

Software Engineering And Project Management Lab:- 04

Aim:- To Study Project Scheduling Using Gantt Chart

What is a Gantt Chart?

A Gantt chart is a visual project management tool that displays tasks or activities along a timeline, making it easy to see what needs to be done, when, and by whom. Each task is represented as a horizontal bar, with its position and length reflecting the start date, duration, and end date.

Key Components of a Gantt Chart:

- Tasks/Activities: Listed on the vertical axis.
- Timeline: Shown on the horizontal axis (days, weeks, months, etc.).
- Bars: Each bar represents a task, showing its start and end dates.
- Dependencies: Lines or arrows that connect tasks to show relationships (e.g., Task B can't start until Task A is finished).
- Milestones: Special markers for significant events or deadlines.
- Progress Indicators: Visual cues showing how much of a task is complete.

Advantages:

- Provides a clear visual overview of the project schedule.
- Makes it easy to track progress and deadlines.
- Helps identify task dependencies and potential bottlenecks.
- Useful for resource planning and workload balancing.
- Facilitates communication among team members and stakeholders.

Limitations:

- Can become cluttered and hard to read for large or complex projects.
- Updating the chart can be time-consuming if there are frequent changes.
- May not show the full complexity of task relationships in very detailed projects.
- Focuses on schedule, not on resource allocation or costs unless combined with other tools.

About The project:

The project aims to build an LLM Agent app, covering requirements gathering, architecture, UI/UX design, backend and API integration, user management, testing, deployment, and documentation.

There are 26 tasks in total, organized by phases such as planning, development, integration, testing, and deployment.

Most tasks are already marked as "deployed," with some still "not started," "in development," or "testing."

Key tasks include:

1. Defining requirements and MVP features
2. Designing architecture and user interface
3. Implementing user management and authentication
4. Developing core LLM integration and prompt management
5. Testing, deployment, and collecting user feedback

Software Engineering And Project Management Lab:- 04

Aim:- To Study Project Scheduling Using Gantt Chart

Here are a few sample tasks:

Project Overview								
Name	Assignee	Feature Category	Testing Requirements	User Impact	Implementation Deadline	Feature Description	Status	Due date
Define requirements and scope for LLM Agent app		Backend Development	-	-	-	-	DEPLOYED	Mon
Design architecture and user interface		User Interface	-	-	-	-	DEPLOYED	Wed
Implement user management and authentication		Backend Development	-	-	-	-	DEPLOYED	Tue
Testing, deployment, and documentation		Testing and QA	-	-	-	-	DEPLOYED	3/6/26
PLANNING DONE			-	-	-	-	DEPLOYED	3/5/26
+ Add Task								
Conduct integration testing		Testing and QA	-	-	-	-	TESTING	3/11/26
+ Add Task								
Develop core LLM integration		API Integration	-	-	-	-	IN DEVELOPMENT	3/10/26
Integrate LLM API		API Integration	-	-	-	-	NOT STARTED	3/6/26
Test LLM integration		Testing and QA	-	-	-	-	NOT STARTED	3/9/26
Implement prompt management		Backend Development	-	-	-	-	NOT STARTED	3/10/26
+ Add Task								
Write and execute test cases		Testing and QA	-	-	-	-	NOT STARTED	3/19/26
Perform regression testing		Testing and QA	-	-	-	-	NOT STARTED	3/23/26
Monitor production deployment		Testing and QA	-	-	-	-	NOT STARTED	3/24/26
Collect user feedback post-deployment		Testing and QA	-	-	-	-	NOT STARTED	3/25/26
+ Add Task								

Task Details								
Name	Assignee	Feature Category	Testing Requirements	User Impact	Implementation Deadline	Feature Description	Status	Due date
Write and execute test cases		Testing and QA	-	-	-	-	NOT STARTED	3/19/26
Perform regression testing		Testing and QA	-	-	-	-	NOT STARTED	3/23/26
Monitor production deployment		Testing and QA	-	-	-	-	NOT STARTED	3/24/26
Collect user feedback post-deployment		Testing and QA	-	-	-	-	NOT STARTED	3/25/26
+ Add Task								

Subtask View								
Name	Assignee	Feature Category	Testing Requirements	User Impact	Implementation Deadline	Feature Description	Status	Due date
Write and execute test cases		Testing and QA	-	-	-	-	NOT STARTED	3/19/26
Perform regression testing		Testing and QA	-	-	-	-	NOT STARTED	3/23/26
Monitor production deployment		Testing and QA	-	-	-	-	NOT STARTED	3/24/26
Collect user feedback post-deployment		Testing and QA	-	-	-	-	NOT STARTED	3/25/26
+ Add Task								
Develop core LLM integration		API Integration	-	-	-	-	IN DEVELOPMENT	3/10/26
Conduct integration testing		Testing and QA	-	-	-	-	TESTING	3/11/26
+ Add Task								
Define requirements and scope for LLM Agent app		Backend Development	-	-	-	-	DEPLOYED	Mon
Design architecture and user interface		User Interface	-	-	-	-	DEPLOYED	Wed
Implement user management and authentication		Backend Development	-	-	-	-	DEPLOYED	Tue
Testing, deployment, and documentation		Testing and QA	-	-	-	-	DEPLOYED	3/6/26
PLANNING DONE			-	-	-	-	DEPLOYED	3/5/26
+ Add Task								

Software Engineering And Project Management Lab:- 04

Aim:- To Study Project Scheduling Using Gantt Chart

Team Space / **PromptZerk - Prompt Engineering Platform**

Add Channel List View Board View **Gantt** + View

Today Week Auto fit Export

Name	Due Date
Define requirements and scope for LLM...	Mar 2
Identify user personas and use c...	Feb 26
Define MVP features and deliverab...	Feb 27
Gather technical requirements	Mar 2
Implement user management and auth...	Mar 3
Set up user roles and permissions	Feb 27
Implement authentication logic	Mar 2
Develop user registration flow	Mar 3
Design architecture and user interface	Mar 4
Create system architecture diagr...	Mar 16
Review and finalize design	Mar 15
Design UI wireframes	Mar 10
◆ PLANNING DONE	Mar 5
Testing, deployment, and documentation	Mar 6
Deploy app to production environ...	Mar 4
Prepare user and developer doc...	Mar 5
Write test cases and perform QA	Mar 6
Develop core LLM integration	Mar 10
Integrate LLM API	Mar 6
Test LLM integration	Mar 9
Implement prompt management	Mar 10
Conduct integration testing	Mar 11
Write and execute test cases	Mar 19
Perform regression testing	Mar 23
Monitor production deployment	Mar 24
Collect user feedback post-deployment	Mar 25

+ Add Task

