KVALUENT EDTECH

Proposal for Training in App Development with Gen AI and Data Engineering

Client: Prodapt Solutions	Proposal No: P2/2024-25/PD/19
Service Type: Technical Hands-On	Date: 21.012025

About

This training program offers a comprehensive hands-on introduction to Generative AI, empowering freshers to build apps with Gen AI solutions. Participants will explore foundational concepts such as model building, neural networks, and regression techniques, progressing to advanced topics like prompt engineering, embedding flows, and AI-enabled application development.

Through hands-on methodology, learners will create chatbots, implement memory-driven workflows, and design multi-agent systems using various tools. The program also incorporates Data Engineering essentials—scalable pipelines, ETL processes, database schemas, Tableau, and ML model deployment—providing a holistic foundation for AI-driven data solutions.

Curriculum Overview

Telecom	5 days
Shell Scripting	3 days
SDLC, STLC, Agile, DevOps Overview	1 day
Cloud Computing	3 days
Python Core (Refresher)	3 days
Python for Gen AI and Data Analysis	5 days
App Dev with Gen AI	15 days
Data Engineering and Machine Learning	12 days
	47 Days

Kvaluent Edtech
24A Lansdowne Place, Kolkata: 700029
Phone: +91 9903967329
Email: team@kvaluent.com
Web: www.kvaluent.com

Calendar

27 Jan 2025	28 Jan 2025	29 Jan 2025	30 Jan 2025	31 Jan 2025	01 Feb 2025	02 Feb 2025
Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Telecom	Telecom	Telecom	Telecom	Telecom		

03 Feb 2025	04 Feb 2025	05 Feb 2025	06 Feb 2025	07 Feb 2025	08 Feb 2025	09 Feb 2025
Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
		Shell Scripting	Shell Scripting	Shell Scripting		

10 Feb 2025	11 Feb 2025	12 Feb 2025	13 Feb 2025	14 Feb 2025	15 Feb 2025	16 Feb 2025
Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Cloud	Cloud	Cloud	SDLC-Agile	Python Revision		

17 Feb 2025	18 Feb 2025	19 Feb 2025	20 Feb 2025	21 Feb 2025	22 Feb 2025	23 Feb 2025
Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Python Revision	Python Revision	Python (DA)	Python (DA)	Python (DA)		

24 Feb 2025	25 Feb 2025	26 Feb 2025	27 Feb 2025	28 Feb 2025	01 Mar 2025	02 Mar 2025
Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Python (DA)	Python (DA)	Gen AI	Gen AI	Gen AI		

03 Mar 2025	04 Mar 2025	05 Mar 2025	06 Mar 2025	07 Mar 2025	08 Mar 2025	09 Mar 2025
Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Gen AI	Gen AI	Gen AI	Gen AI	Gen AI		

10 Mar 2025	11 Mar 2025	12 Mar 2025	13 Mar 2025	14 Mar 2025	15 Mar 2025	16 Mar 2025
Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Gen AI	Gen AI	Gen AI	Gen AI	Gen AI		

17 Mar 2025	18 Mar 2025	19 Mar 2025	20 Mar 2025	21 Mar 2025	22 Mar 2025	23 Mar 2025
Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Gen AI	Gen AI	Data Engg	Data Engg	Data Engg		

24 March 2025	25 March 2025	26 March 2025	27 March 2025	28 March 2025	29 March 2025	30 March 2025
Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Data Engg	Data Engg	Data Engg	Data Engg	Data Engg		

31 March 2025	01 April 2025	02 April 2025	03 April 2025	04 April 2025	05 April 2025	06 April 2025
Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
ML	ML	ML	ML	Wrap Up		

TOC: Gen AI App Development

Understanding of a model and neural networks

- What is a model?
- Understanding about training a model and doing prediction using Linear regression problem
- Understanding about logistic regression and activation functions
- Understand about What is a Neuron and how it works.
- Understanding gradient descent basics and back propagation
- Understanding about an artificial neural network
- AI vs ML vs Deep Learning vs GEN AI

