**PROJECT DESIGN**

**1.Tech Stack**

Backend AI: CodeLlama (via HuggingFace transformers)

Frontend: Streamlit

Deployment Tool: Localtunnel (to expose the Streamlit app to the internet)

Frameworks & Libraries:

transformers

torch

streamlit

localtunnel

**2.⁠ ⁠Installation Steps**

Install transformers and accelerate from GitHub (latest versions)

Install Streamlit for the UI

Install Localtunnel to make your app publicly accessible

**3.⁠ ⁠Code Structure**

* CodeModel.py – Handles Code Generation Logic

Loads the CodeLlama-7b-hf model and tokenizer.

Defines generate\_code(prompt) function:

Accepts user prompt

Uses HuggingFace pipeline for text generation

Returns generated code along with time metrics

* app.py – Streamlit Frontend

Collects user input through a text\_area

Submits prompt through a button and displays output

Includes:

Spinner during processing

Time taken for model inference

Sidebar with usage guide and examples

4. Deployment – Detailed Explanation

After building your model and UI components, deployment is necessary to make CodeGenie accessible via a browser. Since you're using Streamlit for the UI and working in a notebook/cloud environment (like Colab), you expose your local server to the public using Localtunnel.

* Step 1: Run the Streamlit App Locally

!streamlit run app.py: Starts the Streamlit app.

--server.address=localhost: Binds the app to local machine address (not public by default).

&>/content/logs.txt: Redirects output logs to a file so it doesn't block the notebook.

&: Runs the command in the background, so the notebook remains usable.

* Step 2: Find Your Public IP

import urllib

print("Password/Endpoint IP for localtunnel is:", urllib.request.urlopen('https://ipv4.icanhazip.com').read().decode('utf8').strip("\n"))

This command fetches your current IPv4 public address.

Useful to track your session or for debugging access issues.

* Step 3: Expose the App Publicly via Localtunnel