

# PLAN DE TRABAJO

---

## Sistema de Gestión Documental Universitaria TESCHI

---

### CARÁTULA

**INSTITUCIÓN:** Tecnológico de Estudios Superiores de Chimalhuacán (TESCHI)  
**PROYECTO:** Sistema de Gestión Documental Universitaria  
**MATERIA:** Ingeniería de Software  
**DOCENTE:** Modesto Castro Yolanda  
**SEMESTRE:** 7ISC23

### INTEGRANTES DEL EQUIPO:

- 1. **Gálvez Romero Irvin Osvaldo** - Administrador de Base de Datos / Desarrollador Full-Stack
- 2. **Cruz Contreras Ángel Valentín** - Desarrollador Frontend / UI/UX
- 3. **Sánchez Vargas Kevin Antonio** - Analista de Sistemas / Desarrollador Backend
- 4. **Juárez Vargas Alberto** - Líder de Proyecto / DevOps

**FECHA:** [Fecha actual]  
**VERSIÓN:** 1.0

---

## 1. RESUMEN EJECUTIVO

### 1.1 Objetivo del Proyecto

Desarrollar e implementar un Sistema de Gestión Documental Universitaria para el TESCHI que permita la administración eficiente de documentos académicos, gestión de estudiantes y control de procesos administrativos de la institución.

### 1.2 Alcance del Proyecto

- Desarrollo de aplicación web full-stack
- Implementación de sistema de autenticación y autorización
- Gestión de usuarios (estudiantes, profesores, administradores)
- Sistema de subida y revisión de documentos
- Dashboard administrativo con estadísticas
- Sistema de notificaciones por correo electrónico

### 1.3 Duración del Proyecto

**16 semanas** (4 meses)

### 1.4 Presupuesto Estimado

- **Recursos humanos:** \$0 (proyecto académico)

- **Infraestructura:** \$500 USD
  - **Herramientas de desarrollo:** \$200 USD
  - **Total estimado:** \$700 USD
- 

## 2. ESTRUCTURA ORGANIZACIONAL

### 2.1 Roles y Responsabilidades

#### Líder de Proyecto - Irvin Osvaldo

- **Responsabilidades:**
  - Coordinación general del proyecto
  - Gestión de cronograma y recursos
  - Comunicación con stakeholders
  - Toma de decisiones técnicas
  - Desarrollo full-stack
- **Habilidades requeridas:** Liderazgo, gestión de proyectos, desarrollo web

#### Analista de Sistemas / Backend - Ángel Valentín

- **Responsabilidades:**
  - Análisis de requerimientos
  - Diseño de arquitectura
  - Desarrollo del backend
  - Diseño de base de datos
  - Integración de APIs
- **Habilidades requeridas:** Análisis de sistemas, Node.js, bases de datos, APIs REST

#### Desarrollador Frontend / UI/UX - Kevin Antonio

- **Responsabilidades:**
  - Diseño de interfaz de usuario
  - Desarrollo del frontend
  - Experiencia de usuario
  - Integración con backend
  - Testing de interfaz
- **Habilidades requeridas:** React, TypeScript, UI/UX, CSS, JavaScript

#### Administrador de BD / DevOps - Alberto

- **Responsabilidades:**
  - Administración de base de datos
  - Configuración de servidores
  - Despliegue y mantenimiento
  - Monitoreo del sistema
  - Backup y seguridad
- **Habilidades requeridas:** Bases de datos, Linux, Docker, CI/CD, seguridad

## 2.2 Estructura de Comunicación

- **Reuniones diarias:** 15 minutos (stand-up)
  - **Reuniones semanales:** 1 hora (revisión de progreso)
  - **Reuniones de sprint:** 2 horas (planificación y retrospectiva)
  - **Comunicación continua:** Slack/Teams para coordinación
- 

## 3. CRONOGRAMA DETALLADO

### FASE 1: PLANIFICACIÓN Y ANÁLISIS (Semanas 1-2)

#### Semana 1: Inicio del Proyecto

##### Objetivos:

- Definir alcance y objetivos
- Establecer metodología de trabajo
- Configurar herramientas de desarrollo

##### Actividades:

- ☐ Reunión de kick-off del proyecto
- ☐ Definición de metodología ágil
- ☐ Configuración de repositorio Git
- ☐ Establecimiento de estándares de código
- ☐ Creación de estructura de proyecto

##### Entregables:

- Documento de alcance del proyecto
- Repositorio Git configurado
- Estándares de desarrollo definidos

**Responsables:** Todo el equipo

#### Semana 2: Análisis de Requerimientos

##### Objetivos:

- Recopilar y analizar requerimientos
- Definir casos de uso
- Diseñar arquitectura del sistema

