



TaskBuddy Application

Problemática que soluciona

En muchos entornos personales, académicos y laborales persisten problemas como:

a) Falta de organización y priorización de tareas

Las personas suelen manejar tareas en notas dispersas, libretas físicas o mensajes, lo que provoca:

- Desorden
- Pérdida de tareas
- Falta de seguimiento
- Olvido de plazos
- Cero trazabilidad

b) Dificultad para gestionar tareas desde múltiples dispositivos

Muchos sistemas no permiten acceder a las tareas desde lugares distintos, por lo que se pierde el control cuando se cambia de dispositivo.

c) Ausencia de un sistema seguro y privado

La mayoría de usuarios desean que sus tareas sean **privadas**, pero muchas aplicaciones gratuitas:

- No usan autenticación real
- No implementan JWT
- No protegen la información del usuario

d) Falta de funcionalidades básicas pero críticas

El usuario promedio necesita:

- Crear tareas
- Editarlas
- Marcar como completadas
- Eliminarlas
- Consultarlas por usuario

Más que funciones avanzadas, lo que falta es una herramienta **sólida, estable y personalizable**.

Tu aplicación resuelve exactamente estas necesidades.

2. A quién va dirigido

Tu plataforma está diseñada para:

Usuarios individuales

Estudiantes, trabajadores o personas que necesiten una herramienta **simple pero segura** para gestionar sus tareas diarias.

Profesionales o pequeños equipos

Quienes requieren una app rápida, accesible y con autenticación para mantener control de actividades propias.

Desarrolladores o empresas que quieran integrarlo

La API REST puede servir como base para:

- Integrarlo con apps móviles
- Conectarlo a dashboards
- Usarlo en aplicaciones empresariales más grandes

Personas que desean un sistema confiable sin depender de terceros

Al ser una solución completa (Backend + Frontend), permite que **cualquiera pueda hostearlo** sin depender de servicios externos.

3. Futuras mejoras (Escalabilidad)

Como buena base, este sistema es **altamente escalable**. Estas son las mejoras viables y naturales:

A) Escalabilidad funcional (mejoras de funciones)

1. Sistema de prioridades por tarea

Alta, media o baja.

2. Fechas de vencimiento y recordatorios

Implementación de:

- Deadline
- Notificaciones
- Alarmas

3. Subtareas y checklist

Crear tareas anidadas.

4. Categorías o etiquetas

Permite agrupar tareas por temas o proyectos.

5. Historial de cambios

Auditoría por tarea:

- Creación
- Edición
- Completado
- Eliminación

B) Escalabilidad técnica

1. Migrar a una arquitectura por capas o Clean Architecture

Separar:

- Dominio
- Aplicación
- Infraestructura
- Presentación

Esto mejora mantenibilidad y testing.

2. Implementar Refresh Tokens

Autenticación más robusta:

- Previene secuestro de tokens
- Más seguridad a largo plazo

3. Soporte para múltiples bases de datos

SQLite → PostgreSQL / SQL Server / MySQL

Sin cambiar el código principal.

4. Contenedorización (Docker)

Permite desplegar el sistema en:

- Servidores
- Nubes
- Kubernetes

5. Escalar horizontalmente la API

Usando:

- Load balancers
- Microservicios
- Clustering

C) Escalabilidad producto / negocio

1. Panel administrativo

Supervisión de:

- Usuarios
- Tareas
- Estadísticas

2. Generación de reportes avanzados

Más allá del ReportController.

3. Integración con herramientas externas

Como Google Calendar, Notion, Slack, Trello, etc.

4. Aplicación móvil

React Native

Flutter

MAUI