

ITERACION 1

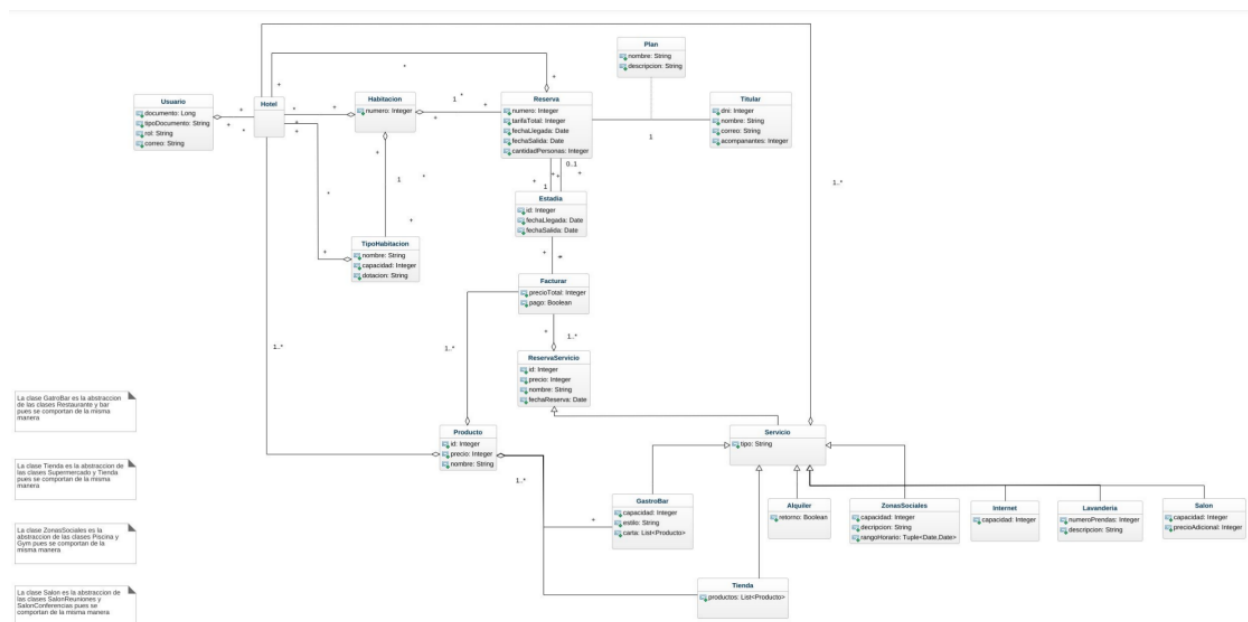
Ángel F. Restrepo Castillo, Juan Pablo Peña Jaime,
Esteban Castelblanco

Universidad de los Andes, Bogotá, Colombia

1 Introducción

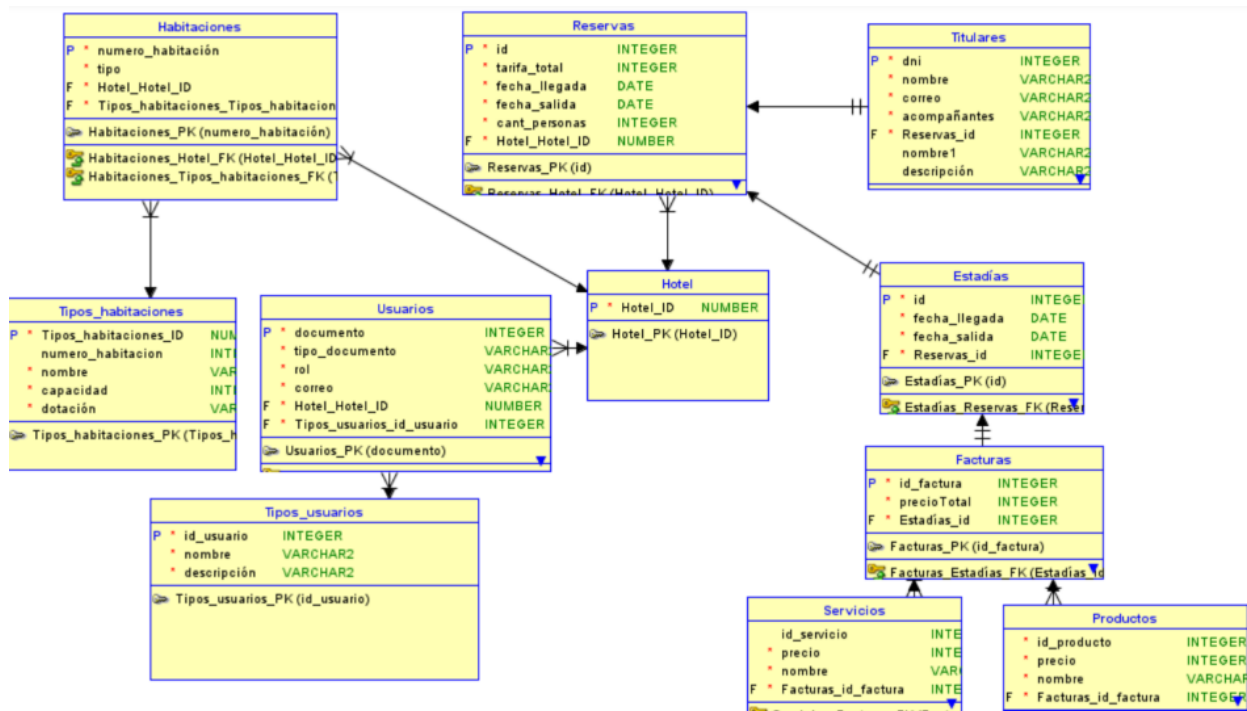
Hotel de los Andes es una empresa que provee principalmente servicios de alojamiento temporal y otros servicios complementarios a sus clientes. Los hoteles adscritos a la cadena hotelera, aunque se rigen por los principios generales de la cadena, se distinguen por su oferta habitacional, por los servicios adicionales ofrecidos y el costo de todos estos servicios. Un hotel se describe entonces mediante su oferta habitacional y su oferta de servicios.