Labs:

- Creating a model using Linear Regression
- Creating a model with Logistic Regression

Understanding about Generative AI and ChatGPT

- What is Generative AI?
- What are LLMs
- Introduction to OpenAI and ChatGPT
- Using ChatGPT for various tasks using manual prompting
- What kind of applications can be developed using these LLMs

Labs:

- Creating a chatbot using OpenAI API and Streamlit/Gradio

Prompt Engineering

- Prompting guidelines
- Prompt components
- Dividing into sub tasks
- Iterative refinement framework
- Zero, one and few shot prompting
- Text classification
- Chain of thought prompting
- Auto COT
- Least to Most prompting

- ReAct Prompting
- Self-consistency
- Generate Knowledge prompting
- Tree of thought
- Reflection Prompting

Labs:

- Using Open AI playground to try all the prompt engineering techniques

Using Lang chain

- How a typical AI enabled app works?
- Why Langchain?
- Using Langchain for first time
- Understanding and creating a first chain
- Understanding Runnables in detail
- Chaining 2 chains
- Chat vs Completion style models
- Implementing a chat chain
- Understanding the use of memory and different types of memory
- Using Lang smith for tracing
- Using Chainlit and Gradio to develop a chatbot

Labs:

- Creating a chatbot using langchain and streamlit
- Using runnables
- Financial Advisor App using streamlit demonstrating chain of chains
- Modifying the existing app to add memory

Understanding embeddings, Vector store and RAG

- Understanding the embedding flow
- Using Chroma DB as vector store
- Building a retriever chain
- Visualizing embeddings
- Using Pinecone as Vector store

Labs:

- Updating Financial app use RAG using chroma

Understanding tools and agents

- Understanding how to configure tools
- Understanding ChatGPT functions
- Defining a tool to interact with Database
- Defining a tool to search the web
- Understanding how agents work
- Using an agent and agent executor
- Memory and agent scratch pad
- Recovering from errors in tools
- Understanding and implementing callbacks

Labs:

- Creating a custom database tool
- Updating Financial Advisor app to use database tool

Langserve

- Understanding about Langserve
- Deploying using Langserve

Labs:

- Exposing runnables as API

Langgraph

- Why Langgraph?
- Understanding Langgraph components
- Building your first agent with lang graph
- Understanding State of Graph and creating StateGraph
- Tool Node
- Conditional Branching in graph
- Adding Memory to graph
- Building React Agent executor with Langgraph
- Parallel processing in langgraph
- Multi Agentic Application using LangGraph
- SubGraphs
- Building SQL Agent

Labs:

- Creating a state graph with memory
- Creating a custom React Agent using Lang Graph
- Understanding parallel processing
- Creating a multi agentic application using Langgraph
- Creating a customer support bot using subgraphs

Llama index

- Understanding Llama Index
- Building blocks of Llama index
- Creating Indexes with Llama index
- Querying with Llama index
- Integrating with Lang Fuse
- Using Llama Index with Pinecone
- Customizing the retriever, synthesizer and post processors
- Using Connectors in LlamaHub
- Hypothetical Document Embeddings
- Llama Parse
- Develop an app for Converting text to SQL using NLSQLTableQueryEngine, SQLTableRetrieverQueryEngine
- Using RouterQueryEngine
- Combining Text to SQL with semantic Search
- Understanding Property Graphs in detail
- Creating Property graphs with neo4j as graph store and chroma as vector store

Labs:

- Creating a chat bot which Converts Natural Language to SQL Query
- Using Router QueryEngine
- Combining Text to SQL with Semantic Search
- Creating and querying a knowledgeGraph

Crew AI

- Understanding about AI Agents
- Creating your first agent using crew AI
- Creating multi agentic application using Crew AI
- Understanding agentic tools , memory and cooperation

- Multi agent collaboration
- Building a crew finally
- Achieving fault tolerance
- Asynch tasks
- Hierarchical collaboration

Labs:

