

TutorIA: Tutor Inteligente Adaptativo para Jóvenes con TDA y Dificultades de Aprendizaje

1. Contexto Teórico del Sistema

Fundamentos Pedagógicos:

Aprendizaje Personalizado: Adaptación del contenido según el perfil cognitivo individual

Teoría de las Inteligencias Múltiples (Gardner): Desarrollo de diferentes tipos de inteligencia

Estilos de Aprendizaje VARK: Visual, Auditivo, Lectura/Escritura, Kinestésico

Aprendizaje Adaptativo: Modificación dinámica del contenido según el progreso

Fundamentos de IA y Machine Learning:

Knowledge Tracing: Seguimiento del conocimiento del estudiante a lo largo del tiempo

RAG (Retrieval-Augmented Generation): Recuperación y generación de contenido contextual

LLM Personalizado: Modelo de lenguaje adaptado para tutoría educativa

Análisis de Patrones Cognitivos: Detección de patrones de atención y aprendizaje

2. Alcance del Sistema

Objetivo Principal:

Desarrollar un tutor inteligente que proporcione apoyo educativo personalizado a niños y adolescentes con TDA y dislexia, adaptándose a sus necesidades cognitivas específicas y facilitando el aprendizaje en múltiples materias escolares.

Alcance Educativo:

5 materias principales: Matemáticas, Lenguaje/Español, Ciencias Naturales, Ciencias Sociales, Inglés

Niveles: Educación primaria y secundaria (7-16 años)

Modalidad: Aprendizaje desde casa con supervisión parental

Alcance Tecnológico:

Plataforma web accesible y responsive

Sistema de IA con capacidades de adaptación cognitiva

Base de datos para seguimiento de progreso

Interfaz multimodal (texto, audio, visual)

3. Población Objetivo

Características Demográficas:

Edad: 7-16 años (educación primaria y secundaria)

Condición: Niños y adolescentes con diagnóstico de TDA/TDAH

Condición adicional: Dificultades de aprendizaje, especialmente dislexia

Contexto: Uso desde el hogar con supervisión familiar

Necesidades Específicas:

TDA/TDAH: Dificultades de atención, hiperactividad, impulsividad

Dislexia: Dificultades en lectura, escritura y procesamiento del lenguaje

Dificultades de aprendizaje: Problemas en matemáticas, comprensión lectora

Necesidad de estructura: Requieren rutinas y organización clara

4. Funcionalidades del Sistema

4.1 Detección y Adaptación Cognitiva

Análisis de patrones de atención: Monitoreo de tiempo de concentración

Detección de distracciones: Identificación de momentos de pérdida de atención

Adaptación de velocidad: Ajuste dinámico del ritmo de presentación

Detección de momentos óptimos: Identificación de horarios de mayor productividad

Personalización de formato: Adaptación según preferencias VARK

4.2 Soporte Especializado para Dislexia

Fuentes especializadas: Uso de tipografías optimizadas (OpenDyslexic, Lexie Readable)

Conversión texto-a-voz: Lectura automática de contenido

Espaciado optimizado: Ajuste de interlineado y márgenes

Ejercicios de lectura adaptativos: Actividades graduales de comprensión

Soporte visual: Uso de colores y elementos gráficos para facilitar la lectura

4.3 Apoyo en Dificultades Matemáticas

Visualización de conceptos: Representación gráfica de problemas abstractos

Explicaciones paso a paso: Desglose detallado de procedimientos

Uso de colores: Diferenciación visual de conceptos matemáticos

Gamificación matemática: Juegos y actividades lúdicas

Ejercicios adaptativos: Dificultad ajustada según el progreso

4.4 Tutoría Multimateria

Adaptación por materia: Contenido específico para cada asignatura

Conexiones interdisciplinarias: Relación entre conceptos de diferentes materias

Recomendaciones personalizadas: Recursos específicos según necesidades

Progreso integrado: Seguimiento unificado del aprendizaje

4.5 Sistema de Seguimiento y Reportes

Knowledge Tracing: Mapeo del conocimiento del estudiante

Reportes para padres: Informes detallados del progreso

Comunicación con maestros: Compartir avances con el sistema escolar

Recomendaciones personalizadas: Sugerencias basadas en el análisis de datos

4.6 Interfaz Adaptativa

Diseño accesible: Interfaz optimizada para necesidades especiales

Elementos no distractores: Diseño limpio y enfocado

Navegación intuitiva: Estructura clara y fácil de usar

Responsive design: Adaptación a diferentes dispositivos

5. Beneficios Esperados

Para los Estudiantes:

Mejora en la autoestima académica

Desarrollo de estrategias de aprendizaje efectivas

Progreso académico más consistente

Mayor engagement con el aprendizaje

Para los Padres:

Visibilidad del progreso académico

Herramientas para apoyar el aprendizaje en casa

Reducción de la ansiedad sobre el rendimiento escolar

Comunicación mejorada con el sistema educativo

Para el Sistema Educativo:

Complemento a la educación tradicional

Datos valiosos sobre patrones de aprendizaje

Herramienta de apoyo para maestros

Inclusión educativa mejorada.

Por el tiempo que tenemos y solo somos 4

MVP Mínimo Viable:

Lo que si debe funcionar:

1. Estudiante puede hacer ejercicios
2. Sistema trackea progreso
3. Recomienda ejercicios apropiados
4. Interfaz es accesible para dislexia
5. Muestra progreso visual

Lo que puede esperar:

1. Análisis cognitivo avanzado
2. Sistema de autenticación
3. Reportes complejos
4. Integración con maestros

OBJETIVOS DIARIOS - TUTORIA MVP

4 Objetivos por Día | 6 Días | 4 Personas

DÍA 1: SETUP + KNOWLEDGE TRACING BÁSICO

Descripción: Configurar las bases del sistema educativo inteligente e implementar el algoritmo fundamental que permite al tutor "entender" el progreso de cada estudiante para adaptar el contenido de manera personalizada.

Entorno de Desarrollo Configurado

Responsable: Persona 1 (Backend + ML) Descripción: Establecer el ecosistema tecnológico que soportará el tutor inteligente, incluyendo las herramientas de Machine Learning necesarias para el análisis cognitivo de los estudiantes. Actividades:

- Configurar Python, FastAPI, scikit-learn
- Crear estructura de carpetas del proyecto
- Implementar modelo básico de Knowledge Tracing

Interfaz Base Accesible

Responsable: Persona 2 (Frontend) Descripción: Crear la base visual del sistema con características de accesibilidad específicas para estudiantes con dislexia, implementando fuentes y colores que faciliten la lectura y comprensión. Actividades:

- Configurar React.js con TypeScript
- Instalar Tailwind CSS, Chart.js, React Router
- Implementar fuentes OpenDyslexic
- Crear componentes base con diseño accesible

Base de Datos Funcional

Responsable: Persona 3 (Full-Stack) Descripción: Construir el almacén de datos que registrará el progreso educativo de cada estudiante, sus patrones de aprendizaje y el contenido adaptativo del sistema. Actividades:

- Configurar PostgreSQL con esquemas básicos
- Implementar modelos de datos con SQLAlchemy
- Crear scripts de migración inicial
- Crear datos de prueba básicos

Documentación y Datos de Prueba

Responsable: Persona 4 (Informática) Descripción: Recopilar y estructurar el contenido educativo que alimentará el sistema, asegurando que el tutor tenga material de calidad para trabajar con estudiantes de diferentes niveles. Actividades:

- Configurar repositorio Git
 - Recopilar ejercicios educativos por materia
 - Crear dataset de conceptos educativos
 - Documentar requerimientos técnicos iniciales
-

DÍA 2: BACKEND CORE

Descripción: Construir la columna vertebral del sistema que conectará la inteligencia artificial con la interfaz de usuario, creando una API robusta que permita la comunicación fluida entre todos los componentes del tutor educativo.