2 Modelo conceptual



Para mejor detalle ver el diagrama en el repositorio

3 Modelo de datos relacional



Para mejor detalle ver el diagrama en el repositorio

Normalización del modelo:

Tipos usuario

1FN (Primera Forma Normal): Es necesario que los atributos de la tabla sean atómicos, para esto, cada uno de los atributos empleados: ID, NOMBRE y DESCRIPCION, cumplen con esa condición. Por lo tanto, si lo está.

2FN (Segunda Forma Normal): Para este caso, no puede haber dependencias parciales. Por el lado de los atributos. Ni NOMBRE ni DESCRIPCIÓN, tienen esta condición. Pues ambos dependen únicamente de la PK de la tabla.

3FN (Tercera Forma Normal): Similarmente, NOMBRE y DESCRIPCIÓN dependen

enteramente de ID como PK. De esta forma, ninguno funciona como relación transitiva de algún atributo diferente. Por lo tanto, se cumple con esta última condición.

USUARIOS

1FN (Primera Forma Normal): Todos los atributos son de índole atómico. De esta forma, se encuentra en 1FN.

2FN (Segunda Forma Normal): NUMERO DOCUMENTO como PK, se encarga de determinar los atributos restantes, impidiendo la creación de dependencias parciales, se encuentra en 2FN.

3FN (Tercera Forma Normal): Al depender TIPO_USUARIO de la columna ID de la tabla TIPOS_USUARIO, como llave primaria de esa tabla, asegura en últimas la imposibilidad de dependencias transitivas de los atributos. Se encuentra, de esta forma en 3FN.

TIPOS_HABITACION

1FN (Primera Forma Normal): No hay atributos repetidos, ni cada atributo cuenta con multivalor, al ser todos atómicos, se encuentra en 1FN.

2FN (Segunda Forma Normal): Para esta ocasión, la clave primaria ID, se encarga de determinar los demás atributos. De esta forma, no hay ningún tipo de relaciones parciales mediante los atributos, asegurando la 2FN.

3FN (Tercera Forma Normal): Para los atributos, cada atributo no primo, depende de manera directa de un atributo que si es primo. De esta forma, no hay formación de relaciones transitivas dentro de la tabla en cuestión.

HABITACIONES

1FN (Primera Forma Normal): Al analizar cada atributo, se caracteriza por poseer valores atómicos, convirtiendo la tabla en 1FN.

2FN (Segunda Forma Normal): Cumple con la 2FN, ya que la clave primaria NUMERO determina completamente todas las demás columnas. No hay dependencias parciales.

3FN (Tercera Forma Normal): No hay dependencias transitivas en esta tabla. La columna TIPO_HABITACION depende directamente de la clave primaria NUMERO.

PLAN

1FN (Primera Forma Normal): No hay valores repartidos, cada atributo es atómico. En conclusión, cumple la 1FN.

2FN (Segunda Forma Normal): La tabla también cumple con la 2FN, ya que la clave primaria ID determina completamente todas las demás columnas. No hay dependencias parciales.

