

# PLANWISE

- Simplifying Productivity for Students and Teams

Capstone Project: Week 1 – Ideation and

Documentation Stage

Importance of Business Research and Team Dynamics

Business Research, Ideation, Product Vision

## **INTRODUCTION:**

Not every missed deadline is a result of laziness. Sometimes, it's just the chaos — juggling WhatsApp messages, scattered sticky notes, outdated spreadsheets, and teammates working without alignment. This is the everyday reality for students, small project teams, and freelancers. The real issue isn't a lack of motivation, but the absence of a system that matches the fast-paced, collaborative, and often messy nature of how we actually work today.

**PlanWise** was born from that struggle. It's not another bloated, corporate-style platform. Instead, PlanWise offers a clean, intuitive space where task management feels natural — enabling users to create, prioritize, assign, and track tasks without getting overwhelmed. It's more than just a to-do list. PlanWise acts as a digital teammate — simple, visual, collaborative, and built for real people managing real work in real time.

## **BUSINESS RESEARCH:**

### **Chosen Domain: EdTech**

For PlanWise, we conducted market research to analyze trends in productivity tools, study user behavior, and evaluate gaps in current task management systems.

## **Market Problem:**

Many students, freelancers, and small teams rely on fragmented methods to manage their daily work — such as sticky notes, messaging apps, spreadsheets, and basic to-do apps. These tools either lack features like collaboration and notifications or are too complex and bloated for everyday use. This results in confusion, missed deadlines, poor time management, and decreased productivity.

## **TARGET USERS :**

- Students
- Freelancers
- Small Project Teams
- Academic Project Groups
- Remote Teams (Future Scope)

## **COMPETITORS :**

- Trello
- Asana
- Todoist
- ClickUp
- Notion
- Microsoft To Do
- Google Tasks + Calendar

## **CUSTOMER FEEDBACK FOR EXISTING TOOLS :**

### **1. Notion**

"I love how flexible Notion is, but setting it up for task management took me hours. I just wanted something that worked out of the box."

— College student on Reddit

"Notion is powerful, but too overwhelming for simple team collaboration. Our group members ended up using WhatsApp instead."

— Engineering student review on G2

## 2. Asana

"Great for project management, but honestly, it's overkill for small teams. We spent more time updating Asana than doing the work."

— Freelancer team leader on Capterra

"I found Asana useful but too formal and business-like for academic projects."

— User review on TrustRadius

## 3. Google Tasks

"Very minimal. I needed collaboration features like assigning tasks or even adding comments — none of that exists."

— App Store reviewer

"Easy to use, but can't track progress or prioritize. Good for solo reminders only."

— Student review on Play Store

### **Unique Value Proposition (USP) :**

- Combines both personal and collaborative task management
- Lightweight but powerful — ideal for students and freelancers
- Simplified UI with real-time features, reminders, and progress bars
- Fills the gap between overly basic to-do apps and complex enterprise tools

## **SWOT Analysis – PlanWise**

A SWOT analysis helps evaluate the internal and external factors that can impact the success of the PlanWise project. It gives strategic insights into where we are strong, what risks we face, and how we can improve.

### **Strengths (S)**

- User-friendly interface with low learning curve
- Real-time collaboration support
- Clean, mobile-responsive dashboard - Designed for both personal and academic use
- Built using MERN stack (modern, scalable technology)

### **Weaknesses (W)**

- Limited brand visibility as a new product
- No advanced integrations (e.g., Google Calendar, Slack) at MVP stage
- Lacks offline functionality in early versions

### **Opportunities (O)**

- Growing demand for remote and hybrid productivity tools
- Rising student and freelancer communities worldwide
- Potential to expand into event planning, academic scheduling, or personal productivity coaching.

### **Threats (T)**

- Strong competition from tools like Trello, Asana, Notion, Todoist
- User loyalty to existing platforms
- Possible feature fatigue if too many additions dilute simplicity

## **PROJECT IDEATION:**

This project proposes the development of a **modern, collaborative task management platform** tailored for students, freelancers, and small teams who need a simple yet powerful way to organize their work. In contrast to traditional tools like sticky notes, cluttered spreadsheets, or overly complex enterprise software, PlanWise is designed to offer a streamlined, user-centric experience focused on clarity, accountability, and team coordination.

The platform will allow users to:

- Create, assign, and track tasks with ease
- Set deadlines and receive automated reminders
- Prioritize tasks by category or urgency
- Collaborate in real time with teammates
- Monitor overall project progress via a clean dashboard

To meet the diverse needs of modern users, PlanWise will also include features like:

- Role-based task assignments for better team structuring
- Progress tracking (e.g., “To Do”, “In Progress”, “Completed” stages)
- Visual indicators for deadlines and overdue tasks

The project is built using a **full-stack MERN approach** (MongoDB, Express.js, React.js, Node.js), ensuring scalability, flexibility, and a seamless user experience.

## **PRODUCT VISION :**

To build a smart, minimal, and collaborative task management platform that empowers students and teams to stay organized, focused, and ahead — without the clutter.

