

Apartamentos

Pepe Calderón, Fernando Sola, Daniel Ayala, Inma Hernández, Margarita Cruz, Carlos Arévalo, David Ruiz



Escuela Técnica Superior de
Ingeniería Informática

Índice

1. Requisitos	2
2. Entrevista	3
3. Modelo Conceptual	5
4. Diagrama de clases	6
5. Intensión	8
6. Extensión (fragmento)	10
7. Álgebra relacional	11
8. Modelo Tecnológico (MariaDB)	13
9. Script SQL para crear la base de datos	14
10. Script SQL para la carga inicial de datos	15
11. Consultas	16

Apartamentos

Requisitos

Catálogo de Requisitos La transcripción que aparece a continuación corresponde a una entrevista con una emprendedora que quiere crear una empresa de gestión de alquileres de apartamentos turísticos.

Entrevista

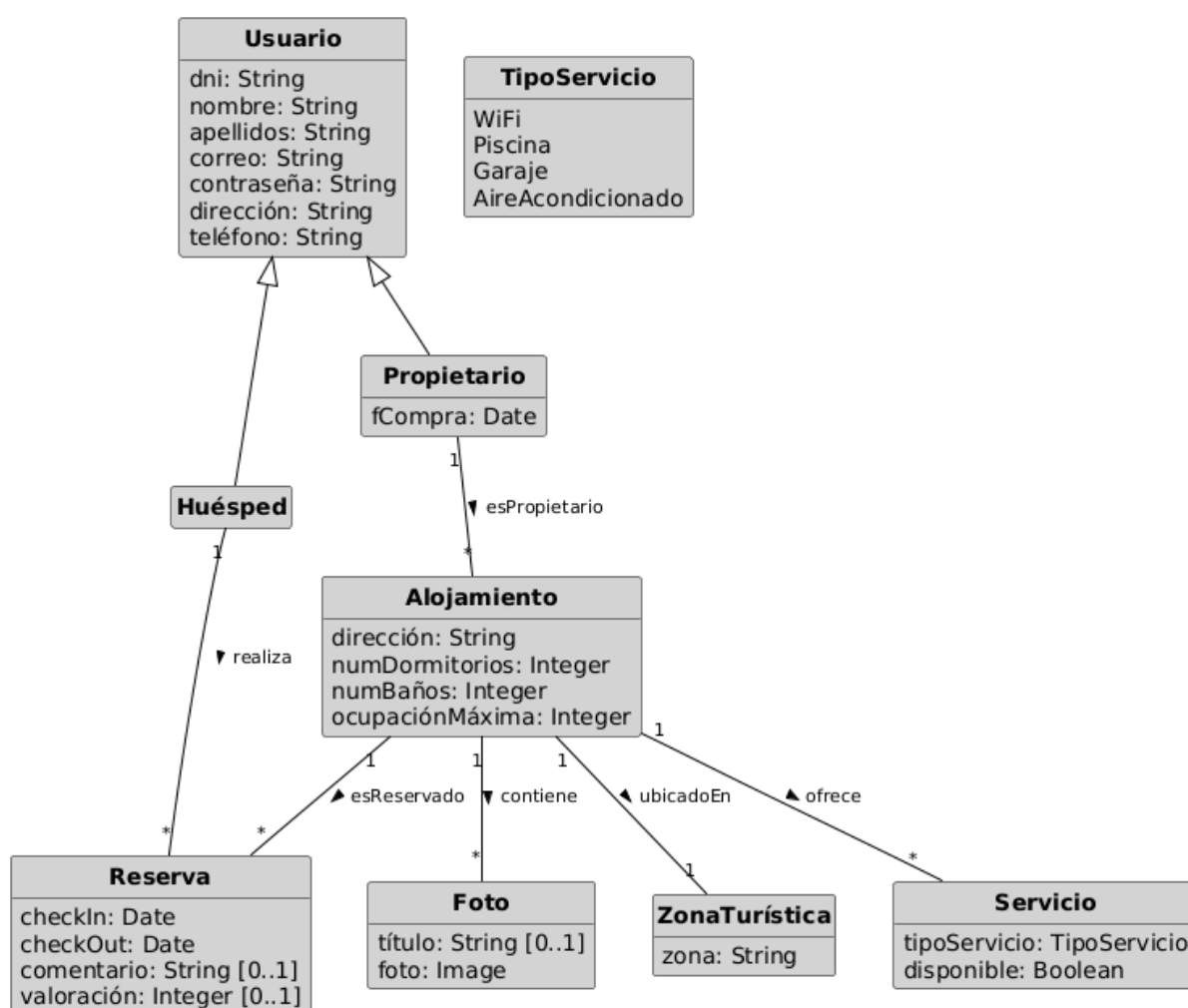
- Pregunta (P): Bien, ¿qué necesita que haga el portal web a desarrollar?
 - Respuesta (R): Bueno, por un lado permitir que los usuarios con alojamientos disponibles los puedan publicar y, por otro lado, que los usuarios que busquen alquilar un alojamiento los puedan buscar por una determinada zona (Costa de la Luz, Costa del Sol, etc.), ver las fotos y solicitar una reserva.
- P: Muy bien, empecemos por los apartamentos.
 - R: Hablemos mejor de alojamientos, no todos son apartamentos, también hay casas rurales y otros tipos distintos.
- P: ¿Pero se tratan todos de la misma forma?
 - R: Sí, para nosotros son iguales.
- P: Bien, pues dígame que información quiere guardar de los alojamientos.
 - R: Pues la dirección, sus características y muchas fotos.
- P: ¿Cuáles son las características?
 - R: Pues el número de dormitorios, el número de baños, la ocupación máxima y si tiene aire acondicionado, wifi, piscina y garaje. Y quien es el propietario, claro. Ah, y la zona turística en la que se encuentra, como le dije antes.
- P: Muy bien, y ¿qué información necesita de los usuarios?
 - R: Bueno, de los usuarios necesito el DNI, nombre y apellidos, correo electrónico, una contraseña para acceder, una dirección y un teléfono de contacto.
- P: ¿Considera usuarios tanto a los propietarios como a los que solicitan reservas?

