Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur

Project Kit

Title of the Project

Design and Implementation of a Work Planner

Abstract of the Project

The Work Planner is an advanced task and project management system built to facilitate collaboration between managers and team members. The primary goal is to provide an efficient workflow solution where the manager has full control over task assignments, approvals, and tracking. Team members have the flexibility to add tasks, request new assignments, and update their progress regularly. The system supports both high-level project management and granular task tracking, enabling seamless collaboration in a structured environment.

The platform ensures transparency, accountability, and smooth communication within teams, making it easier for organizations to meet project deadlines and deliver quality results.

Generic keyword:

Task Management, Project Management, Team Collaboration, Workflow Automation, Task Assignment, Progress Tracking, User Authentication, Project Reporting.

Specific Technology keywords:

HTML, CSS, JavaScript, MySQL, Python,Java,Web-Based Application, Database Management.

Functional Components of the Project :

Manager:

- **Add Team Members:** The manager can onboard team members into the system.
- **Create Projects:** The manager can create and define project details, such as name, deadlines, and priority.
- **Assign Tasks:** Tasks can be assigned to specific team members based on project requirements.
- **Approve Suggested Tasks:** Team members can suggest new tasks, and the manager has the authority to approve or reject these.
- **Monitor Progress:** The manager can view tasks project-wise and team member-wise to track the completion of tasks and assess individual performance

Team Member:

- **View Assigned Tasks**: Team members can see the tasks assigned to them by the manager.
- **Suggest New Tasks**: Team members have the ability to suggest tasks, which are subject to approval by the manager.
- **Update Task Progress**: Team members can mark the completion percentage of each task and provide updates.

Functionality:

Manager :

- Add team members.
- Create projects with details.
- Assign tasks to team members.

- Approve or reject tasks suggested by team members.
- Monitor tasks by project or team member.

Team Member:

- View assigned tasks.
- Suggest new tasks.
- Mark progress on assigned tasks.
- Only approved tasks will be visible and tracked

Customers:

1. **Managers :**
   Manage team members, create projects, assign tasks, approve task suggestions, and monitor task progress.
2. **Team Members :**
   View assigned tasks, suggest new tasks and update task progress once approved by the manager.

Administrators :

Administrators oversee the system's overall functionality, including user account management (both managers and team members), system maintenance, monitoring task flows, ensuring smooth operation, handling security protocols, and resolving technical issues within the platform.

Steps to start-off the project:

The following steps will be helpful to start off the project –

1. Get a firm grasp on the above technology.
2. Get the domain knowledge. (Speak to a corporate manager or an employee to get better understanding of the system.)

3. Decide on the number of users and their profile.
4. Help should be very user friendly.
5. UI should include good images and have a constant look and feel throughoutthe application.

In the **Design and Implementation of a Work Planner** project, the **Administrator** role focuses on managing and maintaining the system's overall functionality, ensuring the platform operates smoothly for both managers and team members. The key responsibilities of administrators include:

1. **User Management:**
    - Administrators oversee the creation, modification, and deletion of user accounts, including both managers and team members.
    - They ensure appropriate roles and permissions are assigned to users (managers can assign tasks, team members can update progress, etc.).
2. **Project and Task Monitoring:**
    - Administrators have access to monitor all projects and tasks at a high level.
    - They ensure tasks are progressing smoothly across all teams, assisting with any task-related conflicts or issues.
3. **System Maintenance:**
    - They are responsible for ensuring the system is up-to-date and fully functional, including routine updates and bug fixes.
    - Administrators monitor server health, database performance, and overall system load to prevent any downtime.
4. **Security Management:**
    - Ensures user authentication is secure and data privacy is maintained. o Administrators manage access control and enforce security policies to protect sensitive project and task data.

5. **Backup and Recovery:**
    - Implement data backup systems to ensure that project and task data can be recovered in case of system failure.
    - They also handle disaster recovery protocols and data integrity management.
6. **Reporting and Analytics:**
    - Administrators can generate system-wide reports, including user activity, task completion rates, and overall project progress.
    - They provide insights to improve the efficiency and productivity of managers and team members.
7. **Support and Issue Resolution:**
    - Administrators provide technical support to managers and team members, resolving issues related to task assignment, progress tracking, and system usage.
    - They manage a helpdesk or support ticketing system to handle user queries and complaints.

Requirements

Hardware Requirements :

Number	Description	Alternative(if Available)
1.	Minimum requirements- Processor, x86-64 bit CPU	
2.	Ram -4Gb, Disk Space -5Gb.	

Software Requirements :

Numbers	Descriptions	Alternatives(if Available)
1.	Client on Intranet - User Interface, Windows OS	
2.	Development end- Eclipse IDE, Spring Boot, VS code, MySQL, Windows OS.	

Manpower requirements :

2 to 3 students can complete this in 4 – 6 months if they work fulltime on it.

Milestones and Timelines

No	Milestone Name	Milestone Description	Timeline (week no.)	Remarks
1	Requirement Specification	Complete specification of the system with necessary assumptions	Week 2-3	Attempt to add relevant functionalities beyond the listed ones..
2	Technology familiarization	Understanding the required technology (React.js, Node.js, MySQL) and its application in the project.	Week 4	Focus on practical application rather than theory..
3	Database creation	Create a database to manage tasks, team members, and projects.	Week 5	Finalize the database structure to process with testing

4	High-Level Design	Design detailed scenarios, flowcharts, or pseudocode to handle task assignment, approval, and progress tracking.	Week 6	Each scenario should map to the system's requirements.
5	Implementation	Develop the front-end for task assignment, approval, and progress updates, along with login options	Week 7-8	Start working on a test plan for the system.
6	Integration of Front-end & DB	Connect the front-end (React.js) with the backend (Node.js, MySQL), ensuring task management features work smoothly.	Week 12-13	Prepare the system for integration testing
7	Integration	Thoroughly test the system by running all test cases, including task assignment and progress approval flows.	Week 14-15	Allocate 2 extra weeks to handle any issues

Guidelines

- Ensure the user interface is intuitive and user-friendly for both managers and team members.
- Maintain clear role-based access control (managers vs. team members)
- Implement secure user authentication and data protection
- Regualrly test all features to ensure smooth task assignment , apporval and progress tracking.

References

- Object-Oriented Modeling and Design with UML - Michael Blaha, James Rumbaugh.
- React.js Official Documentation - https://reactjs.org/docs/getting-started.html
- Node.js and Express.js Documentation - https://nodejs.org/en/docs/
- MySQL Database Management - https://dev.mysql.com/doc/
- WebSocket API - https://developer.mozilla.org/en
- US/docs/Web/API/WebSockets