## **REQUIREMENT GATHERING AND USE CASE ANALYSIS:**

### **Modules to be implemented:**

#### **1. User Management Module**

- User Registration & Login
- Authentication using JWT
- Password hashing and security (e.g., bcrypt)
- User roles (Admin, Member)

#### **2. Task Management Module**

- Create, update, delete, and view tasks
- Set task priorities (High, Medium, Low)
- Assign tasks to users or groups
- Task deadlines and scheduling
- Task status tracking (To-Do, In Progress, Completed)

#### **3. Team Collaboration Module**

- Team creation and member management
- Role assignment within teams
- Share tasks across team members
- Commenting system on tasks (optional)

#### **4. Notification Module**

- Email alerts using **Nodemailer** for:
  - New task assigned
  - Upcoming deadlines
  - Task updates or comments
- Push notifications (if extended to mobile/PWA)

## **5. Dashboard & Analytics Module**

- Visual summary of tasks (Pie charts, Kanban view)
- Filter/sort tasks by deadline, priority, status, assignee
- Progress overview for individuals and teams

## **6. Search & Filter Module**

- Search tasks by title, tags, or description
- Filter tasks based on status, priority, due date, etc.

## **7. User Profile Module**

- View and edit profile
- See assigned/completed tasks
- Performance overview (optional)

## **8. Admin Module (for extended control)**

- Manage all users and teams
- Delete or reset tasks
- Oversee all dashboards

## **9. Responsive UI/UX Module**

- Built using React.js with Tailwind CSS or Material UI
- Mobile-first and accessible design

## **10. Database & API Integration Module**

- MongoDB schema design for tasks, users, teams
- RESTful APIs with Express.js for all CRUD operations
- Secure API endpoints with middleware for role validation

## USE CASE ANALYSIS :

### 1. User Registration/Login

GOAL	DESCRIPTION	EXAMPLE
Allow secure access to the platform	Users can sign up and securely log in using their credentials (JWT-based)	User creates an account with email and password to access their task dashboard

### 2. Create Task

GOAL	DESCRIPTION	EXAMPLE
Let users define and plan their own or team tasks	Users can create tasks with a title, description, deadline, and priority	A user creates a task titled "Presentation" with a deadline for Friday.

### 3. Assign Task

GOAL	DESCRIPTION	EXAMPLE
Enable collaboration by distributing tasks among teammates	Users or Admins can assign tasks to others within the same team	Admin assings "Backend Integeration" to a Developer,team member assings a task to peer

#### **4. Update Task Status**

GOAL	DESCRIPTION	EXAMPLE
Keep task progress transparent and up to date	Users can change task status to “To-Do,” “In Progress,” or “Completed”	User marks “Write Report” as Completed after submitting it.

#### **5. View Dashboard**

GOAL	DESCRIPTION	EXAMPLE
Provide a visual summary of tasks and team activity	Users can view current workload, deadlines, and completion rates	User sees 2 tasks overdue and 3 completed on their dashboard.

#### **6. Receive Notifications**

GOAL	DESCRIPTION	EXAMPLE
Keep users informed on task-related updates	System sends email alerts when tasks are assigned, updated, or due soon	User gets an email: “New task assigned by Admin: Design Wireframes.”

## 7. Manage Team

GOAL	DESCRIPTION	EXAMPLE
Allow admin to organize team structure	Admin can create teams, add/remove members, assign roles	Admin removes an inactive member and adds a new intern to the design team.

