



Tecnológico de Monterrey

Desarrollo e implantación de sistemas de software (Gpo 103)

Prof. Juan Carlos Lavariega

Documentación del Diseño

Horizons

Alan De Loa Larios

Gilberto Ángel Camacho Lara

Eugenio Elizondo Jaime

Gianni André Freire Vásquez

Aldo Leonardo López Ruiz

Carlos Fernando Ramos Mena

3 de junio de 2025. Monterrey, Nuevo León.

Introducción y objetivos

Pathfinder es una plataforma web desarrollada de la mano con Accenture para la asignación de talentos en proyectos empresariales, utilizando inteligencia artificial y una arquitectura full-stack. Su objetivo es permitir que los responsables de recursos humanos, así como líderes de proyectos y gerentes puedan encontrar el personal más adecuado para los diferentes proyectos dentro de la empresa, teniendo en cuenta sus habilidades, experiencia y formación profesional (cursos/certificaciones).

Objetivos:

- Asignación inteligente de los empleados a los diferentes proyectos con inteligencia artificial.
- Vista clara de los proyectos activos y del personal asignado.
- Interfaz clara para los empleados, managers y TFS, mostrando componentes relevantes.
- Recomendación de cursos y certificaciones en función a sus metas e historial de proyectos, ayudándolos a potenciar su crecimiento dentro de la organización.

Contexto

Actualmente, en Accenture existe un flujo operativo que involucra la asignación de empleados a proyectos, así como la capacitación de aquellos que no se encuentran activos en ningún proyecto. Cuando un empleado no está asignado, se le considera parte del "staff en banca", y durante este periodo puede tomar cursos de capacitación. Con base en esta dinámica, se calcula un porcentaje de asignación, el cual refleja el tiempo efectivo que un empleado ha estado trabajando en proyectos; un mayor porcentaje indica una mejor utilización del recurso humano.

El proceso de asignación de personal es responsabilidad del área de Talent Fulfillment Specialist (TFS), la cual enfrenta varios retos. Para asignar al mejor candidato a un rol dentro de un proyecto, el TFS debe evaluar múltiples criterios y consultar las habilidades y experiencia del staff disponible. Sin embargo, esta tarea se ve entorpecida debido a la distribución de los sistemas involucrados.

Por un lado, los cursos que toman los empleados en banca están registrados en un sistema, mientras que los proyectos y sus respectivos roles se gestionan en otro sistema independiente. Esta fragmentación obliga al TFS a consultar diferentes plataformas para obtener información completa, lo cual incrementa significativamente el tiempo necesario para tomar decisiones de asignación.

Adicionalmente, los managers encargados de proyectos crean nuevas iniciativas y definen los roles requeridos para cada uno. Posteriormente, el TFS se encarga de asignar al personal adecuado. Esta separación entre los sistemas y los procesos administrativos genera dispersión de información, lo que a su vez complica la toma de decisiones estratégicas a nivel organizacional.

Para cada usuario (empleado) se gestionan datos relevantes como habilidades, certificaciones, historial de proyectos y porcentaje de asignación. Esta información no solo se utiliza como criterio para asignaciones más precisas a roles dentro de proyectos, sino que también cumple una función clave en la gestión del desarrollo profesional dentro de la empresa. Cada empleado cuenta con metas profesionales individuales, y con base en ellas puede tomar decisiones informadas sobre qué cursos realizar, ya sea durante su periodo en banca o seleccionándolos del catálogo general, con el objetivo de fortalecer su perfil y avanzar en su crecimiento dentro de la organización.

Restricciones

Una de las restricciones más relevantes en el desarrollo de Pathfinder fue la limitación de presupuesto para las tecnologías, debido a ser un proyecto estudiantil. Esta condición nos restringe utilizar tecnologías de paga o servicios de calidad. Se eligieron tecnologías open source como Supabase para la base de datos, así como servicios accesibles de IA como DeepSeek.

Alcance

El alcance de este proyecto abarca tres áreas clave del proceso de gestión de talento y proyectos en la organización:

1. Optimización del proceso de asignación de proyectos:
Se busca agilizar y mejorar la asignación del personal a roles dentro de proyectos mediante el uso de inteligencia artificial. Esta tecnología permitirá analizar habilidades, certificaciones, historial de proyectos y porcentaje de asignación para sugerir de forma eficiente al candidato más adecuado, reduciendo significativamente el tiempo de respuesta del área de Talent Fulfillment Specialist (TFS).
2. Recomendación de cursos alineados a metas profesionales:
Para los empleados en banca o en proceso de desarrollo, se implementará un sistema de recomendación inteligente que, basado en las metas profesionales definidas por cada empleado y en un catálogo de cursos disponible, sugiera las capacitaciones más relevantes. Esto permitirá un enfoque más estratégico y personalizado en el crecimiento profesional del

staff.

