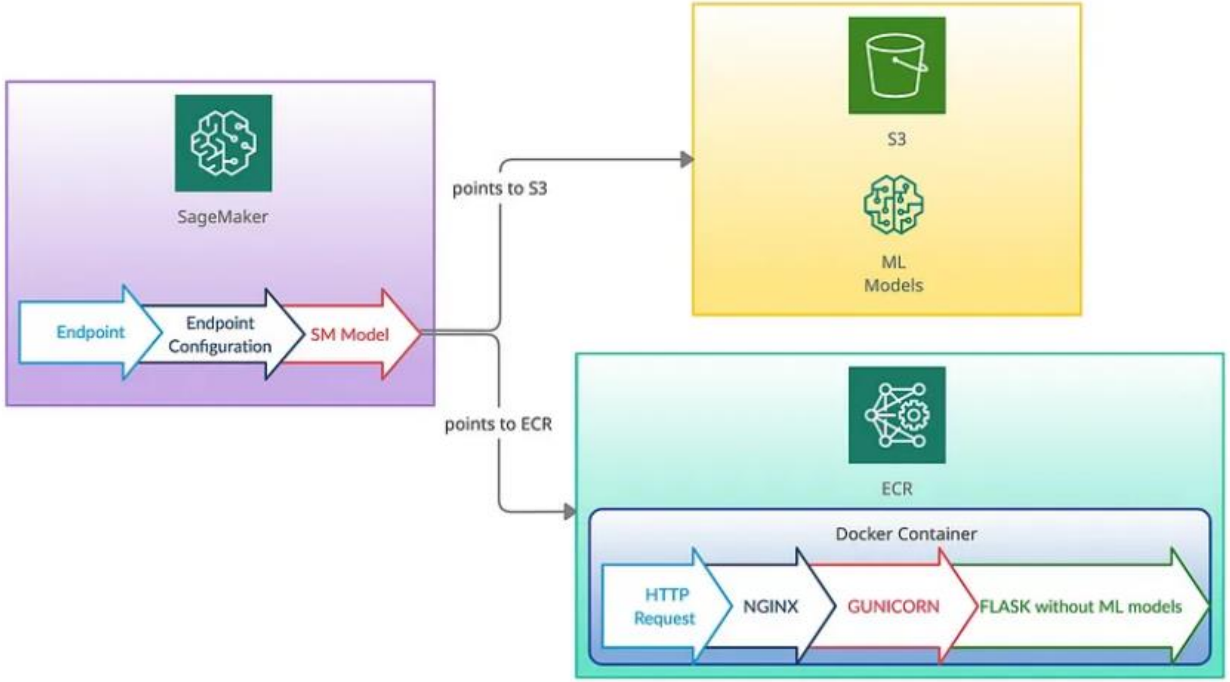
- Cons: Less automatic MLOps functionalities available, also less scalable than sagemaker. Manual handling of the retraining process

GOOGLE CLOUD FUNCTION

- Step 1- Training the model on your local machine.
- Step 2- Creating a new Google Cloud Project.
- Step 3- Storing the pre-trained model in a Google Cloud Storage.
- Step 4- Writing the Google Cloud Function for deployment
- Quite similar to lambda function architecture.
- Pros: simple and easy to use like lambda with S3 & cost efficient
- Cons: Will need manual implementation to retrain model.
- Others Cloud Services by Google: EC2, Google App engine, Vertex AI etc.

AMAZON SAGEMAKER

- Amazon Sagemaker with other services of AWS like S3 Buckets, AWS ECR, AWS lambda & api Gateway.