3FN (Tercera Forma Normal): En esta tabla, no hay dependencias transitivas. Cada atributo no clave depende directamente de la clave primaria ID.

RESERVA

1FN (Primera Forma Normal): La tabla cumple con la 1FN, ya que no hay valores repetidos ni listas de valores en ninguna celda. Cada fila tiene valores únicos para ID_HABITACION, TITULAR, PLAN_ELEGIDO, TARIFA_TOTAL, FECHA_LLEGADA y FECHA_SALIDA.

2FN (Segunda Forma Normal): Cumple con la 2FN, ya que la clave primaria ID_HABITACION determina completamente todas las demás columnas. No hay dependencias parciales.

3FN (Tercera Forma Normal): No hay dependencias transitivas en esta tabla. Todos los atributos no clave dependen directamente de la clave primaria ID_HABITACION.

LLEGADACLIENTES

1FN (Primera Forma Normal): No se encuentran valores repetidos y cada uno de los atributos es atómico.

2FN (Segunda Forma Normal): La tabla también cumple con la 2FN, ya que la clave primaria LlegadaClienteID se encarga de determinar las demás columnas. Asegurando la no existencia de relaciones aprciales.

3FN (Tercera Forma Normal): No se evidencia transitividad de relaciones, pues cada atributo depende de forma única de la PK de la tabla.

Aclaración sentencias SQL: Respecto al uso de las sentencias SQL en cuestión. Nos encargamos de realizar cada una de ellas y, adicionalmente, hicimos su debida verificación en el ambiente de trabajo de Oracle. De la anterior forma, pudimos verificar que la lógica no presentara ningún tipo de fallo. Posteriormente, al enfocarnos en realizar la conexión con la página web desarrollada en HTML, presenciamos múltiples obstáculos para completarla de forma exitosa. De esta forma, no fue posible completar esta tarea. No obstante, queremos enfatizar en la posibilidad de realizar las verificaciones de la mayoría de requerimientos funcionales exigidos en el documento, pues al probarlos dentro de SQL, se evidencia la implementación adecuada a nivel lógico. Por lo tanto, no será necesario hacer algún tipo de replanteamiento futuro, ya que únicamente deberemos enfocarnos en temas de conexión vía página web. Las tablas, además, cuentan con su nivel de normalización adecuado para un mejor nivel de funcionamiento.

Resultados Logrados

En el transcurso de la ejecución del plan de pruebas, se alcanzaron resultados significativos que merecen un análisis más detallado. Como se mencionó previamente, uno de los logros destacados fue la confirmación de que los requerimientos funcionales inicialmente definidos se implementaron exitosamente. Las sentencias SQL diseñadas para interactuar con la base de datos se ejecutaron según lo previsto, lo que permitió una manipulación eficiente y precisa de los datos almacenados en el sistema. Además, la creación de la página HTML, que sirve como interfaz principal del sistema, se completó con éxito. Los usuarios pueden acceder a esta plataforma web y realizar diversas acciones, como efectuar reservas, verificar la disponibilidad de habitaciones y gestionar sus perfiles. La experiencia de navegación por la página es fluida y se ha logrado una presentación visual atractiva y amigable para los usuarios finales, lo que aumenta la usabilidad del sistema.

Resultados No Logrados

No obstante, es fundamental reconocer que hubo ciertos objetivos que no se lograron cumplir dentro del alcance del plan de pruebas. La limitación más relevante radicó en la incapacidad para establecer una conexión eficaz entre la página HTML y la base de datos subyacente. A pesar de que las sentencias SQL se desarrollaron y probaron de manera exitosa, la comunicación integral entre la aplicación web y la base de datos presentó desafíos técnicos no resueltos durante el período de pruebas. Este inconveniente afectó la funcionalidad del sistema en términos de recuperación y almacenamiento de datos, lo que representa un aspecto crítico para la operación exitosa del sistema. Adicionalmente, no se pudo garantizar la implementación completa de la totalidad de los requerimientos funcionales originalmente definidos. Esto se debió en gran medida a una comprensión insuficiente de la arquitectura utilizada en el proyecto. Este

