# Software Requirements Specification (SRS)

**Project Name:** TaskNest
**Version:** 1.0
**Date:** October 2025

# 1. Introduction

## 1.1 Purpose

The purpose of this document is to define the requirements for **TaskNest**, a full-stack task management application designed for students, professionals, and anyone who needs a little help fighting procrastination. This SRS ensures that all stakeholders understand what TaskNest is, what it does, and how it should perform.

## 1.2 Scope

TaskNest is a **web based task management platform** where users can:

- Register and log in securely.
- Create, update, delete, and track tasks.
- Organize tasks by category and priority.
- Access the app on desktop, tablet, and mobile (responsive design).

Unlike sticky notes and random Google Docs, TaskNest provides a **centralized, user friendly hub** for productivity.

## 1.3 Definitions, Acronyms, and Abbreviations

- **SRS** – Software Requirement Specification.
- **CRUD** – Create, Read, Update, Delete.
- **Frontend** – User-facing interface built with React.
- **Backend** – Server-side API with Node.js & Express.
- **DB** – Database (MySQL in this case).

## 1.4 References

- TaskNest GitHub Repository
- Firebase Authentication Documentation
- React, Node.js, Express, MySQL official docs

# 2. Overall Description

## 2.1 Product Perspective

TaskNest is a **standalone application** built with modern full-stack technologies. It integrates Firebase for authentication and MySQL for persistent data storage. The system will follow **MVC principles** and RESTful API architecture.

## 2.2 Product Features

- **User Authentication:** Sign up, log in, and log out via Firebase.
- **Task Management:** Add, edit, delete, and mark tasks as completed.
- **Organization:** Categorize tasks, assign priorities, and filter them.
- **Responsive Design:** Works seamlessly across devices.
- **Security:** User data is encrypted and password-protected.

## 2.3 User Classes and Characteristics

- **Student Users:** Manage assignments, deadlines, and group projects.
- **Professional Users:** Manage meetings, daily goals, and work tasks.
- **General Users:** Track personal tasks like shopping lists, habits, or goals.

## 2.4 Operating Environment

- **Frontend:** React, CSS, HTML
- **Backend:** Node.js, Express.js
- **Database:** MySQL
- **Authentication:** Firebase

# 3. System Features

## 3.1 Authentication Module

- Register new users.
- Login/Logout functionality.
- Store user session securely.

## 3.2 Task Management Module

- Create tasks with title, description, due date, and priority.

- Update or delete tasks.
- Mark tasks as completed.
- Categorize tasks.

### 3.3 Dashboard & UI

- Intuitive and minimalistic design.
- Filter tasks by category, completion, or date.
- Responsive layout for mobile and desktop.

# 4. External Interface Requirements

### 4.1 User Interfaces

- Modern **React based UI** with Tailwind styling.
- Clear buttons for task operations (Add, Edit, Delete).
- Dashboard displaying upcoming and completed tasks.

### 4.2 Hardware Interfaces

- Runs on any device with an internet browser.

### 4.3 Software Interfaces

- **Frontend** communicates with **Backend** via REST API.
- **Backend** interacts with **MySQL database**.
- **Firebase API** handles authentication.

# 5. Non Functional Requirements

### 5.1 Performance Requirements

- Support up to 1,000 concurrent users.
- Task operations should execute within 2 seconds.

### 5.2 Security Requirements

- Use Firebase Authentication and encrypted passwords.
- Secure REST API endpoints with authentication checks.

### 5.3 Reliability

- 99% uptime expected.
- Automatic database backup once daily.

### 5.4 Usability

- Intuitive UI, minimal learning curve.
- Accessible across devices and browsers.

## 6. Future Enhancements

- Dark mode for night owls. 🌙
- Collaboration features (shared task boards).
- Notifications & reminders.
- AI-powered productivity assistant.

## 7. Appendix

**Why "TaskNest"?**
Because just like birds carefully build their nests one twig at a time, users build their productivity one task at a time.