AMAZON SAGEMAKER

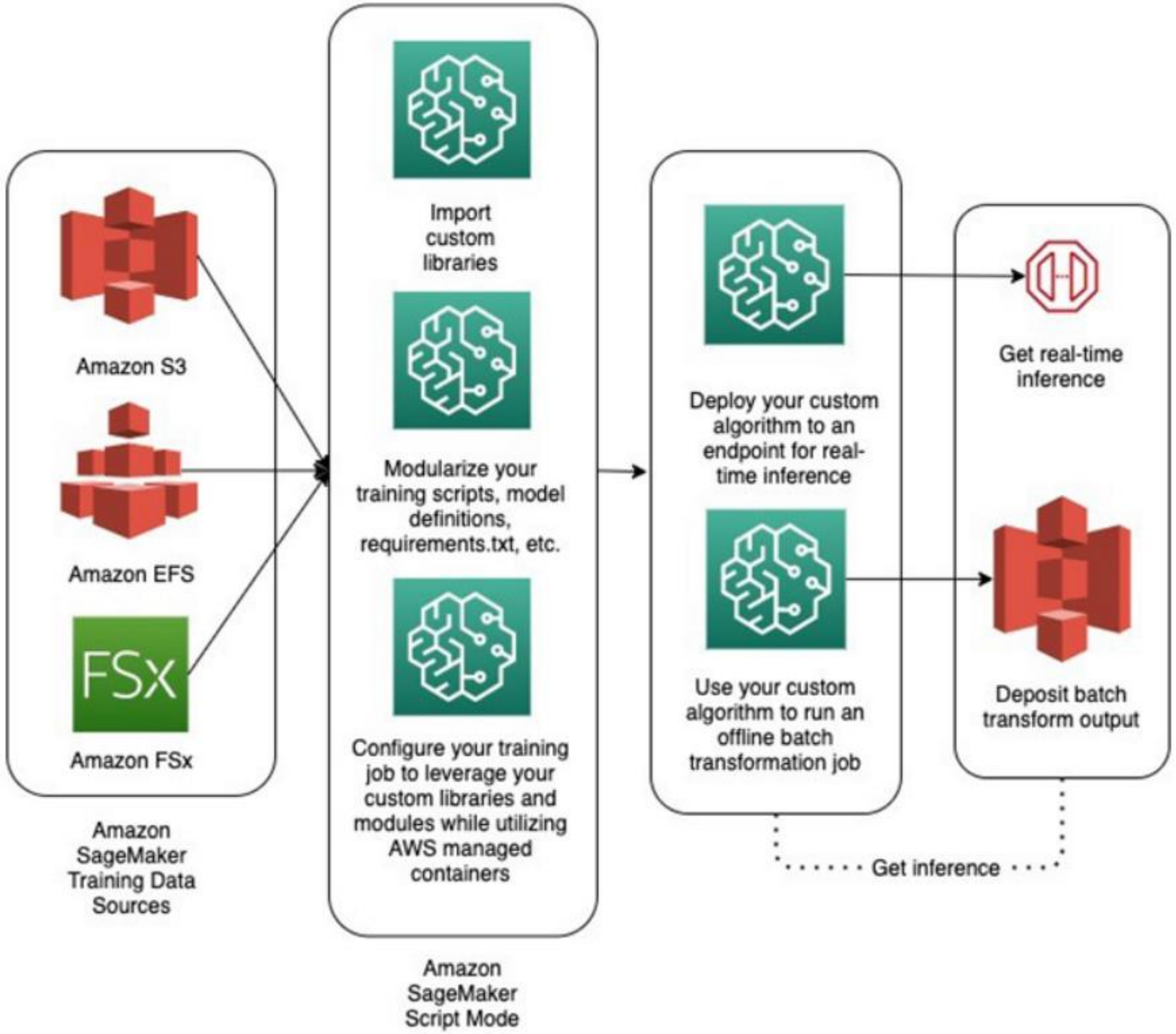
- Amazon SageMaker is a fully managed machine learning service. It helps data scientists and developers to prepare, build, train, and deploy high-quality machine learning (ML) models quickly.

- It provides an integrated Jupyter authoring notebook instance to easily access your data sources for exploration and analysis.

- Pros: automatic MLOps functionalities, Make monitoring & analysis of ML models easy (but we don’t have to delve into model monitoring right now, we have to focus on how to provide users access of the service model).

- Cons: Higher cost than Lambda function architecture & a little bit complicated while deploying pretrained model on sagemaker endpoint instead of building and training model on sagemaker and then deploying it on the endpoint.

AMAZON SAGEMAKER SCRIPT MODE


Amazon S3
Amazon EFS
FSx
Amazon FSx
Amazon SageMaker Training Data Sources
Import custom libraries
Modularize your training scripts, model definitions, requirements.txt, etc.
Configure your training job to leverage your custom libraries and modules while utilizing AWS managed containers
Amazon SageMaker Script Mode
Deploy your custom algorithm to an endpoint for real-time inference
Use your custom algorithm to run an offline batch transformation job
Get real-time inference
Deposit batch transform output
Get inference

AMAZON SAGEMAKER SCRIPT MODE

- Script mode enables you to write custom training and inference code while still utilizing common ML framework containers maintained by AWS. Script mode is easy to use and flexible.

- Pros: We can customize libraries we want to use, we can customize code to train-retrain model & we can also customize inference code by giving our own scripts. (quite similar to Sagemaker + S3 + ECR).

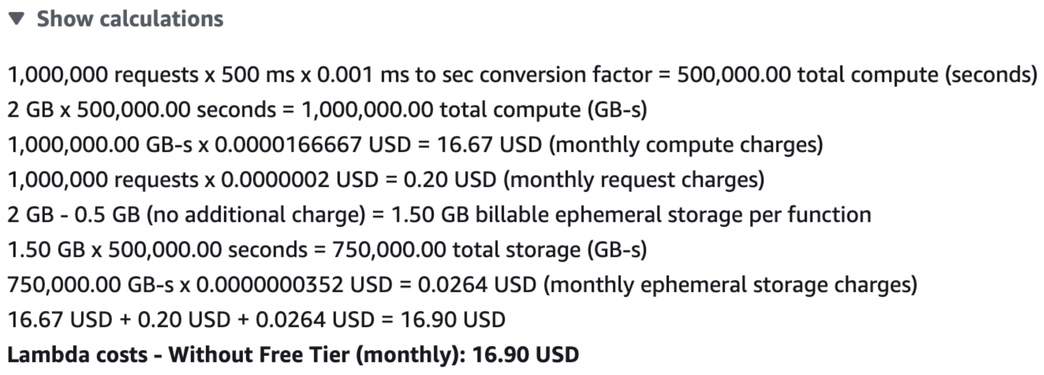
- Cons: Since Sagemaker is involved so all the previous cons are applicable here too.

