# Capstone Project

**1. Smart Support Ticket Categorization using GenAI**

**Description:**

Develop a GenAI-powered system to classify IT support tickets into appropriate categories (e.g., hardware, network, software). The model uses pre-trained transformer models (BERT or DistilBERT) for text classification. Tickets are ingested via API, stored in S3, processed through an ETL Lambda pipeline, trained in SageMaker, and deployed with Docker for inference.

**Tools Involved:**

- **Data Collection & Storage:** JSON/CSV → AWS S3
- **ETL:** AWS Lambda for transformation (cleaning, tokenization)
- **Model:** Fine-tuned Hugging Face Transformers (BERT)
- **Training:** Amazon SageMaker
- **Deployment:** Docker container + FastAPI + ECS
- **Monitoring:** CloudWatch logs
- **CI/CD:** GitHub + GitHub Actions
- **Version Control:** Git
- **Testing:** Pytest

**Dataset:**

- **Dataset Name:** ServiceNow or Zendesk IT Support Dataset (synthetic if not publicly available)
- **Sample Source:** Kaggle Helpdesk Ticket Dataset

**2. GenAI-Powered Resume Screening Assistant**

**Description:**

Design a system that screens resumes and classifies them based on job roles (e.g., Data Scientist, Backend Developer, QA Engineer). The model will parse resume text and compare it with job descriptions using sentence embeddings and transformers (e.g., SBERT).

**Tools Involved:**

- **Data Ingestion:** Upload resumes to S3 (PDF/Text)
- **Preprocessing:** Lambda (text extraction using PDFMiner / Textract)
- **Model:** Sentence-BERT + similarity scoring
- **Training:** SageMaker custom script mode

- **Deployment:** REST API via FastAPI inside Docker
- **Orchestration:** Step Functions (optional) for parsing > embedding > scoring
- **CI/CD:** Jenkins + GitHub
- **Testing:** Pytest

**Dataset:**

- **Dataset Name:** Resume Dataset by Kaggle (CSV + PDFs)
- **Source:** Kaggle Resume Dataset

**3. Smart News Article Topic Classifier**

**Description:**

Create an AI-powered system that classifies news articles into categories like politics, sports, business, etc. This GenAI project will utilize BERT-based models to perform multi-class classification. It includes a full ETL pipeline and deployment to API endpoints.

**Tools Involved:**

- **Data Collection:** News API or use public dataset
- **ETL:** Lambda to clean and tokenize articles
- **Model:** BERT fine-tuned on article titles and content
- **Training:** SageMaker + SM Experiments
- **Deployment:** Docker + FastAPI endpoint
- **CI/CD:** GitHub Actions
- **Monitoring:** S3 logs, CloudWatch
- **Testing:** Pytest

**Dataset:**

- **Dataset Name:** AG News Corpus
- **Source:** AG News Dataset

**4. Legal Document Question-Answering System**

**Description:**

A Q&A system for legal documents using Retrieval-Augmented Generation (RAG). The system ingests legal contracts or case documents, indexes them using FAISS and performs Q&A via OpenAI GPT or LLaMA.

**Tools Involved:**

- **Data Source:** Legal PDFs → S3
- **Preprocessing:** Lambda (PDF to Text), sentence chunking
- **Embedding:** SentenceTransformers + FAISS
- **QA Model:** RAG with GPT or LLaMA in SageMaker
- **Deployment:** Docker + FastAPI
- **UI (Optional):** Streamlit or React
- **CI/CD:** GitHub Actions + CodePipeline
- **Monitoring:** API latency, Lambda logs

**Dataset:**

- **Dataset Name:** CUAD – Contract Understanding Atticus Dataset
- **Source:** CUAD Dataset

**5. GenAI-Driven Product Review Sentiment Analyzer**

**Description:**

Build a system that classifies product reviews as Positive, Negative, or Neutral using BERT and transformers. The end-to-end pipeline covers data ingestion, model training, and deployment with sentiment scoring APIs.

**Tools Involved:**

- **Data Ingestion:** Product reviews → AWS S3
- **ETL:** Lambda (cleaning, sentiment labeling)
- **Model:** DistilBERT fine-tuned for sentiment analysis
- **Training:** SageMaker with training script
- **Deployment:** Docker + API Gateway + Lambda
- **CI/CD:** GitHub Actions
- **Monitoring:** CloudWatch, API Gateway metrics

**Dataset:**

- **Dataset Name:** Amazon Product Reviews
- **Source:** Amazon Reviews Dataset

**6. Intelligent Meeting Transcript Summarizer**

**Description:**

Develop an intelligent system that takes meeting transcripts (Zoom, Teams) and summarizes them using transformer-based summarization models (e.g., BART, T5). The model is deployed to serve real-time summarization over API.

**Tools Involved:**

- **Data Ingestion:** Meeting transcripts (Text or VTT) → S3
- **ETL:** Lambda (cleaning, chunking, formatting)
- **Model:** Pre-trained BART or T5 summarizer (Hugging Face)
- **Training:** SageMaker (optional fine-tuning)
- **Deployment:** FastAPI Docker App
- **API Access:** Expose via API Gateway or ALB
- **CI/CD:** Jenkins or GitHub Actions
- **Monitoring:** Summary quality metrics, request logs

**Dataset:**

- **Dataset Name:** AMI Meeting Corpus or synthetic Zoom transcripts
- **Source:** AMI Corpus