API REST Completa

Responsable: Persona 1 (Backend + ML) Descripción: Desarrollar los canales de comunicación que permitirán al frontend acceder a los algoritmos de ML y obtener recomendaciones personalizadas en tiempo real. Actividades:

- Implementar endpoints para Knowledge Tracing
- Crear endpoints de estudiantes y ejercicios
- Integrar modelos ML con API
- Implementar sistema de logging y validación

Dashboard Base Funcional

Responsable: Persona 2 (Frontend) Descripción: Crear la interfaz principal donde los estudiantes interactuarán con el tutor inteligente, mostrando su progreso y recibiendo ejercicios adaptados a sus necesidades. Actividades:

- Crear dashboard principal con routing
- Implementar componentes de formularios
- Conectar frontend con API básica
- Crear componentes de visualización básica

Servicios de Base de Datos

Responsable: Persona 3 (Full-Stack)Descripción: Implementar la capa de servicios que gestionará eficientemente el almacenamiento y recuperación de datos educativos, optimizando el rendimiento del sistema.Actividades:

- Implementar ORM completo con SQLAlchemy
- Crear servicios de base de datos optimizados
- Configurar CORS y middleware
- Optimizar consultas de base de datos

Testing y Validación

Responsable: Persona 4 (Informática)Descripción: Verificar que todos los componentes del sistema funcionen correctamente juntos, asegurando la calidad y confiabilidad del tutor educativo.Actividades:

- Crear tests unitarios básicos
- Documentar API endpoints
- Validar funcionalidad básica del sistema
- Crear guía de instalación

DÍA 3: SISTEMA DE RECOMENDACIÓN

Descripción: Implementar la "inteligencia" del tutor que analizará el perfil único de cada estudiante para recomendar ejercicios en el punto óptimo de dificultad, maximizando el aprendizaje y la retención.

Algoritmos de Recomendación

Responsable: Persona 1 (Backend + ML)Descripción: Desarrollar el cerebro del sistema que analiza patrones de aprendizaje y determina qué ejercicio será más efectivo para cada estudiante en cada momento.Actividades:

- Implementar motor de recomendación inteligente
- Crear adaptación de dificultad dinámica
- Desarrollar filtrado por materia
- Optimizar algoritmos ML para recomendaciones

Componentes de Ejercicios

Responsable: Persona 2 (Frontend)Descripción: Crear las herramientas interactivas que permitirán a los estudiantes resolver ejercicios de manera intuitiva, recibiendo feedback inmediato y visual sobre su progreso.Actividades:

- Crear componentes interactivos de ejercicios
- Implementar sistema de respuestas y feedback
- Desarrollar visualizaciones de progreso
- Crear componentes de recomendaciones

Integración Frontend-Backend

Responsable: Persona 3 (Full-Stack) Descripción: Conectar la interfaz de usuario con los algoritmos de recomendación, asegurando que las sugerencias del tutor lleguen al estudiante de manera fluida y efectiva. Actividades:

- Integrar sistema de recomendación con frontend
- Implementar lógica de negocio completa
- Optimizar consultas y rendimiento
- Crear sistema de eventos

Ejercicios y Testing

Responsable: Persona 4 (Informática) Descripción: Crear y validar el contenido educativo que el sistema utilizará, asegurando que los ejercicios sean apropiados y efectivos para el aprendizaje adaptativo. Actividades:

- Crear ejercicios educativos completos por materia
- Implementar tests de integración
- Validar algoritmos de recomendación
- Crear guías de usuario básicas

DÍA 4: FRONTEND ADAPTATIVO

Descripción: Transformar la interfaz en una experiencia verdaderamente adaptativa que se ajuste a las necesidades específicas de estudiantes con TDA y dislexia, creando un entorno de aprendizaje óptimo.

Optimización de Backend

Responsable: Persona 1 (Backend + ML) Descripción: Mejorar el rendimiento de los algoritmos para que respondan rápidamente a las necesidades del frontend adaptativo, proporcionando análisis en tiempo real. Actividades:

- Optimizar endpoints para frontend
- Implementar análisis de patrones básico

- Crear sistema de reportes y estadísticas
- Implementar caché de recomendaciones

Dashboard Adaptativo Completo

Responsable: Persona 2 (Frontend) Descripción: Implementar una interfaz que se adapte dinámicamente a las necesidades cognitivas de cada estudiante, con elementos visuales y de navegación optimizados. Actividades:

- Implementar dashboard completamente adaptativo
- Crear sistema de temas accesibles
- Desarrollar visualizaciones avanzadas
- Optimizar accesibilidad y navegación

Integración Completa

Responsable: Persona 3 (Full-Stack) Descripción: Asegurar que todos los componentes del sistema trabajen en perfecta sincronía, proporcionando una experiencia fluida y sin interrupciones para el estudiante. Actividades:

- Integrar completamente frontend con backend
- Implementar manejo de errores robusto
- Crear sistema de notificaciones y feedback
- Optimizar rendimiento general

Testing de Usabilidad

Responsable: Persona 4 (Informática) Descripción: Validar que la interfaz sea verdaderamente accesible y efectiva para estudiantes con necesidades especiales, asegurando que cumpla con estándares de usabilidad. Actividades:

- Realizar testing de usabilidad completo
- Validar características de accesibilidad
- Crear documentación de usuario
- Preparar presentación del proyecto

DÍA 5: INTEGRACIÓN Y OPTIMIZACIÓN

Descripción: Pulir y optimizar todos los componentes del sistema para crear una experiencia de aprendizaje fluida, rápida y efectiva que maximice el potencial educativo del tutor inteligente.

Optimización ML Final

Responsable: Persona 1 (Backend + ML) Descripción: Refinar los algoritmos de Machine Learning para que proporcionen recomendaciones más precisas y análisis más profundos del progreso educativo. Actividades:

- Optimizar algoritmos ML para rendimiento final
- Implementar análisis de atención básico
- Crear reportes avanzados del sistema
- Documentar código ML

Pulido de UI/UX

Responsable: Persona 2 (Frontend) Descripción: Mejorar la experiencia visual y de interacción, añadiendo elementos que hagan el aprendizaje más atractivo y efectivo para los estudiantes. Actividades:

- Pulir interfaz de usuario completamente
- Implementar animaciones y mejoras visuales
- Optimizar responsive design y accesibilidad
- Crear componentes extras

Testing y Deployment

Responsable: Persona 3 (Full-Stack) Descripción: Verificar que el sistema esté listo para funcionar en un entorno de producción, asegurando estabilidad y rendimiento óptimo. Actividades:

- Realizar testing de integración completa
- Crear scripts de deployment
- Configurar ambiente de producción
- Documentar arquitectura

Demo y Documentación

Responsable: Persona 4 (Informática) Descripción: Crear una demostración efectiva del sistema y documentar su funcionamiento para facilitar su uso y comprensión por parte de educadores y estudiantes. Actividades:

- Crear demo funcional del proyecto
- Preparar documentación final
- Crear video de presentación

- Organizar entrega del proyecto
-

DÍA 6: TESTING Y ENTREGA FINAL

Descripción: Realizar una evaluación exhaustiva del sistema completo y preparar la entrega final, asegurando que el tutor inteligente esté listo para mejorar la educación de estudiantes con necesidades especiales.

Testing ML Exhaustivo

Responsable: Persona 1 (Backend + ML) Descripción: Validar que los algoritmos de inteligencia artificial funcionen correctamente en todos los escenarios, asegurando la precisión de las recomendaciones educativas. Actividades:

- Testing exhaustivo de algoritmos ML
- Validación completa de recomendaciones
- Optimización final de rendimiento
- Documentación técnica final

Testing UI/UX Final

Responsable: Persona 2 (Frontend) Descripción: Verificar que la interfaz sea completamente accesible y funcional en diferentes dispositivos y navegadores, optimizando la experiencia del usuario final. Actividades:

- Testing de usabilidad y accesibilidad final
- Optimización final de interfaz
- Testing cross-browser
- Preparación de demo de interfaz

Testing de Integración Final

Responsable: Persona 3 (Full-Stack) Descripción: Asegurar que todos los componentes del sistema trabajen perfectamente juntos, realizando pruebas de integración completas y configurando el entorno de producción. Actividades:

- Testing de integración completa del sistema
- Configuración final de producción
- Backup completo del sistema
- Documentación de deployment

Entrega Final Completa

Responsable: Persona 4 (Informática) Descripción: Preparar la presentación final del proyecto, organizando toda la documentación y creando materiales que demuestren el valor educativo del tutor inteligente. Actividades:

- Testing final completo del sistema
- Crear documentación de usuario final
- Preparar presentación del proyecto
- Organizar entrega final completa

PLAN DE TRABAJO DETALLADO - TUTORIA MVP

Equipo: 4 personas | Duración: 6 días

DÍA 1: SETUP + KNOWLEDGE TRACING BÁSICO

Objetivo: Configurar entorno de desarrollo e implementar sistema básico de seguimiento de conocimiento