3. Gestión estructurada de proyectos y control de accesos:

Se estandariza la creación de proyectos por parte de los managers y se establece un sistema con tres niveles de cuenta, cada uno con permisos específicos según el rol. Además, se asegura un manejo estructurado de la información de cada usuario, incluyendo habilidades, certificaciones, historial de proyectos y metas profesionales, lo cual fortalece la toma de decisiones y el seguimiento del desarrollo individual.

Estrategia de solución

Para la implementación de Pathfinder se utilizó una arquitectura cliente-servidor, con una clara separación entre el frontend desarrollado en **React** y el backend implementado en **Node.js**. La persistencia de datos se gestionó mediante **Supabase**, una plataforma de base de datos relacional basada en PostgreSQL.

La solución se descompuso en diferentes módulos funcionales incluyendo la gestión de proyectos, asignación de empleados, así como recomendaciones de asignaciones y cursos con inteligencia artificial, implementando la API de Deep Seek.

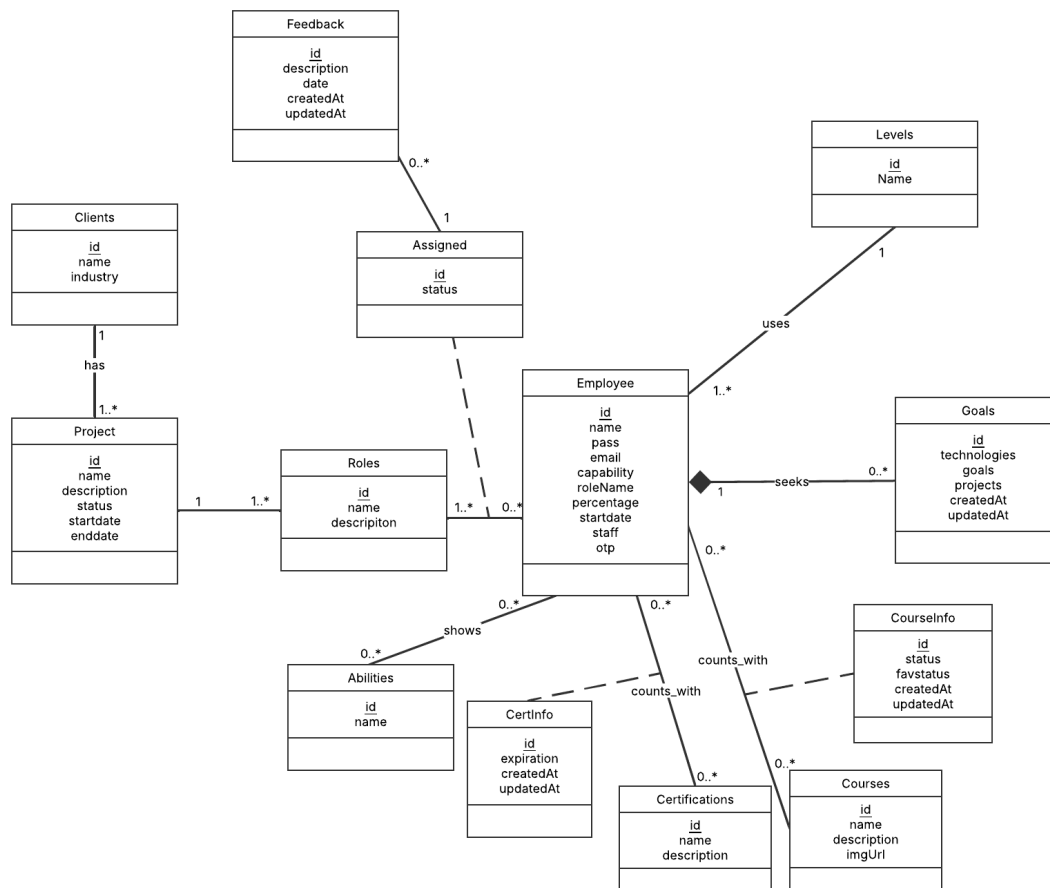
A nivel organizacional, se aplicó el manifiesto AGILE, contando con entregas iterativas y apoyo constante del cliente (Accenture). La carga de trabajo la dividimos en 4 sprints de dos semanas, en los cuales se presentaron avances constantes.

Modelo de vistas de arquitectura 4+1

Vista lógica

Pathfinder cuenta con una arquitectura en capas. En la capa de presentación se muestran los componentes React que se comunican vía API con los controladores de Node.js, el cual contiene la lógica del negocio e interactúa con Supabase y la API de DeepSeek para funciones de inteligencia artificial.

