

Definition of done

Frontend:

Diseño General

- El prototipo respeta la identidad visual de la empresa (colores corporativos, tipografía consistente y logotipo visible en cabecera y pie de página).
- El diseño mantiene coherencia en todos los apartados (inicio, detalle de habitación, catálogo, footer).
- Se asegura la correcta disposición de los elementos en diferentes resoluciones (responsive design planificado).

Cabecera / Navegación

- Incluye logo de la empresa y menú con las secciones principales: *Nuestro Hotel*, *Habitaciones*, *Servicios*, *Iniciar Sesión*.
- Barra de búsqueda con **filtros de fecha de ingreso, fecha de salida y categoría** totalmente operativos en el prototipo.
- Botón visible de “Buscar Habitaciones” que ejecuta el filtrado.

Página de Inicio

- Imagen principal destacada con mensaje de bienvenida al hotel.
- Sección de **Servicios y Comodidades** con íconos claros (comedor, piscina, wifi, gym, taxi).
- **Catálogo de habitaciones** con cards que muestran nombre, imagen, y botón “**Ver Detalles**”.

- Footer con:
 - Información de contacto (dirección, teléfono, email).
 - Enlaces rápidos (Nuestro Hotel, Galería, Eventos, Contacto, Trabaja con Nosotros).
 - Sección de suscripción a ofertas con campo de correo electrónico.

Página de Detalle de Habitación

- Muestra galería de imágenes de la habitación seleccionada (con miniaturas navegables).
- Incluye título, precio por noche, descripción clara y botón **“Reservar”**.
- Sección de **Comodidades específicas** de la habitación (wifi, tv cable, estación de café/té).
- Listado de **Otras habitaciones recomendadas** con cards y botones de acceso a detalles.

Consistencia y Usabilidad

- Botones de acción (Ver Detalles, Reservar, Iniciar Sesión, Suscribirse) son visibles y mantienen un mismo estilo de color/hover.
- Iconografía y tipografía son consistentes en todas las vistas.
- Paleta de colores aplicada correctamente: azul, blanco y naranja corporativo en botones y acentos.
- Contraste adecuado para accesibilidad (texto sobre imágenes legible).

Compleitud del Prototipo

- Todas las pantallas principales del flujo del usuario están diseñadas: búsqueda → catálogo → detalle de habitación → acción de reservar.
- Se contempla la experiencia completa de navegación incluyendo la suscripción a ofertas y login.
- No existen secciones “en blanco” o sin contenido representativo.

Backend:

Estructura del proyecto

- El proyecto debe estar organizado bajo un esquema **MVC (Model-View-Controller)**, con separación clara entre **rutas, controladores y modelos**.
- Se debe contar con un archivo principal `app.js` que inicialice el servidor y configure los middlewares necesarios.

Funcionalidades implementadas

- CRUD completo para **Usuarios**: creación, consulta, actualización y eliminación.
- CRUD completo para **Habitaciones**: creación, consulta, actualización y eliminación.
- Consultas específicas:
 - Obtener usuario por correo electrónico.
 - Obtener habitaciones filtradas por tipo.

Validaciones y manejo de errores

- Validación de **campos obligatorios** (nombre, email, contraseña, etc.).
- Validación de **formatos** (correo electrónico válido, precios positivos, fechas coherentes, etc.).
- Manejo adecuado de **errores de servidor (5xx)** y **errores del cliente (4xx)** con mensajes descriptivos.

Calidad del código

- El código debe seguir convenciones estándar de JavaScript (ESLint/Prettier).
- Uso de **promesas/async-await** para el manejo de operaciones asíncronas.
- Manejo de conexiones a base de datos a través de un modelo estructurado.

Despliegue y ejecución

- El servidor debe poder ejecutarse con **npm start**.
- La API debe correr sin errores en **localhost** en el puerto definido.