- Creating a single agent with single task
- Creating a single agent with single task and tools
- Creating a single agent with multiple tasks
- Creating custom tool
- Creating a crew with Asynch tasks
- Using hierarchical process delegation
- Creating a flow with crews

AutoGen

- Why Autogen?
- Conversible Agent
- Understanding Roles and Conversations
- Terminating Conversations
- Human in the loop
- Code executors
- Understanding tools
- Creating ReAct agent using AutoGen
- Implementing sequential chat, group chat and nested chats

Labs:

- Using Conversable Agent
- Using human input and code executors
- Implementing Sequential chat and group chat
- Implementing nested chat
- RAG using Autogen

Open AI Assistant

- Understanding Capabilities of OpenAI Assistant
- Using tools and Function Calling
- RAG using single and multiple files.

Labs:

- Creating a chatbot similar to ChatGPT

TOC: Python

- Basic Syntax, Variables, Data types
- Control flow, Writing Scripts, Processing CSV Data.
- Core data structures (lists, sets, dictionaries)
- String manipulation
- Cleaning and Validating JSON data.
- OOP basics (Classes, Objects)
- Encapsulation, Inheritance, Polymorphism, Abstraction
- File operations (CSV, JSON)
- Writing ETL pipeline classes for reusable transformations
- Error handling with try-except, Logging errors, large file processing
- Introduction to Data Analysis
- Numpy, Creating and manipulating arrays, Broadcasting and vectorized operations
- Introduction to Pandas, Data Manipulation with Pandas
- Loading datasets with pandas, Data inspection, Cleaning, and Analysis
- Multithreading in Pandas operations
- REST API basics, HTTP methods, Consuming APIs using requests library, Fetching data from Public APIs.
- FastAPI for Building Data APIs
- Parsing JSON and automating data extraction
- SQL database connection, executing queries with sqlite3/psycopg2, and ORM with SQLAlchemy

Sample Hands-On Works

- Python script to read, clean, and print a CSV file (e.g., telecom usage data)
- Write a program to clean and validate user data stored in JSON format, Parse telecom logs to extract call durations.
- Analyze call data to identify the top 10 most frequent callers.
- Python script to fetch and process data from a public AP

TOC: Data Engineering and ML

Fundamentals of Data Engineering

- Data sources and data collection methods.
- Data quality and data cleaning techniques.
- Data Lakes and Warehousing – Building scalable data storage systems.

Concept of data pipelines and their components

- Real-time vs batch processing
- Data sources, data ingestion, data transformation, and data storage
- How ETL fits into the context of data pipelines
- Tools Overview: Apache Kafka, Apache NiFi, Apache Airflow, AWS Glue

Dive deeper into the ETL-specific stages: Extraction, Transformation, and Loading

- Data Egress and Agress
- ETL Pipelines, ETL Automation
- ETL tools: Talend, AWS Glue
- Data Transformation Techniques (Revisiting), Data Cleaning

Database Schemas

- Relational and NoSQL databases
- Understanding normalization and denormalization
- Designing efficient schemas for analytics
- Design a schema for an application
- Database Optimization (Indexing, partitioning, and query optimization)

Build and schedule a basic ETL pipeline that reads data from a file and writes to a database

Build an ETL pipeline that extracts data from CSV, transforms it (e.g., cleaning data), and loads it into a MySQL database

Use Python and Pandas to clean and transform a messy dataset, Perform data transformations (e.g., filtering, joining) using PySpark for larger datasets.

Design a normalized database schema in MySQL or PostgreSQL for an application (products, orders, customers).