Diagrama de clases:



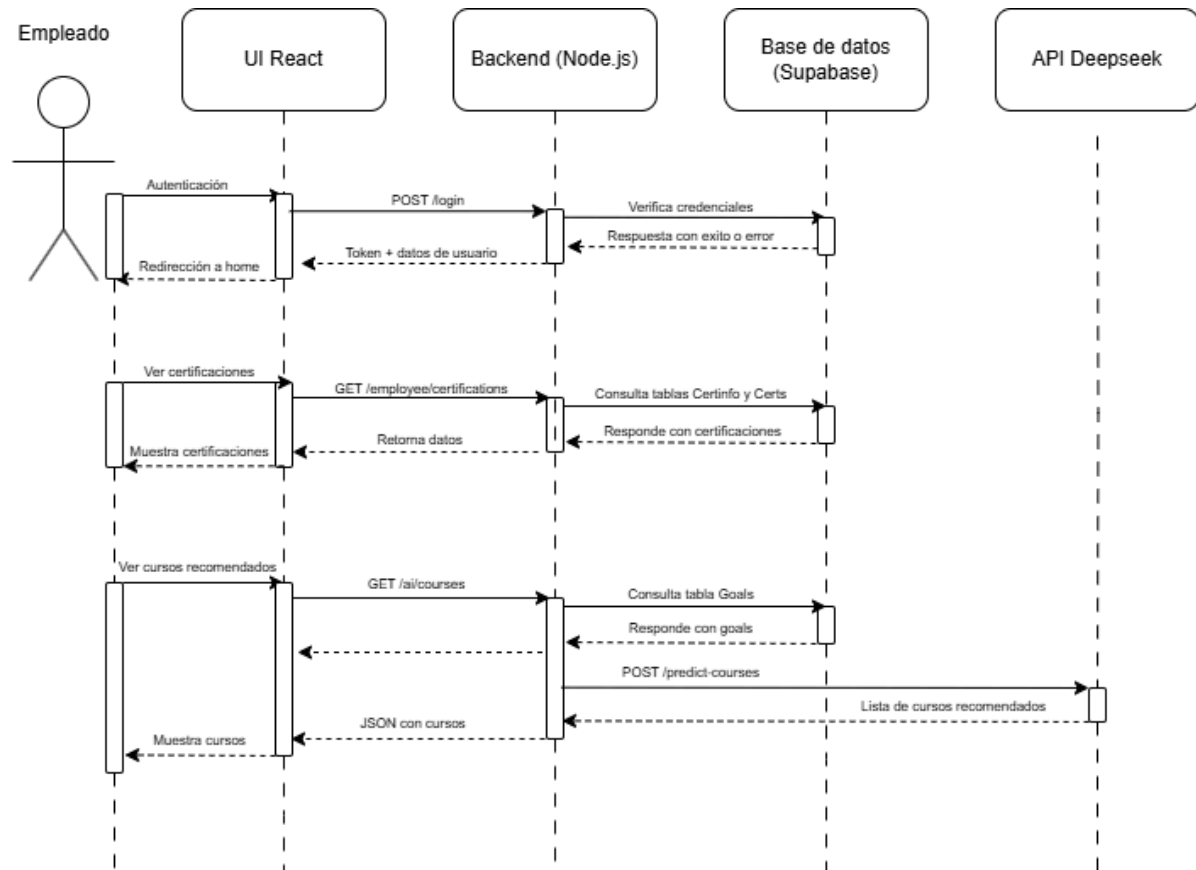
Vista de procesos

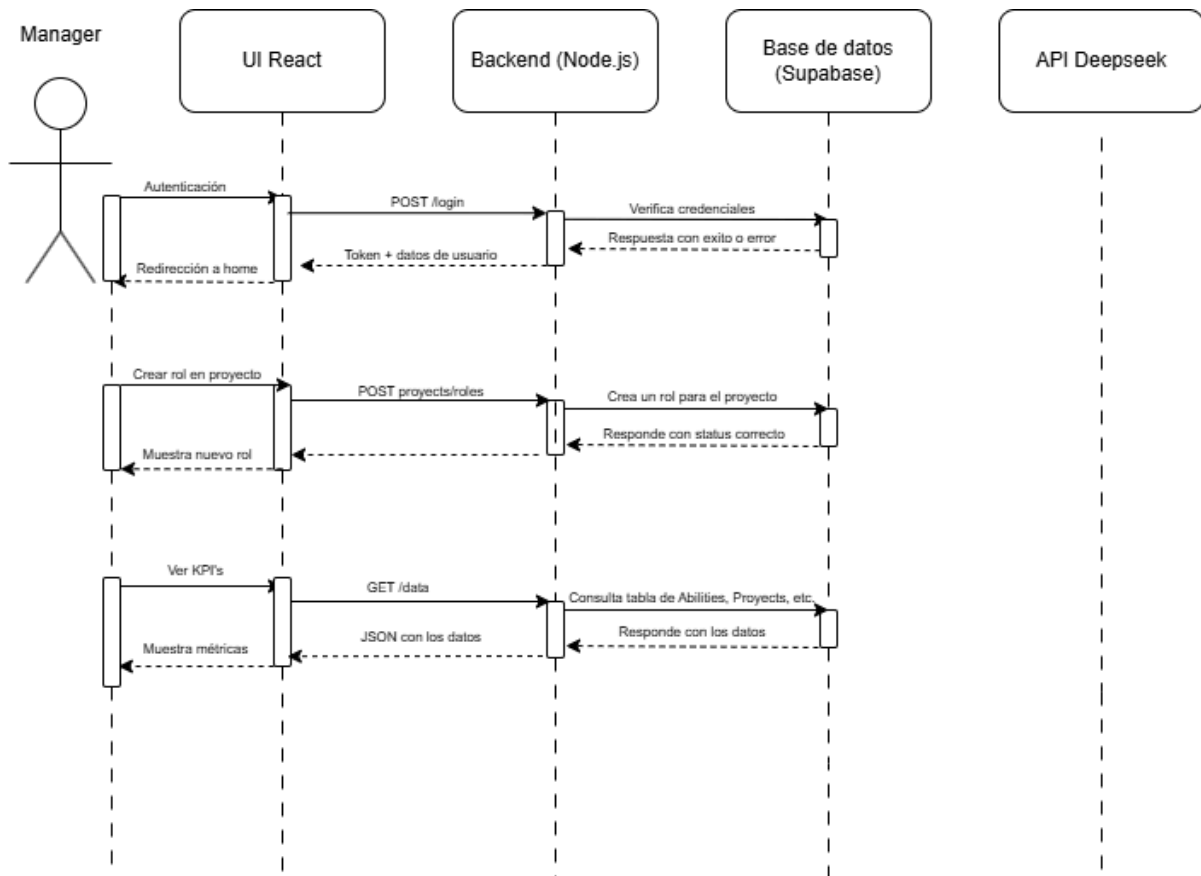
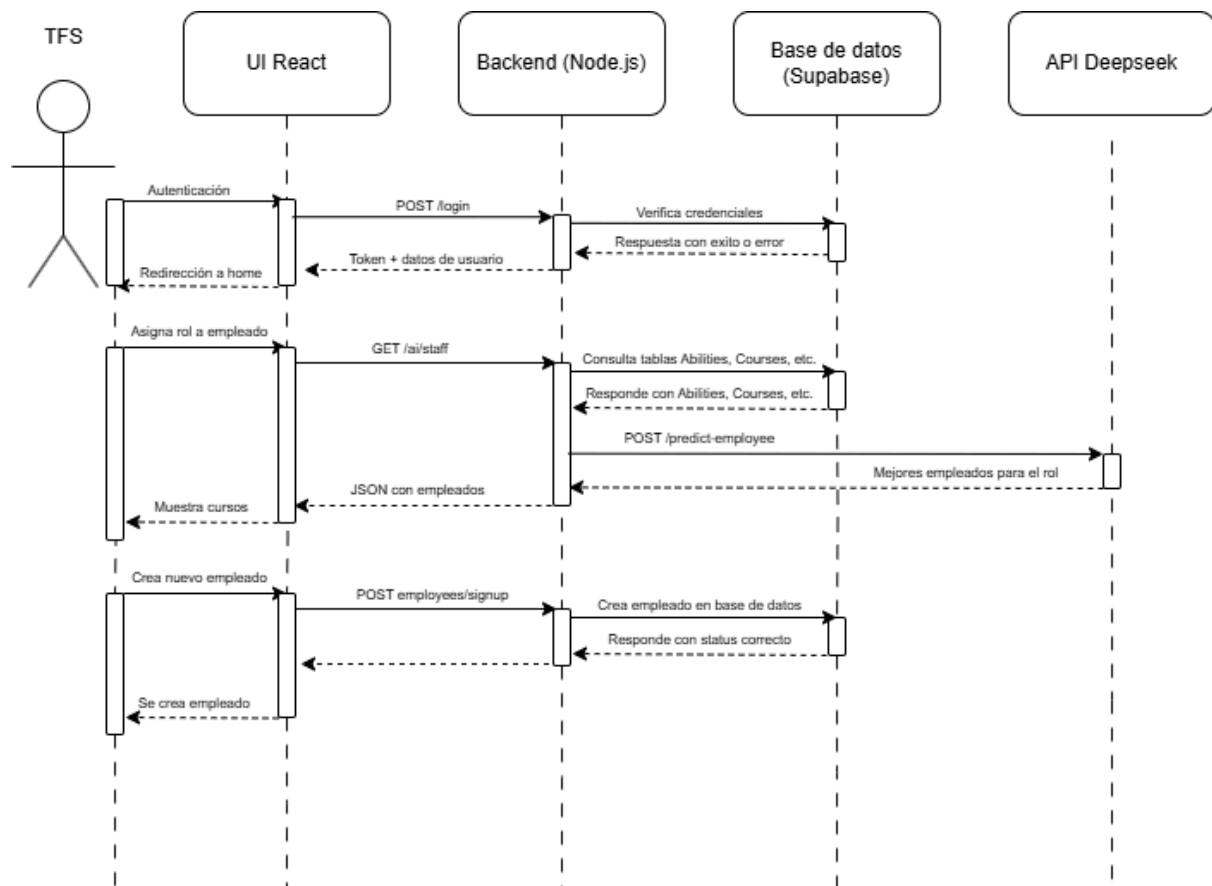
El flujo de el sistema empieza por el frontend en React el cual realiza llamadas al backend en Node.js, el cual a su vez se comunica con Supabase para recuperar los datos y con la API del modelo de inteligencia artificial de DeepSeek para realizar asignaciones inteligentes. Los procesos se comunican por medio de solicitudes HTTP.

