**IDEATION**

**1. Project Idea**

CodeGenie is an intelligent, AI-powered application designed to automate code generation using CodeLlama, a specialized Large Language Model (LLM) developed by Meta for programming tasks. The tool takes natural language prompts from the user and generates fully functional code snippets in real-time, displayed via a Streamlit-based user interface.

**2. Problem Statement**

Developers and students often spend significant time writing repetitive code or debugging syntax issues. There's a need for a system that can:

* Understand human language input.
* Convert it into correct and efficient code.
* Help in learning and enhancing coding skills.

**3. Solution Overview**

CodeGenie leverages the CodeLlama LLM to transform user prompts into usable code. It:

* Allows users to describe the desired functionality in plain English.
* Automatically generates accurate and readable code.
* Offers educational insights like syntax checking and suggestions.

**4. Learning Prerequisites**

To build CodeGenie, the following foundational knowledge is essential:

* LLMs and Prompt Engineering: Understanding how large language models work and how to craft effective prompts.

**Reference: LLM Prompt Engineering for Beginners**

* CodeLlama: Familiarity with Meta's CodeLlama and how it differs from general-purpose LLMs in handling code-specific tasks.

**Reference: Meta’s Blog on CodeLlama**

* Streamlit: Basics of building web interfaces with Streamlit to create a seamless frontend for user interaction.

**Reference: Streamlit Tutorial**

**5. Project Goals**

Build a functional web application where users can enter text prompts and receive AI-generated code.

* Support developers by automating routine coding tasks.
* Assist students by providing code suggestions, comments, and feedback.

**6. Use Case Scenarios**

**Scenario 1: Automated Code Writing**

A developer inputs “create a Python function to sort a list using bubble sort,” and CodeGenie returns the complete, functional code with comments and library imports.

**Scenario 2: Educational Support**

A student inputs their assignment prompt. CodeGenie:

Generates sample solutions.

Highlights optimization areas.

Provides educational commentary on the code structure.

**Scenario 3: Rapid Prototyping for Startups and Freelancers**

Startups or freelance developers often need to quickly prototype features to pitch ideas or test concepts. With CodeGenie, they can simply describe what they want — for example, "create a Flask API with two endpoints for user registration and login" — and receive a working codebase within seconds.

This allows them to:

Save time during MVP development.

Test and iterate faster with real working code.

Focus on innovation, not repetitive coding.

**7. Why CodeGenie?**

Saves time by eliminating boilerplate code writing.

Enhances code quality and consistency.

Provides accessible learning support to new programmers.

Offers a simple UI to interact with powerful AI backend.

---

Would you like help drafting the next folder (Project Design) too?