**Plataforma Reserva de Servicios en Peluquerías**

Hadechini La Barbería es una empresa que ofrece diversos servicios. Recientemente los clientes de Hadechini La Barberíase han estado quejando por las largas esperas desde que reciben su turno y son atendidos. Ya que Hadechini La Barberíano quiere manejar las reservas de forma manual, lo contactan a usted y su equipo con el fin de crear una plataforma para citar servicios ofrecidos.

Los tipos de usuarios que debe manejar el sistema son administrador, usuario externo y usuario interno.

Para acceder a la agenda de cada uno de los servicios el usuario externodeberá registrarse en la plataforma y deberá estar en la capacidad de:

* Visualizar servicios, y la agenda de cada uno de ellos.
* Visualizar agenda dado un servicio.
* Programar una cita para un servicio, previa comprobación de disponibilidad.
* Cancelar una cita previamente programada (Con mínimo 6 horas de antelación).
* Calificar la atención una vez cumplida la cita.
* Realizar y gestionar sus comentarios sobre la atención recibida.

Un usuario interno debe estar en la capacidad de:

* Crear los empleados que atienden.
* Configurar cada uno de los servicios ofrecidos (descripción, costo y duración).
* Asignar los servicios a los empleados. Un servicio puede ser ofrecido por varias personas, pero una persona solo puede estar asociada a un servicio.
* Visualizar agenda dado un empleado/servicio.
* Cambiar el estado de las reservas (Programado, Servido/Completado).
* Cancelar reservas.
* Generar los siguientes reportes:
  + Usuarios atendidos por día.
  + Ingresos por día y por servicio.
  + Espacios disponibles en la agenda dado día y servicio.

Un administrador se encarga de gestionar los usuarios internos e indicar el servicio del cual se encarga (pueden haber múltiples usuarios internos encargados de un mismo servicio) y además ejerce control total de la plataforma.

***Notas***:

* ***Los datos suministrados por los usuarios deben cumplir la política de privacidad de datos vigente, es decir las contraseñas de los usuarios deben almacenarse de forma cifrada y la conexión al servidor debe realizarse de forma segura.***
* ***La base de datos debe ser no relacional.***
* ***Se debe utilizar una arquitectura desacoplada, es decir, una API back end y un front end que consuma dicha API.***
* ***La API back end debe estar desarrollada en Express.js***
* ***El front end debe estar desarrollado en React.js***
* ***Se debe utilizar Bootstrap como librería CSS para manejar los estilos de su aplicación.***