**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

| Date | 30 July 2025 |
| --- | --- |
| Team ID | PNT2025TMID09555 |
| Project Name | The Roadmap Generator |
| Maximum Marks | 4 Marks |

## 

### **Overview**

Roadmap Generator is a Streamlit-based web app that generates personalized skill-learning paths using **LLaMA 2** (via CTransformers). It allows users to input a desired skill and returns a customized roadmap with resources.

## **Architecture Overview**

[User Interface - Streamlit Web App]

|

v

[Streamlit Backend Server - app.py]

|

v

[Prompt Builder + Model Handler (Python Utils)]

|

v

[CTransformers (LLaMA 2 Inference)]

|

v

[Model File (.gguf) via setup.sh]

## **Technology Stack**

| **Layer** | **Tool/Library** | **Purpose/Role** |
| --- | --- | --- |
| **Frontend** | Streamlit | UI for user inputs and displaying roadmap |
| **Backend (API Logic)** | Python + Streamlit Scripts | Receives user input, calls LLaMA 2, sends response |
| **Model Integration** | [CTransformers](https://github.com/marella/ctransformers) | Interface for running .gguf LLaMA 2 models |
| **Model File** | LLaMA 2 (GGUF Format) | Pre-trained model for text generation |
| **Environment Mgmt** | requirements.txt + venv | Package & environment management |
| **Model Setup** | setup.sh | Downloads the LLaMA 2 model file from Hugging Face |
| **Deployment** | Streamlit Cloud / Localhost | App hosting & access |
| **Version Control** | Git + GitHub | Code repository and version tracking |

## 

## 

## 

## 

## **Folder Structure**

roadmapper/

├── app.py

├── setup.sh

├── requirements.txt

├── README.md

├── .gitignore

├── model/ ← (stores downloaded .gguf model)

│ └── llama-2-7b-chat.Q3\_K\_M.gguf (auto-downloaded)

└── utils/

└── model\_loader.py ← Loads model using CTransformers

## **Workflow**

1. User opens the app and enters a skill (e.g., "Natural Language Processing").
2. app.py receives the input and formats a prompt.
3. model\_loader.py uses **CTransformers** to load the .gguf model and generate a response.
4. The roadmap is returned and displayed using Streamlit.
5. Model is downloaded (once) using setup.sh if not already present.

## **Dev Notes**

* **Model Size**: Avoid committing .gguf to GitHub; use setup.sh for dynamic download.
* **Modularity**: Logic is separated into utility scripts for reusability and maintainability.
* **Model Format**: LLaMA 2 in GGUF format works with CTransformers, optimized for CPU/GPU.