PERSONA 1 (Backend + ML)

Tareas:

- ☐ Configurar entorno Python con virtual environment
- ☐ Instalar dependencias: FastAPI, scikit-learn, pandas, numpy, sqlalchemy
- ☐ Crear estructura de carpetas del proyecto
- ☐ Implementar modelo básico de Knowledge Tracing
- ☐ Crear función de tracking de respuestas correctas/incorrectas
- ☐ Implementar cálculo de nivel de dominio por concepto
- ☐ Desarrollar predicción simple de dificultad

Entregables:

- Entorno Python configurado
- Modelo de Knowledge Tracing funcional
- Funciones de tracking de respuestas
- Algoritmo de cálculo de dominio
- Predicción de dificultad básica

PERSONA 2 (Frontend)

Tareas:

- ☐ Configurar React.js con TypeScript
- ☐ Instalar dependencias: Tailwind CSS, Chart.js, React Router
- ☐ Crear estructura de componentes base
- ☐ Configurar fuentes OpenDyslexic
- ☐ Definir paleta de colores accesible
- ☐ Crear componentes base de UI

Entregables:

- React con TypeScript configurado
- Dependencias instaladas
- Componentes base creados
- Fuentes OpenDyslexic configuradas
- Paleta de colores definida

PERSONA 3 (Full-Stack)**Tareas:**

- ☐ Configurar base de datos PostgreSQL
- ☐ Crear esquemas de base de datos básicos
- ☐ Implementar modelos de datos con SQLAlchemy
- ☐ Crear scripts de migración inicial
- ☐ Configurar conexión a base de datos
- ☐ Crear datos de prueba básicos

Entregables:

- Base de datos PostgreSQL configurada
- Esquemas de datos creados
- Modelos SQLAlchemy implementados
- Scripts de migración
- Datos de prueba básicos

PERSONA 4 (Informática)**Tareas:**

- ☐ Configurar repositorio Git
- ☐ Crear README del proyecto
- ☐ Recopilar datos de ejercicios educativos
- ☐ Crear dataset de conceptos por materia
- ☐ Documentar requerimientos técnicos
- ☐ Crear casos de prueba básicos

Entregables:

- Repositorio Git configurado
 - README del proyecto
 - Dataset de ejercicios educativos
 - Documentación de requerimientos
 - Casos de prueba básicos
-

DÍA 2: BACKEND CORE

Objetivo: Implementar API REST completa y sistema de base de datos funcional

PERSONA 1 (Backend + ML)**Tareas:**

- ☐ Implementar API REST con FastAPI
- ☐ Crear endpoints para Knowledge Tracing
- ☐ Integrar modelos ML con API
- ☐ Implementar sistema de logging
- ☐ Crear middleware de autenticación básica
- ☐ Implementar validación de datos

Entregables:

- API REST funcional con FastAPI
- Endpoints de Knowledge Tracing
- Integración de modelos ML
- Sistema de logging implementado
- Middleware de autenticación básica

PERSONA 2 (Frontend)**Tareas:**

- ☐ Crear componentes de dashboard base
- ☐ Implementar routing con React Router
- ☐ Crear componentes de formularios

- ☐ Implementar estado global (Context API)
- ☐ Crear componentes de visualización básica
- ☐ Implementar llamadas a API

Entregables:

- Dashboard base funcional
- Sistema de routing implementado
- Componentes de formularios
- Estado global configurado
- Componentes de visualización básica

PERSONA 3 (Full-Stack)

Tareas:

- ☐ Implementar ORM completo con SQLAlchemy
- ☐ Crear servicios de base de datos
- ☐ Implementar validación de datos robusta
- ☐ Crear sistema de caché básico
- ☐ Configurar CORS y middleware
- ☐ Optimizar consultas de base de datos

Entregables:

- ORM completo configurado
- Servicios de base de datos
- Validación de datos robusta
- Sistema de caché básico
- CORS y middleware configurados

PERSONA 4

Tareas:

- ☐ Crear datos de prueba completos
- ☐ Implementar tests unitarios básicos
- ☐ Documentar API endpoints

