DevelopersHub Corporation

AI/ML Engineering – Advanced Internship Tasks

Due Date: 17th January, 2026

Overview

As part of your **AI/ML Engineering Internship** at **DevelopersHub Corporation**, you are required to complete **at least 3 out of the 5 advanced tasks** listed below within the given timeline.

These tasks are carefully designed to give you hands-on experience with **cutting-edge machine learning and artificial intelligence techniques**, including **transformer models, ML pipelines, multimodal learning, conversational AI, and LLM applications**.

You are encouraged to complete all tasks to strengthen your technical portfolio. Technologies and tools you'll use include:
`Hugging Face Transformers`, `scikit-learn`, `LangChain`, `Streamlit`, `Gradio`, CNNs, `joblib`, and `LLMs`.

Advanced Task Set

Task 1: News Topic Classifier Using BERT

Objective:
Fine-tune a transformer model (e.g., BERT) to classify news headlines into topic categories.

Dataset:
AG News Dataset (Available on Hugging Face Datasets)

Instructions:

- Tokenize and preprocess the dataset

- Fine-tune the `bert-base-uncased` model using Hugging Face Transformers

- Evaluate the model using accuracy and F1-score
- Deploy the model using **Streamlit** or **Gradio** for live interaction

Skills Gained:

- NLP using Transformers

- Transfer learning & fine-tuning

- Evaluation metrics for text classification

- Lightweight model deployment

Task 2: End-to-End ML Pipeline with Scikit-learn Pipeline API

Objective:
Build a reusable and production-ready machine learning pipeline for predicting customer churn.

Dataset:
Telco Churn Dataset

Instructions:

- Implement data preprocessing steps (e.g., scaling, encoding) using `Pipeline`

- Train models like Logistic Regression and Random Forest

- Use `GridSearchCV` for hyperparameter tuning

- Export the complete pipeline using `joblib`

Skills Gained:

- ML pipeline construction

- Hyperparameter tuning with GridSearch

- Model export and reusability
- Production-readiness practices

Task 3: Multimodal ML – Housing Price Prediction Using Images + Tabular Data

Objective:
 Predict housing prices using both structured data and house images.

Dataset:
 Housing Sales Dataset + Custom Image Dataset (your own or any public source)

Instructions:

- Use CNNs to extract features from images

- Combine extracted image features with tabular data

- Train a model using both modalities

- Evaluate performance using MAE and RMSE

Skills Gained:

- Multimodal machine learning

- Convolutional Neural Networks (CNNs)

- Feature fusion (image + tabular)

- Regression modeling and evaluation

Task 4: Context-Aware Chatbot Using LangChain or RAG

Objective:
 Build a conversational chatbot that can remember context and retrieve external information

during conversations.

Dataset:
Custom corpus (e.g., Wikipedia pages, internal documents, or any knowledge base)

Instructions:

● Use LangChain or Retrieval-Augmented Generation (RAG)

● Implement context memory for conversational history

● Retrieve answers from a vectorized document store

● Deploy the chatbot with **Streamlit**

Skills Gained:

● Conversational AI development

● Document embedding and vector search

● Retrieval-Augmented Generation (RAG)

● LLM integration and deployment

Task 5: Auto Tagging Support Tickets Using LLM

Objective:
Automatically tag support tickets into categories using a large language model (LLM).

Dataset:
Free-text Support Ticket Dataset

Instructions:

● Use prompt engineering or fine-tuning with an LLM
● Compare zero-shot vs fine-tuned performance

● Apply few-shot learning techniques to improve accuracy

● Output top 3 most probable tags per ticket

Skills Gained:

● Prompt engineering

● LLM-based text classification

● Zero-shot and few-shot learning

● Multi-class prediction and ranking

Submission Requirements (Per Task)

Each completed task must be **uploaded to your GitHub repository** and submitted via **Google Classroom**.

Checklist:

1. **Jupyter Notebook / Script File**

   ○ Problem Statement & Objective

   ○ Dataset Loading & Preprocessing

   ○ Model Development & Training

   ○ Evaluation with relevant metrics

   ○ Visualizations (if applicable)

   ○ Final Summary / Insights

2. **Code Quality**
   ○ Clear structure, logical flow, comments explaining major steps

3. **GitHub Repository**

   ○ Clear repository name and organization

○ README.md must include:

■ Objective of the task

■ Methodology / Approach

■ Key results or observations

4. **Submission on Google Classroom**

○ Submit the GitHub repo link for each completed task

Important Notes

● **Deadline:** Complete at least **3 out of 5 tasks** by **17th January,**

2026

● You are encouraged to complete all 5 for advanced portfolio building

● Share your work on GitHub and LinkedIn to showcase your skills