Diagramas de secuencias

Link:

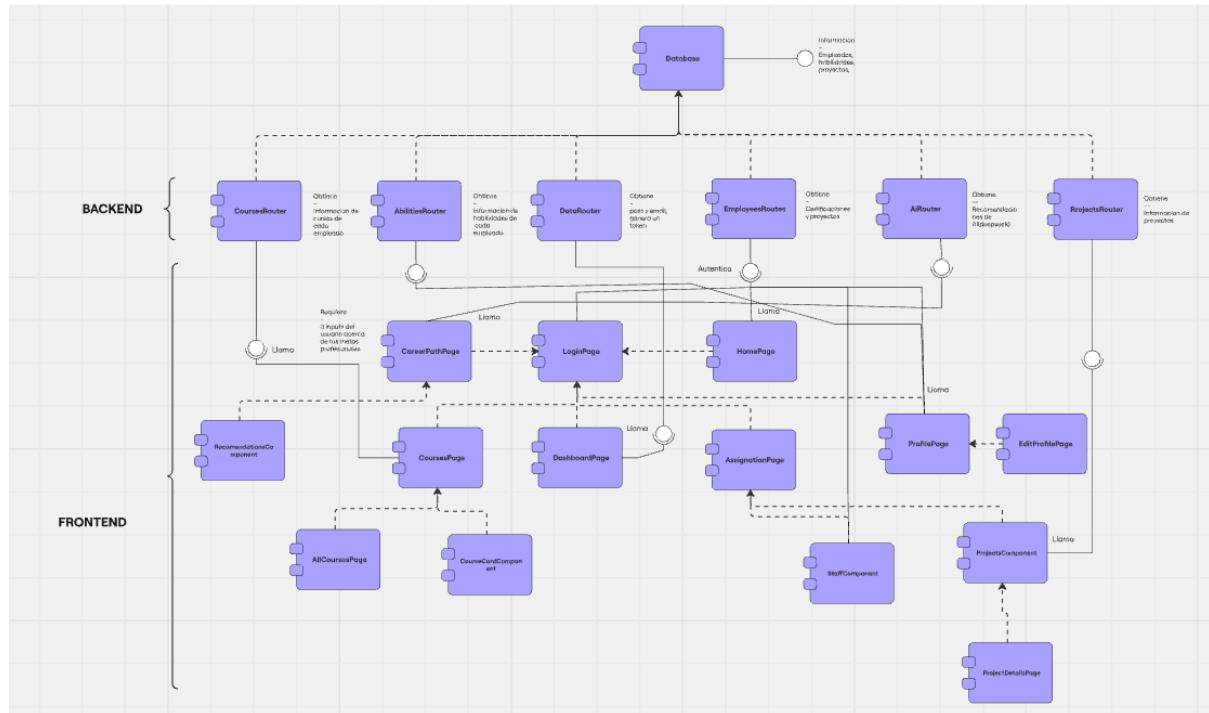
https://drive.google.com/file/d/1NZUmc1214c1tHGpxPxxkzqh2JsxvhLeMy/view?usp=drive_link





Vista de implementación(diagrama de componentes)

https://miro.com/welcomeonboard/RkQyQldGRW53cVpib1IWT0daOGFwSGZrNGxPN2dzYWdEQIRtVHJGQ2IOdGNkRIBUbGM5VWwvVXJjZ2JpbGtrMVdRMkMtY09OaE9IT2I2RUhnOUZBUm5sWTRWSDJZcmVMRTIFczdpUHRUc1JxbmpmVmRsT2ZSQWd6VnpBNzIjMGhQdGo1ZEY3bUdPQWRZUHQzSGI6V2NBPT0hdjE=?share_link_id=771152133750



Vista de despliegue

El sistema se despliega en una arquitectura distribuida con los siguientes componentes alojado en diferentes servicios:

Componente	Ubicación	Tecnología	Función
Frontend	Vercel (Cloud)	React	Interfaz de usuario responsiva.
Backend	Render (Cloud)	Node.js	API REST, lógica del negocio.
Base de datos	Supabase (Cloud)	PostgreSQL	Almacenamiento de los datos.
IA externa	DeepSeek API	Vía externa HTTP	Asignaciones y cursos inteligentes.

