**Project Design Phase-II**

**Data Flow Diagram & User Stories**

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

## **Data Flow Diagram (DFD)**

### **Level 0: Context Diagram**

+-----------------+ Input Skill +--------------------+

| User | -------------------------> | Roadmap Generator |

| | <------------------------- | (Streamlit App) |

+-----------------+ Roadmap + Resources +--------------------+

### **Level 1: Detailed DFD**

+---------------------+

| User |

+---------+-----------+

|

| Input skill (e.g., NLP, ML)

v

+-------------------------+

| Streamlit Frontend UI |

+-------------------------+

|

| Pass input via Streamlit

v

+-------------------------+

| app.py (Backend) |

| - Receives input |

| - Validates data |

+-------------------------+

|

| Processed input

v

+-----------------------------+

| Model Loader (utils/) |

| - Load LLaMA2 w/ GGUF |

| - CTransformers interface |

+-----------------------------+

|

| Prompt + generate

v

+-----------------------------+

| LLaMA2 Model (gguf) |

| - Generate roadmap |

| - Include free resources |

+-----------------------------+

|

| Output roadmap

v

+-----------------------------+

| Streamlit Renderer (UI) |

| - Display roadmap |

| - Links to resources |

+-----------------------------+

|

| See & follow roadmap

v

+---------------------+

| User |

+---------------------+

## **User Stories**

### **Story 1: Skill Seeker**

*As a learner, I want to input the name of a skill I want to learn (e.g., NLP), so that I receive a personalized roadmap.*

* Input skill from frontend
* Receive roadmap with free resources
* Follow step-by-step plan

### 

### **Story 2: Developer**

*As a developer, I want a backend that loads the LLaMA2 model efficiently and interfaces smoothly with the Streamlit frontend.*

* Load .gguf model only once
* Use CTransformers for memory efficiency
* Avoid reloading model on every input

### **Story 3: Deployer**

*As the app owner, I want the app to deploy seamlessly on Streamlit Cloud without manually uploading large models.*

* Use setup.sh to auto-download model
* Ignore large files via .gitignore
* Keep GitHub repo lightweight

### **Story 4: Casual Visitor**

*As a new user, I want a clean and intuitive UI that explains what the app does and how to use it.*

* Hero section with project description
* Simple text input box
* Easy-to-read roadmap output