# AI-Augmented SDLC Project Assignment

This assignment is designed to evaluate your ability to learn quickly, explore AI frameworks, and apply AI to real-world software development lifecycle (SDLC) phases. You will select one or two phases of the SDLC and demonstrate how AI can be applied deeply in those phases.

## Objective

Showcase your ability to:  
- Learn and apply AI frameworks (e.g., LangGraph, GRUAI, AgentFlows, Hugging Face, LLMs).  
- Use prompting and dynamic prompting to improve AI results.  
- Integrate AI into one or two chosen SDLC phases.  
- Deliver working outputs along with clear documentation of your approach.

## Choose from the Following Phases

1. 1. Requirements Engineering (Business Requirements → User Stories)

- Take a plain English problem description and use AI to generate structured requirements/user stories.  
- Produce acceptance criteria.  
- Optionally classify requirements (functional vs non-functional).  
Stretch Goal: Show dynamic prompting to refine requirements based on feedback.

1. 2. Design Phase (Architecture & Flows)

- Use AI to propose a high-level system design (textual or diagrammatic).  
- Justify the design decisions (AI output + your reasoning).  
Stretch Goal: Use multi-agent flows where one agent proposes and another critiques/refines.

1. 3. Coding & Test Generation

- Use AI to generate skeleton code for a feature (choose any language).  
- Extend/refine it to make it work.  
- Use AI to generate unit/integration test cases, then automate them.  
Stretch Goal: Show how dynamic prompting improves test quality.

1. 4. CI/CD & Deployment Support

- Use AI to create a simple pipeline configuration (GitHub Actions, GitLab, Jenkins).  
- Generate deployment artifacts (Dockerfile, Kubernetes manifest).  
Stretch Goal: Use AI to detect missing best practices and auto-fix them.

## Constraints & Deliverables

- Timeframe: 2 days.  
- Tools: Any AI/LLM framework (your choice, justify why).  
- Deliverables:  
 1. Working prototype or demo showing chosen phase(s).  
 2. Documentation (2–3 pages) covering:  
 • Framework(s) used and why.  
 • How AI was integrated.  
 • Examples of prompts/dynamic prompting.  
 • What worked, what didn’t.  
 3. Code repository with commits showing progress.

## Evaluation Criteria

You will be evaluated on:  
- Depth of exploration of AI in the chosen phase(s).  
- Framework understanding and learning agility.  
- Creativity in applying AI.  
- Clarity of documentation and reasoning.  
- Practical relevance of your approach.

## Example Assignments (Choose One)

1. Requirements + Testing: Take a business description → AI-generated user stories → AI-generated test cases.  
2. Design Only: Use multi-agent AI flow to propose/refine architecture for a mini app.  
3. Coding + Testing: Use AI to generate starter code and tests for an 'Expense Splitter' or 'Homework Tracker' app.