Set up a NoSQL database (MongoDB) and design a flexible schema for a product catalog

Tableau

- Getting Started with Tableau
- Creating Basic Visualizations: Building bar charts, line charts, and scatter plots
- Using filters and sorting data
- Adding interactivity with dashboards
- Calculated fields and table calculations
- Using parameters to enhance visualizations
- Designing effective dashboards
- Best practices for data storytelling and user interaction
- Publishing and sharing dashboards

Project Using Above Topics – ETL, Database, Data Analysis, Tableau

Machine Learning

- Machine Learning Basics
- Common ML algorithms: Linear Regression, Decision Trees, K-Means Clustering
- Workflow: Problem definition, data preparation, model building
- Django for ML
- Building a Linear Regression Model: Use Scikit-learn to train a simple linear regression model for predicting
- Feature engineering and selection
- Use Python and Scikit-learn to engineer new features from raw data
- Model Training and Evaluation
- Train-test split and cross-validation
- Use Scikit-learn to train and evaluate a Decision Tree model on a classification dataset
- Building and Training Models
- Deploying Machine Learning Models
- Create a Flask REST API to expose your trained ML model as a web service
- Containerization
- Create a Docker container for the Flask API built earlier
- Cloud Deployment with AWS
- Use AWS Lambda to deploy an ML model and invoke it via a REST API.

Overview of MLOps

MLOps Framework

Stages in MLOps, Compare MLflow, Kubeflow, and TensorFlow

TOC: Shell Scripting (Common Curriculum)

- Unix/Linux introduction
- Linux Commands
- What is Shell?
- Different Shell
- Why Shell?
- Shell Programming
    - Syntax and Code Examples
    - Introduction to echo
    - Using Variables (Environmental, Local, Special, Shell)
    - Using Arrays
- Operators
    - Arithmetic Operators
    - Boolean Operators
    - Relational Operators
- Control
    - If-then-Else
    - For loop through a range of variables
    - For loop with iterations
    - While condition
    - Case Statement
- Expression (String Comparisons, Number Comparisons, Files operators)
- Loops (Nested Loops, Infinite Loops, Loop Control)
- Shell Substitution
- Functions and Subroutines
- Input-Output Redirection and File handling Debugging Techniques

TOC: SDLC, STLC, Agile and DevOps Overview

Software Development Life Cycle

Definition and purpose

Phases of SDLC

SDLC Models

Software Testing Life Cycle

Definition and purpose

How it fits into SDLC

Phases of STLC

Test Levels and Types (Unit testing, integration testing, system testing, UAT)

Functional vs. non-functional testing

Agile Methodology

Definition and principles

Agile manifesto and its significance

Agile vs Traditional Models

Agile Frameworks (Scrum, Kanban...)

Agile Artifacts (User stories, Epics and tasks)

Scrum Framework

Introduction to Scrum, How it fits into Agile

Key Roles in Scrum (Product Owner, Scrum Master, Development Team)

Scrum Artifacts (Product backlog, Sprint backlog, Increment)

Scrum Ceremonies

Group activity: Simulate a Scrum sprint for a mini project

Overview of DevOps

Definition and purpose of DevOps

Core Concepts of DevOps (Culture, Automation, Measurement, Sharing: Knowledge)

DevOps Lifecycle

Plan

Develop: Source control and coding practices

Build and Test: Continuous Integration (CI) and testing tools

Release and Deploy: Continuous Deployment (CD) and containerization

Monitoring and performance

TOC: Cloud Fundamentals (Azure)

Overview of cloud computing

Cloud service models: IaaS, PaaS, SaaS

Cloud deployment models: Public, Private, Hybrid

Introduction to Azure portal

Overview of Azure Services

Understanding Azure regions and availability zones

Azure Resource Manager (ARM)

Azure Storage (Blob, File Storage, Tables, Queues)

Azure database services

Azure Identity and Access Management (IAM

Overview of Azure DevOps (Repos, Pipelines, Boards)

Continuous Integration and Deployment (CI/CD) in Azure

TOC: Telecom

Telecom Foundation

Topic 1: Introduction to Service Networks
- Overview of service networks and their importance in modern communication systems
- Different types of service networks (voice, data, video, etc.)
- Service network architectures and their components
- Challenges and considerations in designing service networks

