

## **Proyecto: Aplicativo web para hotel**

### **Contexto**

Su equipo ha sido contratado por un hotel nuevo que desea digitalizar su servicios. El cliente desea una página web a través de la cual se pueda manejar el sistema de reservas y también se pueda administrar los diferentes servicios y actividades que se ofrecen a los clientes.

### **Negocio**

El hotel ya está abierto hace algunos meses. Inicialmente las reservas las manejaban a través de un servicio de terceros, sin embargo, los gastos son altos y saben que si desarrollan su propia aplicación web pueden ahorrar dinero. Uno de los principales atractivos del hotel es la cantidad de servicios que ofrecen adentro y fuera de sus instalaciones. Cuando un cliente quiere un servicio, baja a recepción y le pide a uno de los trabajadores del hotel que contrate el servicio. El problema es que ahorita los servicios se anotan por Excel compartido en OneDrive donde varios trabajadores pueden escribir, lo cual ha traído problemas de integridad de datos.

El proceso para realizar una reserva es el siguiente: el cliente puede ingresar al landing page y ver todos los servicios en general que ofrece el hotel. Sin embargo, cuando el cliente quiere realizar una reserva o ver las que ya ha realizado, es obligatorio que realice login. Para ingresar al portal de cliente, únicamente necesita de correo y contraseña, en caso de no tener una cuenta también se puede registrar, siempre asegurando que no se haya registrado antes.

Ya en el portal puede acceder a todos los servicios, incluido el de reserva de habitaciones. Para realizar una reserva el usuario selecciona una fecha de inicio, una fecha de fin, el tipo de habitación que quiere seleccionar (normal, executive, VIP, Luxury) y la cantidad de personas. Después se realiza una consulta y se muestra si se tiene o no habitaciones disponibles con esos requerimientos. En caso de que haya una habitación se completa la reserva. Es importante que nadie más pueda tomar la habitación reservada durante el periodo de tiempo seleccionado.

Adicionalmente, en el portal del cliente se pueden ver y actualizar la información de usuario. También se debe ofrecer la oportunidad de cancelar una reserva. Actualizar una reserva es más complicado, básicamente el usuario ingresa los nuevos datos, y solo en el caso de que haya disponibilidad se libera la anterior reserva y se realiza la nueva.

Finalmente debe de haber una sección donde el usuario pueda ver las reservas que realizó en el pasado y cuales tiene activas en este momento.

Por otro lado está el portal de los operarios del hotel. Este portal es bastante sencillo, permite ver todas las reservas que se han generado y cancelarlas (en caso de que el cliente llame). También permite acceder al sistema de cuentas de cada habitación para poder agregar un servicio que se quiera contratar. El procedimiento es el siguiente: primero el operario selecciona la habitación, se muestran los datos de la persona que se está quedando, y permite seleccionar entre una variedad de servicios. El operario selecciona la que el cliente pida, y se agrega a su cuenta. En caso de equivocaciones debe ser sencillo poder eliminar un ítem de la cuenta. Tenga presente que la cuenta va a crecer a lo largo de la estadía. También el cliente puede decidir pagar y en ese caso la cuenta se vacía.

Por último en el portal de operarios se debe permitir realizar el check-out. Lo único que se debe revisar es que no hayan gastos en la cuenta o en caso contrario, se paguen. Después de esto el cliente se puede ir tranquilo.

Finalmente, tenemos el portal de administrador. El administrador tiene 3 responsabilidades. La primera, poder realizar el CRUD de los operadores(crearles una cuenta, eliminarla, etc). Segundo, poder realizar el CRUD de servicios que se ofrecen en el hotel. Tercero poder crear habitaciones y asignarles un tipo. También en caso de que una habitación se esté reparando deshabilitarla temporalmente.

## Notas

- Cada tipo de habitación debe tener su descripción en la base de datos.
- No se debe realizar CRUD para los administradores.
- No usar herencia.

## Requerimientos

- **Feature 00:** Defina que clase de hotel está desarrollando y los servicios que ofrece.
- **Feature 01:** Diagrama de Clases, entidad-relación y mockups + navegación.

AC1 Diagrama de clases	<b>Given</b> la información del negocio <b>Then</b> generar un <u>diagrama de clases (UML)</u> para cada una de las entidades. Este debe incluir cardinalidad, nombre de las relaciones y métodos.
AC2 Diagrama entidad-relación	<b>Given</b> la información del negocio <b>Then</b> generar un diagrama entidad-relación. Este debe mostrar PK, FK, campos que pueden ser nulos, cardinalidad, campos únicos.
AC3 Mockups	<b>Given</b> la información del negocio <b>Then</b> generar los mockups del diseño de la página.

