CONCLUSIONS AND FUTURE SCOPE

5.1 Work Accomplished

The CodeCodez system has made substantial progress toward its envisioned goal of automating the complete software generation lifecycle through task decomposition and multi-agent LLM coordination. The following key milestones have been successfully achieved:

1. Natural Language to Full-Fledged Software Projects

A core achievement of the project is the development of a pipeline that accepts natural language prompts and transforms them into structured, complete software projects. Users can specify project types, functionality, or features in plain English, and the system intelligently interprets this input to produce:

- Organized directory structures,
- Modular source code,
- Required configuration files (e.g., requirements.txt, .env, etc.),
- Frontend-backend separation where applicable.

This automation reduces the gap between ideation and initial development, removing the need for boilerplate setup and file scaffolding.

2. Task Decomposition Engine Using Trees and DAGs

The implementation of a hierarchical task decomposition engine has been central to the system's capability. Using tree and graph-based planning algorithms, the system breaks down complex software requirements into granular, logical tasks and subtasks.

Each node in the tree represents a task (e.g., API creation, UI layout, or test generation), and dependencies are tracked via a Directed Acyclic Graph (DAG) structure. This ensures task ordering is preserved, and parallelization opportunities are leveraged effectively.