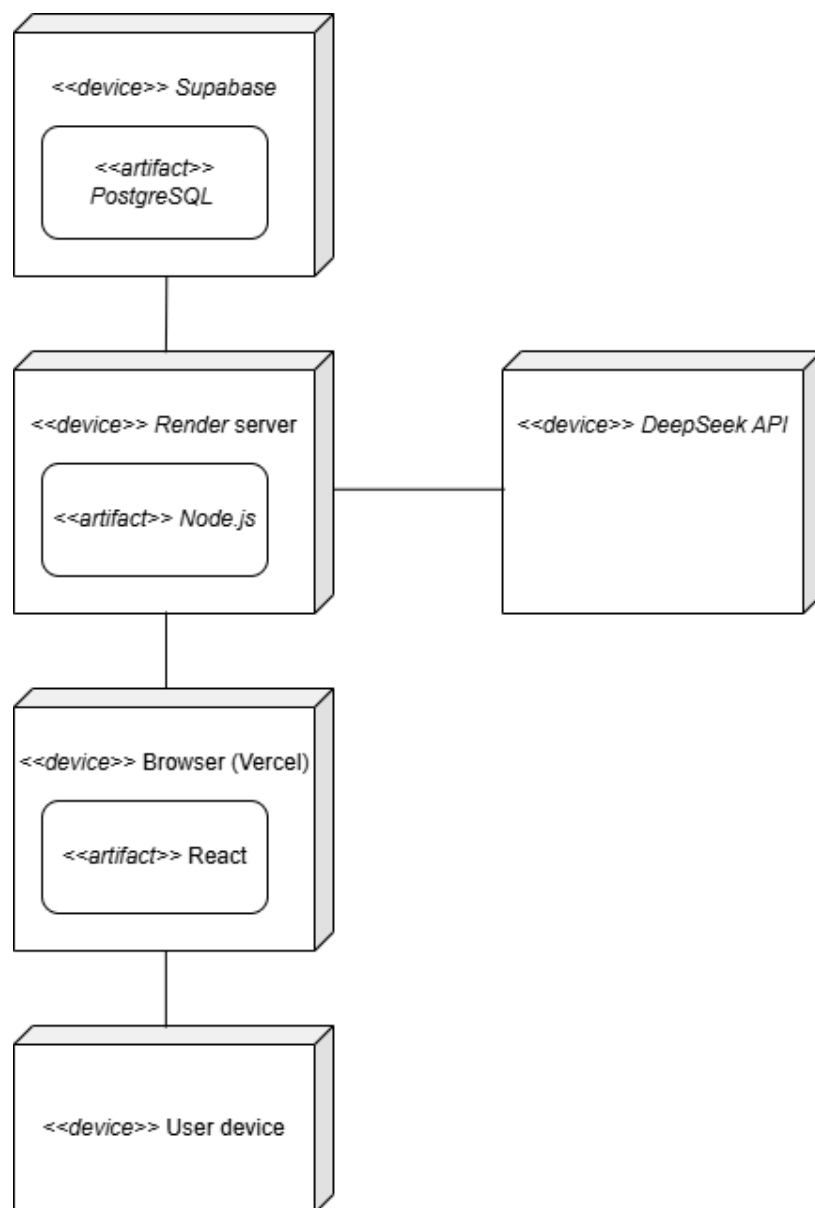
Se utilizó esta estructura de despliegue debido a:

- Esta es una solución accesible para un equipo estudiantil, con la mínima infraestructura.
- Supabase y render escalan fácilmente.

Características de calidad y rendimiento:

- Alta disponibilidad de los datos gracias a los servicios cloud.
- Baja latencia entre los componentes.
- El sistema está dividido en módulos claros, lo que facilita la actualización o modificación de componentes.