- ☐ Crear guía de instalación
- ☐ Configurar ambiente de testing
- ☐ Validar funcionalidad básica

Entregables:

- Datos de prueba completos
 - Tests unitarios básicos
 - Documentación de API
 - Guía de instalación
 - Ambiente de testing configurado
-

DÍA 3: SISTEMA DE RECOMENDACIÓN

Objetivo: Implementar sistema inteligente de recomendación de ejercicios

PERSONA 1 (Backend + ML)

Tareas:

- ☐ Implementar algoritmo de recomendación
- ☐ Crear sistema de filtrado por materia
- ☐ Desarrollar adaptación de dificultad dinámica
- ☐ Integrar recomendaciones con API
- ☐ Optimizar algoritmos ML
- ☐ Implementar análisis de patrones básico

Entregables:

- Sistema de recomendación funcional
- Filtrado por materia implementado
- Adaptación de dificultad dinámica
- API de recomendaciones integrada
- Algoritmos ML optimizados

PERSONA 2 (Frontend)

Tareas:

- ☐ Crear componentes de ejercicios
- ☐ Implementar sistema de respuestas
- ☐ Crear visualizaciones de progreso
- ☐ Implementar feedback visual
- ☐ Crear componentes de recomendaciones
- ☐ Implementar sistema de navegación

Entregables:

- Componentes de ejercicios
- Sistema de respuestas funcional
- Visualizaciones de progreso
- Feedback visual implementado
- Componentes de recomendaciones

PERSONA 3 (Full-Stack)

Tareas:

- ☐ Implementar lógica de negocio completa
- ☐ Crear servicios de recomendación
- ☐ Optimizar consultas de base de datos
- ☐ Implementar sistema de eventos
- ☐ Crear middleware de validación
- ☐ Integrar frontend con backend

Entregables:

- Lógica de negocio implementada
- Servicios de recomendación
- Consultas optimizadas
- Sistema de eventos
- Integración frontend-backend

PERSONA 4 (Informática)

Tareas:

- ☐ Crear ejercicios educativos completos
- ☐ Implementar tests de integración
- ☐ Validar algoritmos de recomendación
- ☐ Documentar funcionalidades
- ☐ Crear guías de usuario
- ☐ Testing de usabilidad básico

Entregables:

- Ejercicios educativos completos
 - Tests de integración
 - Validación de algoritmos
 - Documentación de funcionalidades
 - Guías de usuario
-

DÍA 4: FRONTEND ADAPTATIVO

Objetivo: Implementar interfaz adaptativa optimizada para dislexia y TDA

PERSONA 1 (Backend + ML)

Tareas:

- ☐ Optimizar endpoints para frontend
- ☐ Implementar análisis de patrones básico
- ☐ Crear endpoints de reportes
- ☐ Optimizar rendimiento de ML
- ☐ Implementar caché de recomendaciones
- ☐ Crear endpoints de estadísticas

Entregables:

- Endpoints optimizados
- Análisis de patrones básico
- Sistema de reportes
- Rendimiento ML optimizado

- Caché de recomendaciones

PERSONA 2 (Frontend)

Tareas:

- ☐ Implementar dashboard completo
- ☐ Crear componentes adaptativos
- ☐ Implementar sistema de temas
- ☐ Crear visualizaciones avanzadas
- ☐ Optimizar accesibilidad
- ☐ Implementar fuentes OpenDyslexic

Entregables:

- Dashboard completo
- Componentes adaptativos
- Sistema de temas
- Visualizaciones avanzadas
- Accesibilidad optimizada

PERSONA 3 (Full-Stack)

Tareas:

- ☐ Integrar frontend con backend completamente
- ☐ Implementar manejo de errores robusto
- ☐ Crear sistema de notificaciones
- ☐ Optimizar rendimiento general
- ☐ Implementar validaciones frontend
- ☐ Crear sistema de feedback

Entregables:

- Integración completa
- Manejo de errores robusto
- Sistema de notificaciones
- Rendimiento optimizado

- Validaciones frontend

PERSONA 4

Tareas:

- ☐ Testing de usabilidad completo
- ☐ Validación de accesibilidad
- ☐ Crear documentación de usuario
- ☐ Preparar presentación del proyecto
- ☐ Testing de integración completa
- ☐ Crear casos de uso