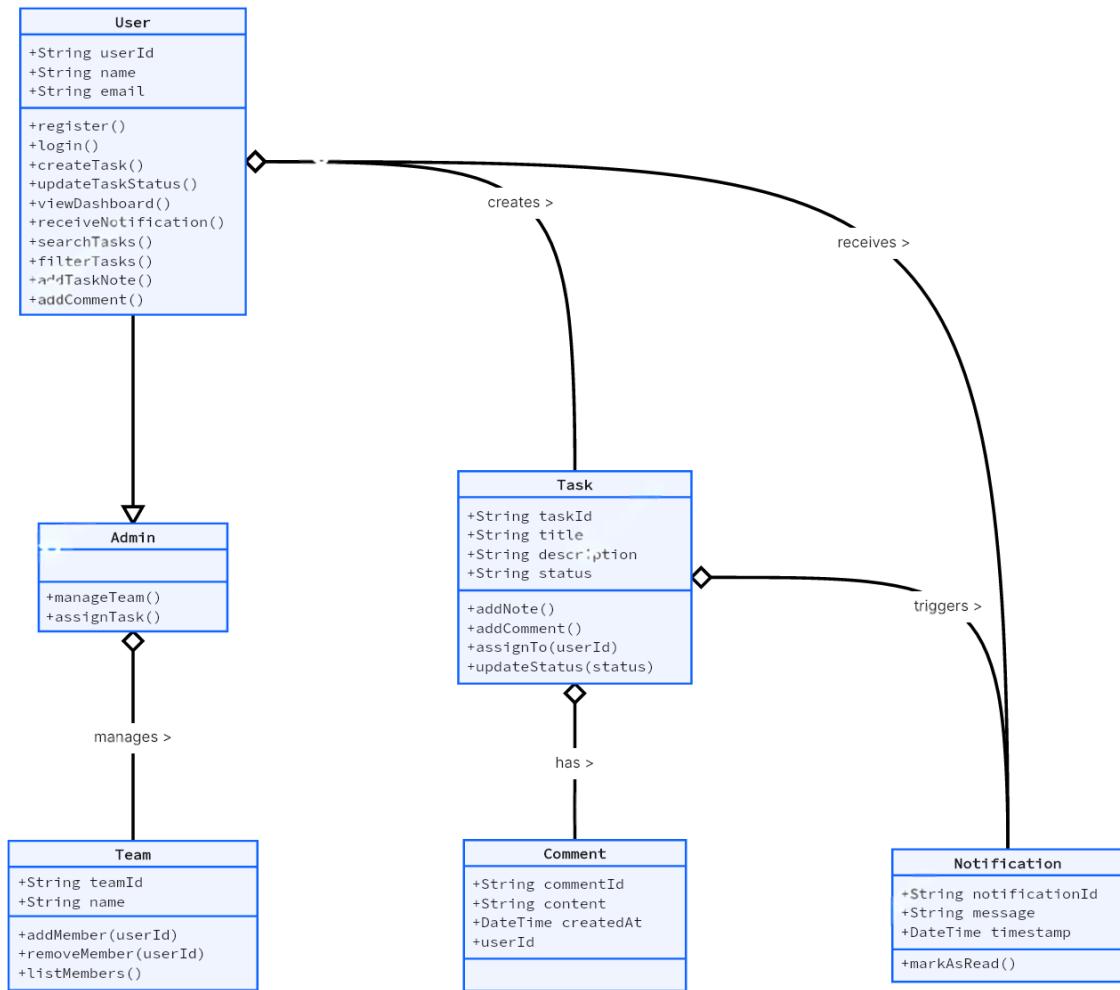
## 8. Search & Filter Tasks

GOAL	DESCRIPTION	EXAMPLE
Help users quickly find and organize relevant tasks	Users can filter tasks by status, priority, assignee, or search by keyword	User filters to see only “High Priority” tasks due this week.

## 9. Add Task Notes/Comments

GOAL	DESCRIPTION	EXAMPLE
Improve team communication within a task	Users can add comments, links, or updates to a task to keep everyone informed	User adds a note: “Waiting for client feedback before proceeding to final design.”

## USE CASE DIAGRAM



## FUNCTIONAL REQUIREMENTS – PLANWISE

### 1. User Management

- Users can register using a valid email address .
- Users can log in, log out, and reset their passwords securely.
- Roles are defined: Admin and Regular User.

### 2. Admin Dashboard

- Admin can manage users (approve, remove, or update roles).
- Admin can create/manage teams, assign tasks, and view team progress.

- Admin can oversee platform-wide activity logs and task metrics.

### **3. Task Management**

- Users can create, update, and delete tasks.
- Tasks include details: title, description, deadline, priority, tags.
- Tasks can be assigned to self or teammates (within the same team).
- Task status can be changed to: To-Do, In Progress, Completed.

### **4. Dashboard Overview**

- Users see a personalized dashboard with active, pending, and completed tasks.
- Admins view a global/team dashboard showing workloads and productivity insights.

### **5. Notes & Comments**

- Users can add comments or notes within a task.
- Notes support real-time collaboration and replies.
- Comments are linked to user profiles and include timestamps.

### **6. Search & Filter**

- Tasks can be searched by keywords.
- Filters include status, deadline, priority, assignee, and tags.
- Filters update task listings instantly.

### **7. Notifications**

- Users receive email alerts when:
  - A task is assigned or updated
  - A task is due soon
  - Someone replies to their comment or note

- OTP sent via email during sign-up and password reset.

## 8. Team Management

- Admins can create teams and assign users.
- Teams can view and manage tasks collaboratively.
- Admins can monitor team performance and activity.

# NON-FUNCTIONAL REQUIREMENTS – PLANWISE

## 1. Security

- Passwords are hashed with bcrypt.
- JWT-based authentication for secure login sessions.
- Role-based access restricts certain operations to Admins only.
- File uploads and inputs are validated to prevent security risks.

## 2. Usability

- Clean and responsive user interface (React-based).
- Mobile- and desktop-friendly layout.
- Clear task workflows with intuitive buttons and navigation.

## 3. Performance

- Tasks and dashboard data load within 2 seconds.
- Search and filtering respond in under 1 second.
- Optimized backend (Node.js + Express) ensures fast task processing.

## 4. Scalability

- Built using the MERN Stack for modular growth.
- APIs and frontend components are reusable and scalable.

- Designed to support growth from small teams to large cohorts.

## **5. Reliability & Availability**

- Target uptime: 99.9% with fallback handling.
- Retry logic for failed email or data submissions.
- Regular data backups and error logging.

## **6. Maintainability**

- Modular code structure with proper documentation and comments.
- Admin dashboard allows manual corrections or changes.
- Easily extensible for adding future features (like Calendar view).