	<b>And</b> Cada mockup debe estar asociado a una URL <b>And</b> los mockups se deben poder navegar de manera interactiva o mostrar cómo sería la navegación. <b>And</b> diferenciar a cuáles pantallas puede acceder cada uno de los tipos de usuario
AC4 Logo	<b>Given</b> la información del negocio <b>Then</b> crear un logo <b>And</b> crear un nombre
AC5 Colores	<b>Given</b> la información del negocio <b>Then</b> definir una paleta de colores <a href="https://www.youtube.com/watch?v=yYwEnLYT55c">https://www.youtube.com/watch?v=yYwEnLYT55c</a> (No es necesario usar HSB, ni tener una paleta compleja, lo importante es definir los colores de su aplicación, variaciones, y colores neutrales)

- **Feature 02:** Repositorio – Github

AC6 Repositorio	<b>Given</b> código generado durante el desarrollo <b>Then</b> crear un repositorio en Github público.  <b>And</b> generar una rama de “desarrollo” y otra “producción” o “master”. <sup>1</sup>
AC7 Repositorio	<b>Given</b> los entregables del feature 1 <b>Then</b> cargar los entregables en el repositorio.
AC8 Repositorio	<b>Given</b> los entregables del feature 1 <b>Then</b> personalizar el repositorio con información del negocio y del desarrollo

- **Feature 03:** Base de datos

AC9 Entidades	<b>Given</b> la información de cada componente <b>Then</b> generar las entidades en el sistema y en la BD.
AC10 Repositorios	<b>Given</b> cada una de las entidades <b>Then</b> generar repositorios
AC11 Datos prueba	<b>Given</b> la información de cada componente <b>Then</b> generar datos iniciales de prueba que permitan simular el funcionamiento real de la página. <sup>2</sup>

- **Feature 04:** Main page.

<sup>1</sup> La rama de producción será la que verá el cliente y calificará. En la rama de desarrollo suba y junte los cambios de las demás personas del grupo. Cuando todo esté funcional en la rama de desarrollo suba los cambios a producción. Puede crear más ramas si lo cree necesario.

<sup>2</sup> Para la entrega 1, únicamente es necesario cargar datos relacionados a servicios, clientes, habitaciones y tipos de habitaciones.

AC12 Landing	<b>Given</b> la información del negocio y los diseños generados <b>Then</b> generar una landing page para promocionar el hotel. <b>And</b> permitir navegar al login de usuario.
AC13 Login	<b>Given</b> el correo y contraseña <b>Then</b> permitir al usuario realizar login y acceder a su portal
AC14 Signup	<b>Given</b> la información de una persona <sup>3</sup> <b>When</b> cliente no está registrado <b>Then</b> Registrar cliente en el sistema
AC14 Menú	<b>Given</b> la información de los Servicios <b>Then</b> mostrar la información de estos en tarjetas.
AC15 Detalle producto	<b>Given</b> el identificador de un Servicio <b>Then</b> mostrar sus detalles

- **Feature 05:** Portal de cliente.

AC16 Pedidos activos	<b>Given</b> las credenciales de un cliente. <b>When</b> el cliente haya realizado login. <b>Then</b> este podrá ver sus reservas activas.
AC17 Historial	<b>Given</b> las credenciales de un cliente. <b>When</b> el cliente haya realizado login. <b>Then</b> este podrá ver su historial de reservas.
AC18 Eliminar reserva	<b>Given</b> las credenciales de un cliente. <b>When</b> el cliente haya realizado login. <b>Then</b> este podrá cancelar una reserva.
AC18 modificar reserva	<b>Given</b> las credenciales de un cliente. <b>When</b> el cliente haya realizado login. <b>Then</b> este podrá modificar una reserva.
AC18 Información usuario	<b>Given</b> las credenciales de un cliente. <b>When</b> el cliente haya realizado login. <b>Then</b> este podrá realizar leer y modificar su información personal.

- **Feature 06:** Reserva cliente

---

<sup>3</sup> Los datos mínimos que debe tener todo cliente son: nombre, apellido, correo, contraseña, cédula, teléfono.

