Design Flow of ML Solution "Cardiovadcualr Disease Prediction" using Machine Learning on AWS
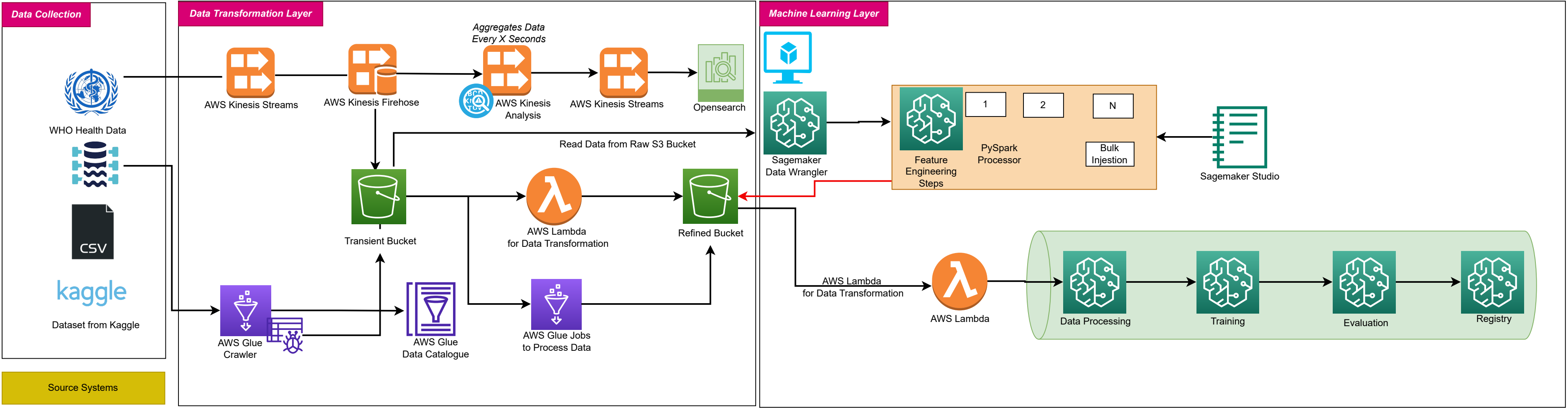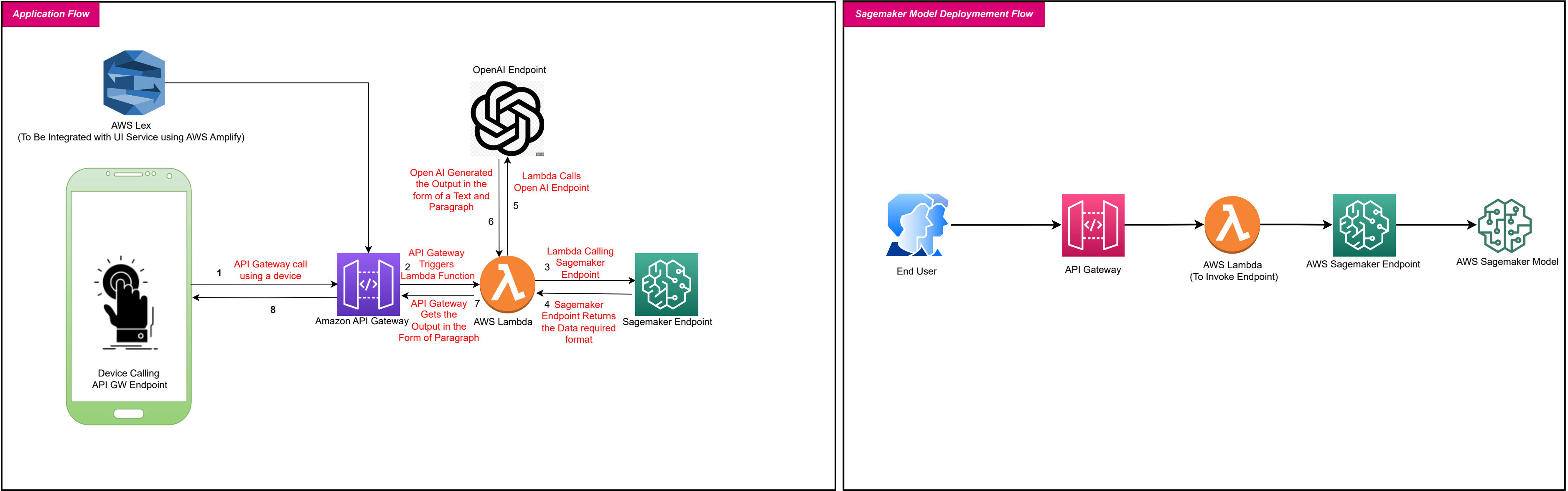
Data Collection and Transformation

ML Operations

Data Collection
- WHO Health Data
- CSV
- kaggle
- Dataset from Kaggle

Source Systems

Data Transformation Layer
- AWS Kinesis Streams
- AWS Kinesis Firehose
- AWS Kinesis Analysis — Aggregates Data Every X Seconds
- AWS Kinesis Streams
- Opensearch
- Transient Bucket
- AWS Lambda for Data Transformation
- Refined Bucket
- AWS Glue Crawler
- AWS Glue Data Catalogue
- AWS Glue Jobs to Process Data
- Read Data from Raw S3 Bucket

Machine Learning Layer
- Sagemaker Data Wrangler
- Feature Engineering Steps
- PySpark Processor: 1, 2, N
- Bulk Injession
- Sagemaker Studio
- AWS Lambda for Data Transformation
- AWS Lambda
- Data Processing → Training → Evaluation → Registry

Application Flow with the Help of APP SDK and AWS Services

Sagemaker Model Deployment Strategy

Application Flow

- AWS Lex (To Be Integrated with UI Service using AWS Amplify)
- Device Calling API GW Endpoint
- OpenAI Endpoint
- Amazon API Gateway
- AWS Lambda
- Sagemaker Endpoint

1 API Gateway call using a device
2 API Gateway Triggers Lambda Function
3 Lambda Calling Sagemaker Endpoint
4 Sagemaker Endpoint Returns the Data required format
5 Lambda Calls Open AI Endpoint
6 Open AI Generated the Output in the form of a Text and Paragraph
7 API Gateway Gets the Output in the Form of Paragraph
8

Sagemaker Model Deploymement Flow
- End User → API Gateway → AWS Lambda (To Invoke Endpoint) → AWS Sagemaker Endpoint → AWS Sagemaker Model

XG Boost Classification

Classfication

TREE #1
- Savings>$1M — Yes → Age > 45? , No → Class #0
- Age > 45? — Yes → Class #1, No → Class #0

TREE #2
- Savings>$1M — Yes → Age > 45? , No → Class #0
- Age > 45? — Yes → Class #1, No → Class #0

TREE #N
- Savings>$1M — Yes → Age > 45? , No → Class #0
- Age > 45? — Yes → Class #1, No → Class #0

OUT = CLASS #1   OUT = CLASS #1   OUT = CLASS #0

MAJORITY VOTE = CLASS #1