OTHER ALTERNATIVES

- Other alternatives are Amazon EC2, Google App Engine, Vertex AI etc.
- Google App Engine (GAE) is a platform for building and hosting scalable web applications and mobile backends. It's a fully managed, serverless platform that allows developers to build applications in any programming language.
- Vertex AI is analogous to Sagemaker in AWS.
- Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, scalable computing capacity in the Amazon Web Services (AWS) Cloud

COSTING

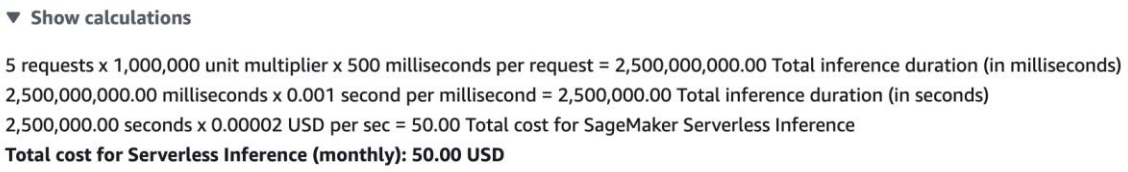
AWS Lambda (Without Free Tier)



▼ Show calculations

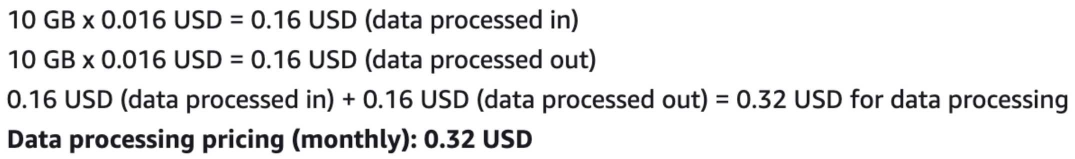
1,000,000 requests x 500 ms x 0.001 ms to sec conversion factor = 500,000.00 total compute (seconds)
2 GB x 500,000.00 seconds = 1,000,000.00 total compute (GB-s)
1,000,000.00 GB-s x 0.0000166667 USD = 16.67 USD (monthly compute charges)
1,000,000 requests x 0.0000002 USD = 0.20 USD (monthly request charges)
2 GB - 0.5 GB (no additional charge) = 1.50 GB billable ephemeral storage per function
1.50 GB x 500,000.00 seconds = 750,000.00 total storage (GB-s)
750,000.00 GB-s x 0.0000000352 USD = 0.0264 USD (monthly ephemeral storage charges)
16.67 USD + 0.20 USD + 0.0264 USD = 16.90 USD
Lambda costs - Without Free Tier (monthly): 16.90 USD

AWS SageMaker



▼ Show calculations

5 requests x 1,000,000 unit multiplier x 500 milliseconds per request = 2,500,000,000.00 Total inference duration (in milliseconds)
2,500,000,000.00 milliseconds x 0.001 second per millisecond = 2,500,000.00 Total inference duration (in seconds)
2,500,000.00 seconds x 0.00002 USD per sec = 50.00 Total cost for SageMaker Serverless Inference
Total cost for Serverless Inference (monthly): 50.00 USD



10 GB x 0.016 USD = 0.16 USD (data processed in)
10 GB x 0.016 USD = 0.16 USD (data processed out)
0.16 USD (data processed in) + 0.16 USD (data processed out) = 0.32 USD for data processing
Data processing pricing (monthly): 0.32 USD

TIMELINE

Milestone	Due Date	Release	Deliverable?
Draft temporary document for architecture & timeline	1/2/24	R1	Yes
Finalizing architecture	5/2/24	R1	Yes
Making the high level design for extension	20/2/24	R1	Yes
Deciding on tools to use based on design	23/2/24	R1	Yes
Distribution of implementation work	23/2/24	R1	No
Building a primitive version of the app	5/3/24	R1	No
Testing the primitive app for bugs or faults	10/3/24	R2	No
Building the final version of the app	20/3/24	R2	Yes
Reiterations & modifications	15/4/24	R2	Yes
Final extensive testing and fixing	18/4/24	R2	No
Deployment and Final release of the app	20/4/24	R2	Yes

LINKS TO REFER

https://course19.fast.ai/

https://developer.nvidia.com/blog/machine-learning-in-practice-deploy-an-ml-model-on-google-cloud-platform/

https://medium.com/geekculture/84af8989d065

https://aws.amazon.com/blogs/machine-learning/bring-your-own-model-with-amazon-sagemaker-script-mode/

https://calculator.aws/#/addService (For Cost Estimation)

THANK YOU

- PRIET, GARVIT, SIDDHARTH, SHREYANSH