AC19	<b>Given</b> una fecha de entrada, fecha de salida, cantidad de personas y tipo de habitación. <b>When</b> el cliente haya realizado login <b>And</b> haya habitación disponible en las fechas asignadas <b>Then</b> generar la reserva para el cliente
------	--

- **Feature 07(portal Operador):** Manejo de servicios

AC21 Login	<b>Given</b> usuario y contraseña <b>Then</b> Acceder al portal del operador.
	<b>When</b> el operador haya realizado login. <b>Then</b> ver datos de reservas <b>And</b> poder filtrar las reservas activas
AC22	<b>When</b> el operador haya realizado login. <b>Given</b> el número de reserva <b>Then</b> ver datos acerca de la reserva y los clientes
AC22	<b>When</b> el operador haya realizado login. <b>Given</b> el número de reserva <b>Then</b> cancelar una reserva <sup>4</sup> .
AC23	<b>When</b> el operador haya realizado login. <b>Given</b> el número de habitación y un servicio a adquirir <b>Then</b> Mostrar la información de los clientes <b>And</b> Mostrar información del servicio <b>And</b> permitir adquirir el servicio <sup>5</sup>
AC24	<b>When</b> el operador haya realizado login. <b>Given</b> un número de habitación <b>Then</b> Mostrar todos los servicios adquiridos durante su estadía <b>And</b> permitir el pago de los servicios
AC25	<b>When</b> el operador haya realizado login. <b>Given</b> un número de habitación <b>Then</b> permitir el pago de todos los servicios adquiridos.

- **Feature 08(Portal administrador):** CRUD habitaciones

AC26 CREATE	<b>When</b> se haya realizado login como administrador <b>Given</b> la información de una habitación <b>Then</b> registrarla en la base de datos
AC27 READ	<b>When</b> se haya realizado login como administrador <b>Then</b> mostrar la información de las habitaciones
	<b>When</b> se haya realizado login como administrador <b>Given</b> la información de una habitación <b>Then</b> modificar su estado a “no disponible”.

<sup>4</sup> Solo se pueden cancelar reservas que todavía no están activas, es decir que si fecha de inicio es posterior a la fecha actual

<sup>5</sup> En caso de que en la habitación no haya ninguna reserva activa, no se debe poder asignar ningún servicio

AC28 DELETE	<b>When</b> se haya realizado login como administrador <b>Given</b> la información de una habitación <b>Then</b> eliminarla de la base de datos
----------------	---

- **Feature 09(Portal administrador):** CRUD servicios

AC34 READ	<b>When</b> se haya realizado login como administrador <b>Then</b> mostrar la información de los diferentes servicios disponibles
AC35 CREATE	<b>When</b> se haya realizado login como administrador <b>Given</b> la información de un servicio <b>Then</b> registrarlo en la base de datos
AC36 UPDATE	<b>When</b> se haya realizado login como administrador <b>Given</b> la información de un servicio <b>Then</b> actualizar sus datos en el sistema
AC37 DELETE	<b>When</b> se haya realizado login como administrador <b>Given</b> la información de un servicio <b>Then</b> eliminarlo del sistema.

- **Feature 10:** Roles

AC40	Given la información de roles y funcionalidades Then implementar el control de acceso para las funcionalidades
------	---

## Entregas

- Primera Entrega
  - Features: 1 al 5
- Segunda Entrega
  - Features: 6 a 9
  - SPA
- Tercera Entrega
  - Features: 10
  - Pruebas automatizadas a nivel del cliente web de al menos 2 casos de uso.
  - 25 pruebas automatizadas a nivel del backend.
  - Las pruebas automáticas deben cargar datos de prueba en la base de datos, los cuales deben removerse después de cada prueba

## Calificación

La nota de los grupos se dividirá en 2 partes (2.5 unidades C/U). La primera será la calificación del cliente, que se calculará a partir de la entrega de cada Sprint. La segunda será la calificación del product manager el cuál revisa que todos los requerimientos a nivel funcional de la página se cumplan.

**Calificación del cliente:**

Cada semana se realizarán entre una y dos entregas en formato de sprint donde los grupos de trabajo deben mostrar adelantos FUNCIONALES de su trabajo. Al final de cada Sprint los grupos deben enviar un video donde se muestre en menos de 5 minutos los avances funcionales del proyecto (no es necesario que se muestre código excepto que se pida).

El cliente se fija de manera general en temas funcionales, UX, UI y CX, para dar su calificación por sprint.

Al final de semestre la calificación del cliente aporta para distinguir cuál ha sido el mejor proyecto durante todo el curso.

**Calificación del Product owner:**

Este revisará al final de cada corte a nivel funcional y de código cómo va el producto general a entregar. Se usará un enfoque de calificación “substractiva” donde cada uno de los grupos empieza con las 2.5 unidades y por cada error cometido se van substrayendo decimas de la nota final.

Algunas de las penalizaciones son:

- Código no comentado
- Malos nombres de variables
- No hay validación de los formularios
- Problemas de rendimiento de la página
- No se cumplen con los requisitos funcionales propuestos
- Código duplicado (puede ser por mal uso de funciones o de componentes)
- No usar los patrones de diseño mostrados en clase
- Mal diseño (de base de datos, diagrama de clases, navegación)
- Usar paquetes para organizar el proyecto de manera coherente
- Usar recursos (imágenes y videos) de alta calidad donde no es necesario