Diagrama de despliegue



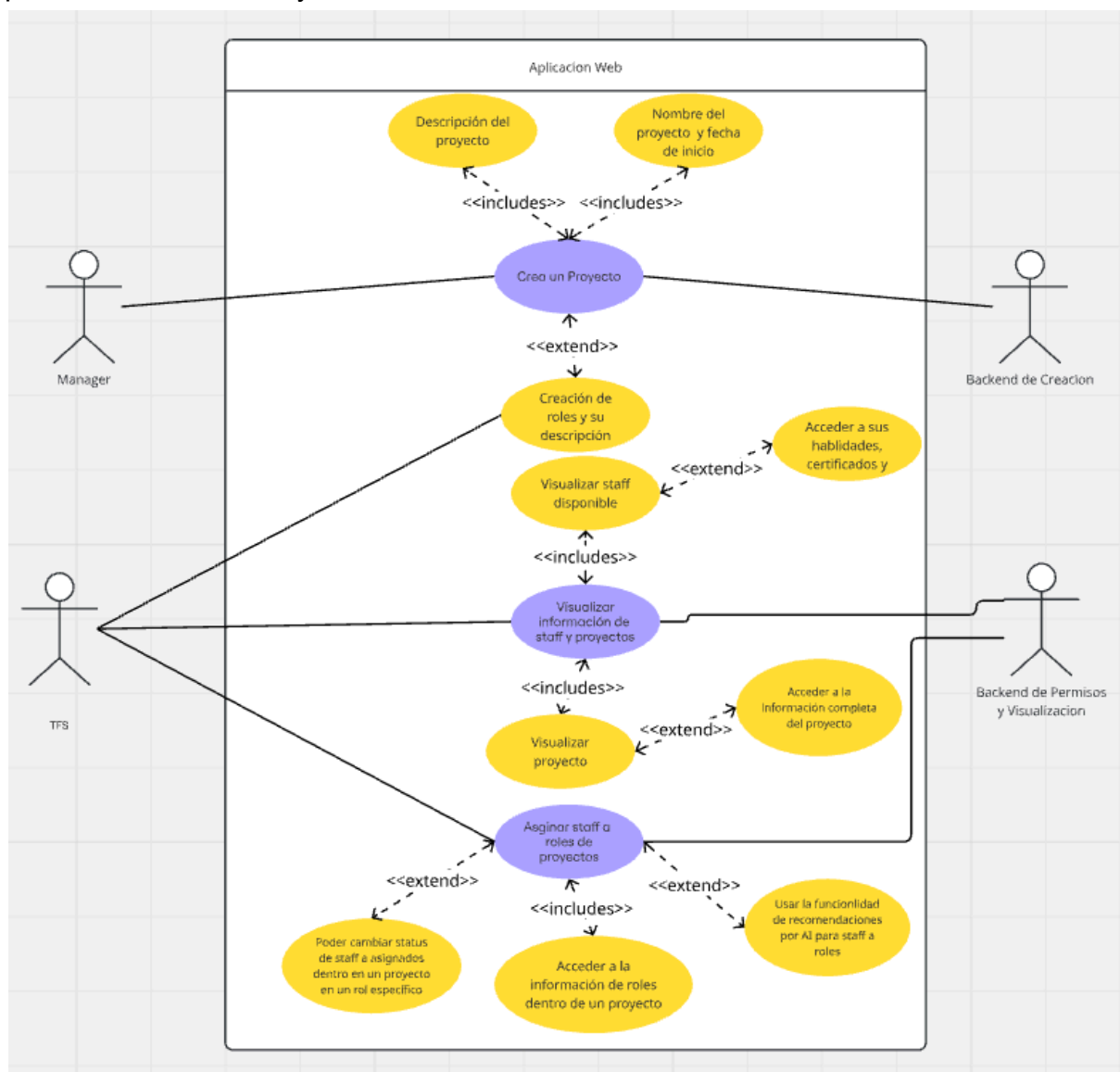
Caso de Uso: Creación de Proyecto y Asignación de Personal

Actores Principales: TFS y manager

Precondiciones:

- El manager debe tener permisos para crear proyectos con descripción, roles y una fecha de inicio.
- El TFS debe de tener permisos para poder ver la información completa de los proyectos y los roles para poder asignar al staff al rol además de poder usar una funcionalidad de IA que lo ayudara en este proceso.

Objetivo: Permitir que un manager cree un proyecto, con una descripción, roles y fecha de inicio para que así el TFS asigne staff a ese proyecto mediante la plataforma web, con ayuda de AI.



