

# Documentación del Proyecto: Herramienta de Tareas

David Mata Guerra  
Alejandro Macías Fonseca  
Alexis Emilio Cárdenas Camacho  
Gabriel Juárez Ramírez  
Universidad Autónoma de Querétaro

5 de noviembre de 2024

# Índice general

<b>1. Descripción General del Proyecto</b>	<b>3</b>
1.1. Resumen . . . . .	3
1.2. Alcance y Limitaciones . . . . .	3
1.3. Justificación . . . . .	4
<b>2. Especificación de Requerimientos</b>	<b>5</b>
2.1. Requerimientos Funcionales . . . . .	5
2.2. Requerimientos No Funcionales . . . . .	5
2.3. Casos de Uso . . . . .	5
<b>3. Análisis y Diseño del Sistema</b>	<b>6</b>
3.1. Diagrama de Arquitectura . . . . .	6
3.2. Diagramas UML . . . . .	6
3.3. Modelo de Datos . . . . .	6
3.4. Especificación de Interfaz . . . . .	6
<b>4. Planificación y Gestión del Proyecto</b>	<b>7</b>
4.1. Cronograma . . . . .	7
4.2. Recursos y Roles . . . . .	7
4.3. Control de Versiones . . . . .	7
<b>5. Implementación</b>	<b>8</b>
5.1. Estructura del Código . . . . .	8
5.2. Convenciones de Codificación . . . . .	8
5.3. Dependencias . . . . .	8
<b>6. Pruebas y Validación</b>	<b>9</b>
6.1. Plan de Pruebas . . . . .	9
6.2. Casos de Prueba . . . . .	9
6.3. Registro de Errores . . . . .	9
<b>7. Manual de Usuario</b>	<b>10</b>
7.1. Guía de Uso . . . . .	10
7.2. Resolución de Problemas Comunes . . . . .	10
<b>8. Manual Técnico o de Mantenimiento</b>	<b>11</b>
8.1. Instrucciones de Mantenimiento . . . . .	11
8.2. Recuperación ante Fallos . . . . .	11

<b>9. Documentación de Despliegue</b>	<b>12</b>
9.1. Instrucciones de Instalación . . . . .	12
9.2. Requisitos de Hardware y Software . . . . .	12
<b>10.Licencias y Acuerdos</b>	<b>13</b>
<b>11.Registro de Cambios (Changelog)</b>	<b>14</b>

# Capítulo 1

## Descripción General del Proyecto

### 1.1. Resumen

El propósito de este proyecto es crear una herramienta de tareas, la cual pueda crear, editar, eliminar y marcar como completadas las tareas que el usuario desee. Además, se podrá visualizar las tareas que se han completado y las que están pendientes.

### 1.2. Alcance y Limitaciones

Alcances del proyecto:

- Gestionar el proyecto mediante la herramienta JIRA utilizando la metodología Scrum.
- Desarrollar un prototipado con todas las funcionalidades que se deen implementar en Figma.
- Hacer uso un repositorio en GitHub para el control de versiones de el frontend.
- Hacer uso repositorio en GitHub para el control de versiones de el backend.
- Crear un diagrama relacional de la base de datos.
- Crear una base de datos con la estructura del diagrama relacional.
- Desarrollar la aplicación.
- Lanzar la aplicación en un servidor.
- Realizar test a la aplicación.
- Desglosar los pasos del proyecto en una presentación.

Limitaciones del proyecto:

- El desarrollo de el proyecto tiene un tiempo limitado (3 semanas y 4 días).
- El proyecto no busca fines lucrativos.
- El proyecto se encuentra limitado a la creación de tareas.

### **1.3. Justificación**

El proyecto surge en base a un proyecto final solicitado en la clase de Ingeniería de Requerimientos de la Universidad Autónoma de Querétaro. La idea de crear un gestor de tareas es con la finalidad de poner a prueba los conocimientos adquiridos en la materia y en la carrera.

# Capítulo 2

## Especificación de Requerimientos

### 2.1. Requerimientos Funcionales

- Gestión de tareas:
  - Crear tareas.
  - Editar tareas.
  - Eliminar tareas.
  - Marcar tareas como completadas.
- Visualización de tareas:
  - Interfaz intuitiva y práctica de utilizar.
  - El usuario pueda reconocer todos los cambios realizados de manera visual.

### 2.2. Requerimientos No Funcionales

- Rendimiento:
  - No debe haber errores en la lógica de programación.
  - La aplicación debe ser rápida y eficiente.

### 2.3. Casos de Uso

Incluya diagramas de casos de uso y descripciones.

# Capítulo 3

## Análisis y Diseño del Sistema

### 3.1. Diagrama de Arquitectura

Incluya un diagrama de arquitectura del sistema.

### 3.2. Diagramas UML

Diagramas de clases, de secuencia, o de actividad según se necesite.

### 3.3. Modelo de Datos

Describa el modelo de base de datos y los diagramas entidad-relación.

### 3.4. Especificación de Interfaz

Describa la interfaz de usuario y la experiencia del usuario.

# Capítulo 4

## Planificación y Gestión del Proyecto

### 4.1. Cronograma

Incluya un cronograma o diagrama de Gantt.

### 4.2. Recursos y Roles

Defina los recursos necesarios y los roles del equipo.

### 4.3. Control de Versiones

Describa cómo se gestionará el control de versiones.



# Capítulo 5

## Implementación

### 5.1. Estructura del Código

Describe la organización del código.

### 5.2. Convenciones de Codificación

Incluya las convenciones de codificación que sigue el proyecto.

### 5.3. Dependencias

Liste las dependencias del proyecto (librerías, frameworks, etc.).

# Capítulo 6

## Pruebas y Validación

### 6.1. Plan de Pruebas

Describe las estrategias de prueba.

### 6.2. Casos de Prueba

Liste y describa casos de prueba específicos.

### 6.3. Registro de Errores

Proporcione un formato o sistema para el registro de errores.

# Capítulo 7

## Manual de Usuario

### 7.1. Guía de Uso

Instrucciones para el usuario final.

### 7.2. Resolución de Problemas Comunes

Resuelva dudas o problemas frecuentes.

# Capítulo 8

## Manual Técnico o de Mantenimiento

### 8.1. Instrucciones de Mantenimiento

Cómo mantener y actualizar el sistema.

### 8.2. Recuperación ante Fallos

Procedimientos para recuperación y mantenimiento.

# Capítulo 9

## Documentación de Despliegue

### 9.1. Instrucciones de Instalación

Cómo instalar y desplegar el sistema.

### 9.2. Requisitos de Hardware y Software

Describe los requisitos del entorno de producción.

# Capítulo 10

## Licencias y Acuerdos

Incluya licencias de software y derechos de propiedad intelectual.

# Capítulo 11

## Registro de Cambios (Changelog)

Documente los cambios y versiones del sistema.