##### Actividades:

- ☐ Entrevistas con stakeholders
- ☐ Análisis de procesos actuales
- ☐ Definición de casos de uso
- ☐ Diseño de arquitectura
- ☐ Especificación de requerimientos

**Entregables:**

- Especificación de requerimientos
- Casos de uso definidos
- Diseño de arquitectura
- Plan de pruebas

**Responsables:** Ángel (Líder), Irvin, Kevin, Alberto

**FASE 2: DISEÑO Y CONFIGURACIÓN (Semanas 3-4)****Semana 3: Diseño Detallado****Objetivos:**

- Completar diseño de sistema
- Definir estructura de base de datos
- Diseñar interfaces de usuario

**Actividades:**

- ☐ Diseño detallado de base de datos
- ☐ Diseño de API REST
- ☐ Wireframes de interfaz de usuario
- ☐ Diseño de flujos de trabajo
- ☐ Especificación técnica detallada

**Entregables:**

- Modelo de base de datos
- Especificación de API
- Wireframes de UI
- Documento de diseño técnico

**Responsables:** Ángel (BD), Kevin (UI), Irvin (Arquitectura), Alberto (Infraestructura)

**Semana 4: Configuración del Entorno****Objetivos:**

- Configurar entorno de desarrollo
- Implementar infraestructura base
- Establecer pipeline de CI/CD

**Actividades:**

- ☐ Configuración de servidor de desarrollo
- ☐ Configuración de base de datos
- ☐ Configuración de herramientas de testing
- ☐ Implementación de pipeline CI/CD
- ☐ Configuración de monitoreo

**Entregables:**

- Entorno de desarrollo funcional
- Base de datos configurada
- Pipeline de CI/CD
- Documentación de configuración

**Responsables:** Alberto (Líder), Ángel, Irvin, Kevin

**FASE 3: DESARROLLO BACKEND (Semanas 5-8)****Semana 5: Módulo de Autenticación****Objetivos:**

- Implementar sistema de autenticación
- Configurar seguridad básica
- Implementar gestión de usuarios

**Actividades:**

- ☐ Implementación de JWT
- ☐ Sistema de registro de usuarios
- ☐ Sistema de login/logout
- ☐ Middleware de autenticación
- ☐ Gestión de roles y permisos

**Entregables:**

- API de autenticación funcional
- Sistema de usuarios implementado
- Middleware de seguridad
- Pruebas unitarias

**Responsables:** Ángel (Líder), Irvin, Alberto

**Semana 6: Módulo de Gestión de Estudiantes****Objetivos:**

- Implementar CRUD de estudiantes
- Sistema de búsqueda y filtrado
- Gestión de carreras

**Actividades:**

- ☐ API de gestión de estudiantes
- ☐ Sistema de búsqueda avanzada
- ☐ Gestión de carreras académicas
- ☐ Validaciones de negocio
- ☐ Pruebas de integración

**Entregables:**

- API de estudiantes completa
- Sistema de carreras
- Pruebas de integración
- Documentación de API

**Responsables:** Ángel (Líder), Irvin, Alberto

**Semana 7: Módulo de Gestión de Documentos****Objetivos:**

- Implementar subida de archivos
- Sistema de revisión de documentos
- Gestión de metadatos

**Actividades:**

- ☐ API de subida de archivos
- ☐ Sistema de revisión de documentos
- ☐ Gestión de metadatos
- ☐ Validación de archivos
- ☐ Sistema de notificaciones

**Entregables:**

- API de documentos funcional
- Sistema de archivos
- Notificaciones por correo
- Pruebas de carga

**Responsables:** Ángel (Líder), Irvin, Alberto

**Semana 8: APIs de Dashboard y Reportes****Objetivos:**

- Implementar APIs de estadísticas
- Sistema de reportes
- Optimización de consultas

**Actividades:**

- ☐ API de estadísticas
- ☐ Sistema de reportes
- ☐ Optimización de consultas
- ☐ Caché de datos
- ☐ Pruebas de rendimiento

**Entregables:**

- APIs de dashboard
- Sistema de reportes
- Optimizaciones implementadas
- Pruebas de rendimiento

**Responsables:** Ángel (Líder), Irvin, Alberto

## FASE 4: DESARROLLO FRONTEND (Semanas 9-12)

### Semana 9: Configuración y Componentes Base

#### Objetivos:

- Configurar React con TypeScript
- Implementar componentes base
- Configurar routing

#### Actividades:

- ☐ Configuración de React + TypeScript
- ☐ Configuración de Material-UI
- ☐ Implementación de componentes base
- ☐ Configuración de routing
- ☐ Sistema de autenticación frontend

#### Entregables:

- Aplicación React configurada
- Componentes base implementados
- Sistema de routing
- Autenticación frontend