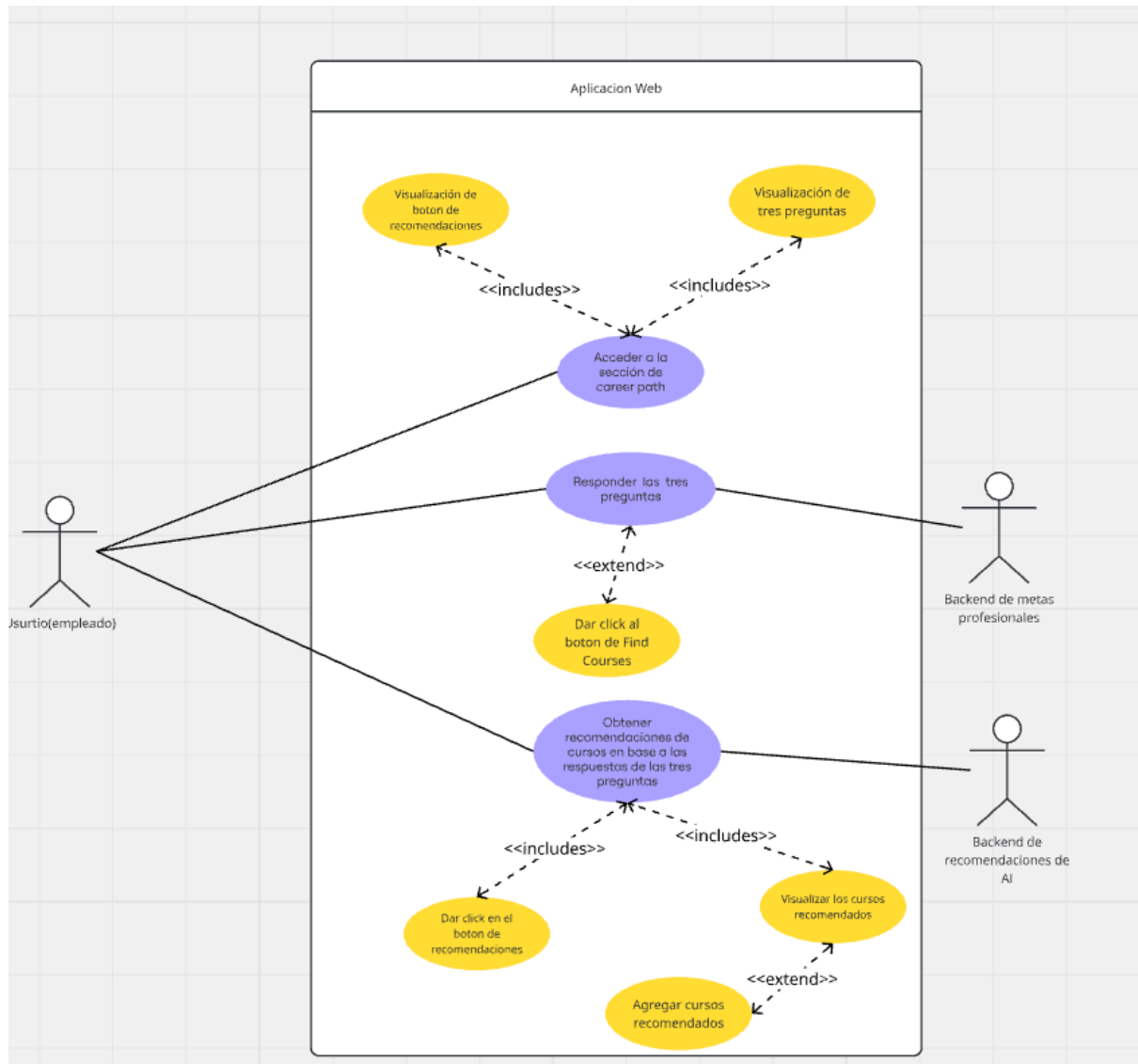
Caso de Uso: Acceso a la Sección de "Career Path" y Recomendación de Cursos

Actor Principal: Usuario de la plataforma (Empleado)

Precondiciones:

- El usuario debe estar autenticado en la plataforma.
- El usuario debe tener un perfil con información relevante, como historial profesional, habilidades, cursos tomados, y certificaciones.

Objetivo: Permitir que el usuario acceda a la sección de "Career Path", responda tres preguntas clave relacionadas con su carrera y hábitos, en base a sus respuestas y KPIs (como su historial profesional, cursos tomados, cursos actuales, habilidades y certificaciones), recibe recomendaciones personalizadas de cursos del catálogo disponible en el sistema.



El “rationale” de la arquitectura por qué tu solución es la más apropiada y no otra. Qué te llevó a tomar la decisión de construir tu aplicación de esta forma y usando la tecnología que seleccionaste.

Nuestra arquitectura sigue el modelo cliente-servidor, dividiendo la aplicación en backend y frontend para garantizar escalabilidad, claridad y mantenibilidad. El frontend está estructurado en páginas, cada una enfocada en una funcionalidad específica, y estas páginas a su vez se descomponen en componentes parametrizados y reutilizables, lo que permite un desarrollo modular y eficiente.

El backend se encarga de la lógica de negocio y la gestión de datos, exponiendo endpoints bien definidos que el frontend consume. Esta separación clara de responsabilidades permite escalar funcionalidades de forma independiente, mejorar la legibilidad del código y facilitar el mantenimiento a largo plazo.

Esta solución resulta más adecuada que una arquitectura monolítica o acoplada, ya que permite una mayor reutilización de componentes, flujo de trabajo separado entre frontend y backend, y actualizaciones más sencillas en partes específicas del sistema sin afectar al conjunto. Además, se alinea con las buenas prácticas actuales en el desarrollo de aplicaciones web y facilita el crecimiento futuro de la plataforma.