Waterfall Model for Task Management App

1. Requirement Analysis

- Gather detailed requirements from stakeholders.
- Define essential features:
- Task creation, assignment, and tracking.
- Notifications and reminders.
- User roles (Admin, Team Members, Project Manager).
- Reports and analytics.
- Document functional and non-functional requirements.

2. System Design

- High-Level Design (HLD):
- Select tech stack (React/Flutter for frontend, Node.js/Django for backend).
- Define architecture (client-server model, cloud storage).
- Low-Level Design (LLD):
- UI wireframes.
- Database schema (tables for users, tasks, comments, attachments).
- API design (task creation, updates, notifications).

3. Implementation (Coding)

Set up version control (GitHub/GitLab).
Develop modules in this order:
1. User authentication and management.
2. Task creation, assignment, and updates.
3. Notifications and reminders.
4. Reports and analytics.

Write clean, documented code.

4. Testing

- Unit Testing: Validate individual functions (e.g., task creation, notifications).
- Integration Testing: Ensure smooth interaction between UI, backend, and database.
- System Testing: Run end-to-end tests covering all workflows.
- User Acceptance Testing (UAT): Get real user feedback before deployment.

5. Deployment & Maintenance

- Deploy the app on cloud (AWS/GCP/Azure).
- Monitor system performance and logs.
- Fix bugs and roll out security updates.
- Plan feature enhancements based on user feedback