**Responsables:** Kevin (Líder), Irvin, Ángel

### Semana 10: Páginas de Autenticación y Dashboard

#### Objetivos:

- Implementar páginas de login
- Dashboard de administrador
- Dashboard de estudiante

#### Actividades:

- ☐ Página de login/registro
- ☐ Dashboard de administrador
- ☐ Dashboard de estudiante
- ☐ Dashboard de profesor
- ☐ Sistema de navegación

#### Entregables:

- Páginas de autenticación
- Dashboards implementados
- Navegación funcional
- Pruebas de interfaz

**Responsables:** Kevin (Líder), Irvin, Ángel

## **Semana 11: Gestión de Estudiantes y Documentos**

### **Objetivos:**

- Páginas de gestión de estudiantes
- Sistema de subida de documentos
- Páginas de revisión

### **Actividades:**

- ☐ Páginas de gestión de estudiantes
- ☐ Formularios de registro
- ☐ Sistema de subida de archivos
- ☐ Páginas de revisión de documentos
- ☐ Sistema de búsqueda

### **Entregables:**

- Gestión de estudiantes completa
- Sistema de documentos funcional
- Búsqueda implementada
- Pruebas de usabilidad

**Responsables:** Kevin (Líder), Irvin, Ángel

## **Semana 12: Integración y Optimización**

### **Objetivos:**

- Integrar frontend con backend
- Optimizar rendimiento
- Implementar mejoras de UX

### **Actividades:**

- ☐ Integración completa frontend-backend
- ☐ Optimización de rendimiento
- ☐ Mejoras de experiencia de usuario
- ☐ Implementación de caché
- ☐ Pruebas de integración

### **Entregables:**

- Sistema integrado completo



- Optimizaciones implementadas
- Mejoras de UX
- Pruebas de integración

**Responsables:** Kevin (Líder), Irvin, Ángel, Alberto

## FASE 5: PRUEBAS Y CALIDAD (Semanas 13-14)

### Semana 13: Pruebas del Sistema

#### Objetivos:

- Ejecutar pruebas completas
- Corregir bugs encontrados
- Validar funcionalidad

#### Actividades:

- ☐ Pruebas unitarias completas
- ☐ Pruebas de integración
- ☐ Pruebas de sistema
- ☐ Pruebas de usabilidad
- ☐ Corrección de bugs

#### Entregables:

- Reportes de pruebas
- Bugs corregidos
- Sistema estable
- Documentación de pruebas

**Responsables:** Todo el equipo

### Semana 14: Pruebas de Aceptación

#### Objetivos:

- Pruebas con usuarios finales
- Validación de requerimientos
- Preparación para producción

#### Actividades:

- ☐ Pruebas con usuarios reales
- ☐ Validación de casos de uso
- ☐ Pruebas de carga
- ☐ Pruebas de seguridad
- ☐ Preparación de documentación

#### Entregables:

- Reporte de aceptación

- Sistema validado
- Documentación de usuario
- Plan de despliegue

**Responsables:** Todo el equipo

## FASE 6: DESPLIEGUE Y ENTREGA (Semanas 15-16)

### Semana 15: Despliegue en Producción

#### Objetivos:

- Desplegar sistema en producción
- Configurar monitoreo
- Capacitar usuarios

#### Actividades:

- ☐ Configuración de servidor de producción
- ☐ Despliegue del sistema
- ☐ Configuración de monitoreo
- ☐ Capacitación de usuarios
- ☐ Pruebas en producción

#### Entregables:

- Sistema en producción
- Monitoreo configurado
- Usuarios capacitados
- Documentación de despliegue

**Responsables:** Alberto (Líder), Todo el equipo

### Semana 16: Entrega y Cierre

#### Objetivos:

- Entregar sistema completo
- Documentar lecciones aprendidas
- Planificar mantenimiento

#### Actividades:

- ☐ Entrega final del sistema
- ☐ Documentación completa
- ☐ Retrospectiva del proyecto
- ☐ Plan de mantenimiento
- ☐ Celebración del equipo

#### Entregables:

- Sistema entregado

- Documentación completa
- Plan de mantenimiento
- Informe final del proyecto

**Responsables:** Todo el equipo

---

## 4. RECURSOS REQUERIDOS

### 4.1 Recursos Humanos

- **4 Desarrolladores** (tiempo completo)
- **1 Líder de Proyecto** (tiempo completo)
- **1 Analista de Sistemas** (tiempo completo)
- **1 Especialista en UI/UX** (tiempo completo)
- **1 Administrador de Sistemas** (tiempo completo)