Topic 2: PSTN (Public Switched Telephone Network)
- Evolution and architecture of PSTN
- PSTN components: local loops, central offices, toll offices, and signalling systems
- Call setup and routing in PSTN, Challenges and modernization of PSTN

Topic 3: Mobile Network Architecture
- Overview of mobile networks (2G, 3G, 4G, and 5G)
- Mobile network components: Base stations, mobile switching centres, and mobile core networks
- Call setup and handoff in mobile networks

Topic 4: Internet Protocol (IP) Networks
- Basics of IP networks and their role in service delivery
- IP network components: routers, switches, and IP addressing
- IP routing protocols: RIP, OSPF, BGP
- Quality of Service (QoS) in IP networks

Topic 5: Data Center Networks
- Introduction to data center networks
- Data center network architecture: core, aggregation, and access layers
- Virtualization and network overlays in data center networks
- Scalability and high availability in data center networks

Topic 6: Content Delivery Networks (CDNs)
- Overview of CDNs and their importance in content delivery
- CDN architecture and components: edge servers, content servers, and caching mechanisms
- CDN load balancing and content routing techniques, Performance optimization in CDNs

Topic 7: Cloud Computing Networks
- Introduction to cloud computing and its network requirements
- Cloud computing network models: public, private, and hybrid clouds
- Cloud networking technologies: virtual private clouds (VPC), Software-Defined Networking (SDN)
- Network security considerations in cloud computing

Topic 8: Access Networks
- Introduction to access networks and their role in connecting end-users to service providers
- Types of access networks: DSL, cable, fiber-optic, wireless (Wi-Fi, cellular)
- Access network components: customer premises equipment (CPE), access nodes, distribution points
- Access network protocols and technologies: xDSL, DOCSIS, GPON, LTE, 5G

Topic 9: Edge Networks
- Overview of edge networks and their significance in optimizing service delivery
- Edge network architecture and components: edge routers, content delivery servers, caching systems
- Content distribution and edge computing in edge networks
- Edge network protocols and technologies: BGP anycast, Anycast CDN, service chaining

Topic 10: Core Networks
- Introduction to core networks and their role in routing and transporting data across the network
- Core network components: routers, switches, multiprotocol label switching (MPLS) infrastructure
- Core network protocols and technologies: IP routing (OSPF, BGP), MPLS, segment routing
- Traffic engineering and quality of service (QoS) in core networks

Topic 11: Cable Networks
- Overview of cable networks and their architecture for delivering broadband services
- Hybrid Fiber-Coaxial (HFC) cable network architecture
- DOCSIS (Data Over Cable Service Interface Specification) standards for cable networks
- Cable network components: headend, distribution network, cable modems

OSS-BSS

Topic 1: Introduction to BSS and OSS
- Overview of BSS and OSS systems
- Importance of BSS and OSS in modern telecommunication networks
- Differences between BSS and OSS systems, Common functionalities of BSS and OSS systems

Topic 2: BSS Fundamentals
- Overview of Billing and Rating systems

- Billing life cycle: Customer acquisition, order management, rating, invoicing, payment, and collection
- Rating basics: Time-based rating, volume-based rating, rating discounts, and rating plans
- Understanding simple billing system

Topic 3: OSS Fundamentals

- Overview of Inventory, Activation, and Assurance systems
- Network and Service Inventory, Resource Management, and Configuration Management
- Service Activation: Service Order Management, Service Provisioning, and Service Activation Testing
- Service Assurance: Fault Management, Performance Management, and Security Management
- Understanding inventory system

Topic 4: BSS and OSS Integration

- Overview of the TM Forum Frameworx
- Overview of the eTOM model
- Integration points between BSS and OSS systems
- Understanding end-to-end billing and service management system

Topic 5: Modern BSS and OSS Systems

- Overview of modern BSS and OSS systems
- Cloud-based BSS and OSS systems
- Microservices-based BSS and OSS systems
- AI/ML-based BSS and OSS systems

Methodology:

Location	: Chennai
Delivery Mode	: Classroom Instructor Led
Batch Size	: up to 30