**Plataforma Casa de Subastas**

La compañía Dorotheum se encarga de realizar subastas de Relojes.Debido al impacto del covid y la pandemia Dorotheum requiere digitalizar sus servicios. Por lo que lo han contactado a usted y su equipo con el fin de desarrollar una plataforma para realizar las diferentes subastas que ellos ofrecen.

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

Para acceder a los productos subastados el usuario externodeberá registrarse en la plataforma suministrando los siguientes datos:

* Nombre completo.
* Tipo de documento de identidad (Cédula de ciudadanía, pasaporte o cédula de extranjería)
* Número de identificación.
* Fecha de nacimiento.
* Fecha de expedición del documento.

Este usuario externo deberá estar en la capacidad de:

* Visualizar subastas.
* Realizar una puja en una subasta.
* Cancelar una puja en una subasta, siempre y cuando la subasta no haya terminado. Cuando un usuario realice este proceso, el usuario con la subasta inmediatamente anterior debe colocarse como el ganador temporal de la subasta, y se le debe notificar.
* Ver historial de pujas.
* Ver pujas ganadas.
* Ver pujas pérdidas.

Un usuario interno debe estar en la capacidad de:

* Configurar subastas indicando el objeto (foto, nombre y descripción), fecha límite de tiempo de cancelación, fecha y hora inicial de la subasta, fecha y hora límite para terminar la subasta, monto inicial de la subasta.
* Generar los siguientes reportes:
  + Pujas por subasta.
  + Mostrar la puja ganadora de una subasta.
  + Subastas disponibles.
  + Subastas finalizadas dado un rango de fechas.

Un administrador se encarga de gestionar los usuarios internos 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.***