### 4.2 Recursos Técnicos

- **Servidor de desarrollo:** 1 unidad
- **Servidor de producción:** 1 unidad
- **Base de datos:** SQLite/PostgreSQL
- **Almacenamiento:** 1TB mínimo
- **Ancho de banda:** 100Mbps

### 4.3 Herramientas de Desarrollo

- **IDE:** Visual Studio Code
- **Control de versiones:** Git + GitHub
- **Gestión de proyectos:** Jira/Trello
- **Comunicación:** Slack/Teams
- **Testing:** Jest, Cypress
- **CI/CD:** GitHub Actions

### 4.4 Licencias y Software

- **Sistema operativo:** Linux (Ubuntu)
  - **Base de datos:** PostgreSQL (open source)
  - **Servidor web:** Nginx (open source)
  - **Runtime:** Node.js (open source)
  - **Framework:** React (open source)
- 

## 5. GESTIÓN DE RIESGOS

### 5.1 Riesgos Técnicos

- **Riesgo:** Problemas de integración frontend-backend
- **Probabilidad:** Media

- **Impacto:** Alto
- **Mitigación:** Pruebas de integración continuas, comunicación constante
- **Riesgo:** Problemas de rendimiento con archivos grandes
- **Probabilidad:** Media
- **Impacto:** Medio
- **Mitigación:** Optimización de código, pruebas de carga

## 5.2 Riesgos de Recursos

- **Riesgo:** Disponibilidad limitada de miembros del equipo
- **Probabilidad:** Baja
- **Impacto:** Alto
- **Mitigación:** Distribución de responsabilidades, documentación detallada

## 5.3 Riesgos de Tiempo

- **Riesgo:** Retrasos en entregas
- **Probabilidad:** Media
- **Impacto:** Medio
- **Mitigación:** Buffer de tiempo, priorización de funcionalidades

---

# 6. MÉTRICAS DE ÉXITO

## 6.1 Métricas de Calidad

- **Cobertura de pruebas:** > 80%
- **Bugs críticos:** 0
- **Tiempo de respuesta:** < 3 segundos
- **Disponibilidad:** > 99%

## 6.2 Métricas de Progreso

- **Entregas a tiempo:** 100%
- **Funcionalidades completadas:** 100%
- **Documentación actualizada:** 100%
- **Satisfacción del cliente:** > 4.5/5

## 6.3 Métricas de Equipo

- **Productividad del equipo:** Mantener ritmo constante
  - **Comunicación:** Reuniones efectivas
  - **Colaboración:** Trabajo en equipo fluido
  - **Aprendizaje:** Nuevas habilidades adquiridas
-

## 7. COMUNICACIÓN Y REPORTES

### 7.1 Reuniones Regulares

- **Daily Stand-up:** 15 min diarios
- **Sprint Planning:** 2 horas cada 2 semanas
- **Sprint Review:** 1 hora cada 2 semanas
- **Retrospectiva:** 1 hora cada 2 semanas

### 7.2 Reportes de Progreso

- **Reporte semanal:** Estado del proyecto
- **Reporte de hitos:** Entregables completados
- **Reporte de riesgos:** Identificación y mitigación
- **Reporte final:** Resumen del proyecto

### 7.3 Comunicación con Stakeholders

- **Presentaciones mensuales:** Progreso del proyecto
  - **Demostraciones:** Funcionalidades implementadas
  - **Feedback:** Incorporación de comentarios
  - **Capacitación:** Entrenamiento de usuarios
- 

## 8. PLAN DE CONTINGENCIA

### 8.1 Escenarios de Contingencia

- **Retraso en desarrollo:** Priorizar funcionalidades críticas
- **Problemas técnicos:** Consultar expertos externos
- **Cambios de requerimientos:** Evaluar impacto y ajustar cronograma
- **Problemas de recursos:** Redistribuir responsabilidades

### 8.2 Plan de Recuperación

- **Identificación temprana:** Monitoreo constante
  - **Acción inmediata:** Resolución rápida de problemas
  - **Comunicación:** Informar a stakeholders
  - **Ajustes:** Modificar plan según necesidad
- 

## 9. LECCIONES APRENDIDAS

### 9.1 Mejores Prácticas

- Comunicación constante entre miembros del equipo
- Pruebas continuas durante el desarrollo
- Documentación actualizada en tiempo real
- Uso de herramientas de gestión de proyectos

## 9.2 Áreas de Mejora

- Estimación más precisa de tiempos
- Mejor gestión de cambios de requerimientos
- Implementación de más automatizaciones
- Mejores prácticas de testing

---

**Documento elaborado por:** Equipo de Desarrollo TESCHI

**Fecha de creación:** [Fecha actual]

**Última actualización:** [Fecha actual]

**Próxima revisión:** [Fecha + 1 mes]