Entregables:

- Reporte de testing de usabilidad
- Validación de accesibilidad
- Documentación de usuario
- Presentación del proyecto
- Testing de integración completo

DÍA 5: INTEGRACIÓN Y OPTIMIZACIÓN

Objetivo: Integrar todos los componentes y optimizar el sistema

PERSONA 1 (Backend + ML)

Tareas:

- ☐ Optimización final de algoritmos ML
- ☐ Implementar análisis de atención básico
- ☐ Crear reportes avanzados
- ☐ Optimizar rendimiento del backend
- ☐ Documentar código ML
- ☐ Implementar monitoreo de sistema

Entregables:

- Algoritmos ML optimizados

- Análisis de atención básico
- Reportes avanzados
- Backend optimizado
- Documentación técnica ML

PERSONA 2 (Frontend)

Tareas:

- ☐ Pulir interfaz de usuario
- ☐ Implementar animaciones suaves
- ☐ Optimizar responsive design
- ☐ Crear componentes extras
- ☐ Documentar componentes
- ☐ Optimizar rendimiento frontend

Entregables:

- UI completamente pulida
- Animaciones implementadas
- Responsive design optimizado
- Componentes documentados
- Frontend optimizado

PERSONA 3 (Full-Stack)

Tareas:

- ☐ Testing de integración completa
- ☐ Optimización de rendimiento general
- ☐ Implementar features extras
- ☐ Crear scripts de deployment
- ☐ Documentar arquitectura
- ☐ Configurar ambiente de producción

Entregables:

- Testing completo

- Rendimiento optimizado
- Features extras implementadas
- Scripts de deployment
- Documentación de arquitectura

PERSONA 4

Tareas:

- ☐ Testing final completo del sistema
- ☐ Crear demo funcional del proyecto
- ☐ Preparar documentación final
- ☐ Crear video de presentación
- ☐ Organizar entrega final
- ☐ Preparar presentación técnica

Entregables:

- Testing final completo
- Demo funcional
- Documentación final
- Video de presentación
- Entrega final organizada

DÍA 6: TESTING Y ENTREGA FINAL

Objetivo: Testing exhaustivo y preparación para entrega

PERSONA 1 (Backend + ML)

Tareas:

- ☐ Testing exhaustivo de algoritmos ML
- ☐ Validación de recomendaciones
- ☐ Optimización final de rendimiento
- ☐ Documentación técnica completa
- ☐ Preparar presentación técnica

- ☐ Backup de datos y configuración

Entregables:

- Testing ML completo
- Validación de recomendaciones
- Rendimiento final optimizado
- Documentación técnica completa
- Presentación técnica

PERSONA 2 (Frontend)**Tareas:**

- ☐ Testing de usabilidad final
- ☐ Validación de accesibilidad completa
- ☐ Optimización final de UI/UX
- ☐ Documentación de componentes
- ☐ Preparar demo de interfaz
- ☐ Testing cross-browser

Entregables:

- Testing de usabilidad final
- Validación de accesibilidad
- UI/UX final optimizada
- Documentación de componentes
- Demo de interfaz

PERSONA 3 (Full-Stack)**Tareas:**

- ☐ Testing de integración final
- ☐ Optimización de rendimiento general
- ☐ Configuración de producción
- ☐ Documentación de deployment
- ☐ Preparar presentación técnica

- [] Backup del sistema completo

Entregables:

- Testing de integración final
- Sistema optimizado para producción
- Documentación de deployment
- Presentación técnica
- Backup completo

PERSONA 4

Tareas:

- Testing final completo del sistema
- Crear documentación de usuario final
- Preparar presentación del proyecto
- Crear video demo final
- Organizar entrega final
- Preparar documentación ejecutiva

Entregables:

- Testing final completo
- Documentación de usuario final
- Presentación del proyecto
- Video demo final
- Entrega final completa

ENTREGABLES FINALES DEL MVP

Funcionalidades Core (Obligatorias):

- Sistema de Knowledge Tracing funcional
- API REST completa con FastAPI
- Sistema de recomendación inteligente
- Dashboard adaptativo para dislexia y TDA

- Base de datos con datos de prueba
- Interfaz accesible con OpenDyslexic