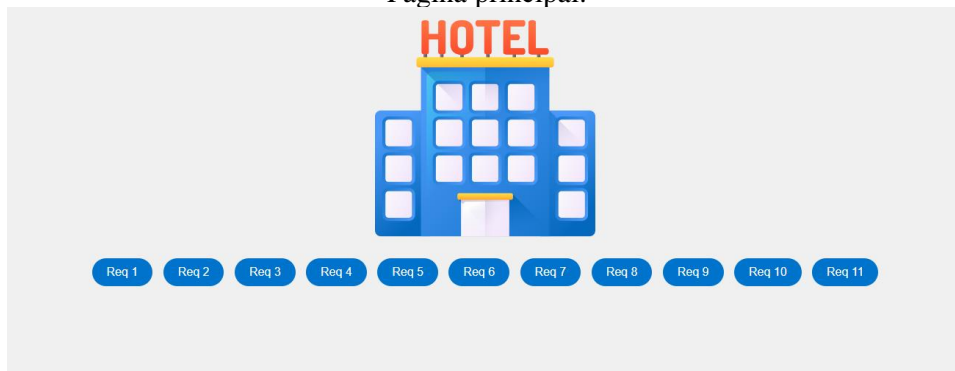
obstáculo condujo a retrasos en el desarrollo y la necesidad de revisar y ajustar ciertos componentes del sistema para alinearlos con las expectativas iniciales. A pesar de estos desafíos, se logró avanzar en la identificación de errores y se aplicaron las correcciones correspondientes.

Reglas de Negocio

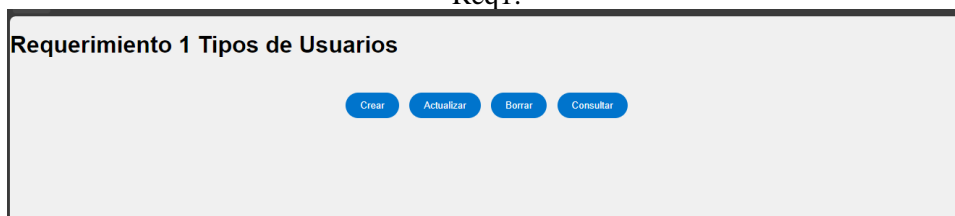
Durante el estudio detallado del caso de estudio, se identificaron reglas de negocio esenciales para el sistema. Estas reglas incluyen la validación de ciertos requerimientos basados en los roles de usuario asignados. La limitación de ciertas acciones a usuarios con roles específicos garantiza la seguridad y la integridad del sistema, al tiempo que se ajusta a los protocolos y políticas establecidos. Además, se observó una relación crítica en el diagrama UML entre las reservas y las estancias. Una reserva puede ser cancelada en cualquier momento, pero si la reserva se completa con éxito, se habilita la posibilidad de consumir productos y servicios adicionales en el hotel. La clase "estadía" desempeña un papel crucial en la validación de que la reserva se haya realizado con éxito, lo que permite al usuario acceder a servicios adicionales, como el uso de instalaciones del hotel.

Resultados HTML:

Página principal:



Req1:



Requerimiento 1 Tipos de Usuarios

Crear

Actualizar

Borrar

Consultar

Crear Datos

Nombre:

Contraseña:

Rol:

Aceptar

Requerimiento 1 Tipos de Usuarios

Crear

Actualizar

Borrar

Consultar

Nombre: juan, Contraseña: sad, Rol: estudiante

```
SELECT * FROM TIPOS_USUARIO
```

ID	NOMBRE	DESCRIPCION
1	24 usuario	un usuario

Req2:

Requerimiento 2 Usuario

Crear

Actualizar

Eliminar

Consultar

Datos Ingresados:

Requerimiento 2 Usuario

Crear Usuario

Tipo de documento:

Número de documento:

Nombre:

Correo electrónico:

Rol:

Aceptar

Requerimiento 2 Usuario

Crear Actualizar Eliminar Consultar

Datos Ingresados:
Tipo de documento: CC
Número de documento: 1232134
Nombre: jorge
Correo electrónico: jorge@uniandes
Rol: estudiante

SELECT * FROM USUARIOS

Valida de Script x Resultado de la Consulta x

SQL | Todas las Filas Recuperadas: 1 en 0,008 segundos

	NUM_DOCUMENTO	TIPO_DOCUMENTO	NOMBRE	CORREO	TIPO_USUARIO
1	25	cc	angel	a@x;	24

Las imágenes anteriores reflejan claramente que, en el grupo de desarrollo, existía un sólido entendimiento y competencia en la creación de páginas HTML, como se evidencia en la implementación exitosa de la interfaz web, además de que creamos e implementamos satisfactoriamente las tablas en nuestra base de datos. Sin embargo, también ponen de manifiesto las dificultades experimentadas en la comprensión de la arquitectura subyacente utilizada para el proyecto. Esta discrepancia entre el dominio de la tecnología front-end y la comprensión de la infraestructura subyacente subraya la necesidad de una formación y colaboración más sólidas en futuros esfuerzos de desarrollo para lograr una implementación más integral y exitosa del sistema.