- R: Sí, porque los propietarios también pueden reservar. Simplemente, si un usuario es propietario, al entrar al portal podrá gestionar sus alojamientos y hacer reservas. Si no tiene alojamientos, sólo podrá hacer reservas.
- P: Muy bien, hábleme de las reservas.
 - R: Bueno, una reserva la hace un usuario sobre un alojamiento, con una fecha de check-in y una fecha de check-out.
- P: ¿Check-in, check-out?
 - R: Sí, entrada y salida, es como se suele hablar en este sector, ¿sabe?
- P: OK, fecha de check-in y fecha de check-out.
 - R: Eso es, tenga en cuenta que el mismo día de check-out se puede hacer un check-in de otra reserva, pero no puede haber más solape de reservas que ese.
- P: ¿Me lo puede explicar?
 - R: Sí claro, dos reservas de un mismo alojamiento no se pueden solapar salvo en el día de check-out de una y el de check-in de la siguiente.
- P: Lo entiendo, debemos controlar el solape de reservas. ¿Algo más sobre reservas?
 - R: Bueno, si la reserva se confirma por parte del propietario y el usuario, me gustaría que el usuario pudiera añadir comentarios con una valoración.
- P: ¿Qué tipo de valoración?
 - R: Por el momento me vale una de esas de estrellitas, de una a cinco estrellas, ¿sabe de qué le hablo?
- P: Sí, de acuerdo, una valoración de 1 a 5 que se visualiza como estrellas.
 - R: Eso es.
- P: ¿Más funcionalidades que necesite del portal?
 - R: Quedan muchos más temas por tratar, como la disponibilidad, la gestión de los pagos, etc., pero para una primera versión creo que puede ser suficiente.

Modelo Conceptual

Modelo conceptual

Diagrama de clases



Modelo relacional

Intensión

Opción A: una relación por subclase (partición completa y disjunta):

```
Usuarios(usuarioId, dni, nombre, apellidos, correo, contrasena, direccion, telefono)
  PK(usuarioId)
  AK(correo)
  AK(dni)
Huespedes(usuarioId)
  PK(usuarioId)
  FK(usuarioId) / Usuarios
Propietarios(usuarioId, fCompra)
  PK(usuarioId)
  FK(usuarioId) / Usuarios
```

Opción B: una sola relación con booleanos (partición completa y solapada):

```
Usuarios(usuarioId, dni, nombre, apellidos, correo, contrasena, direccion, telefono, fCompra, esPropietario, esHuesped)
  PK(usuarioId)
  AK(correo)
  AK(dni)
```

Estructura común del dominio:

```
ZonasTuristicas(zonaId, zona)
  PK(zonaId)
Alojamientos(alojamientoId, propietarioId, zonaId, direccion,
```

```
numDormitorios, numBanos, ocupacionMaxima)
  PK(alojamientoId)
  FK(propietarioId) / Propietarios
  FK(zonaId) / ZonasTuristicas
Reservas(reservaId, huespedId, alojamientoId, checkIn, checkOut,
comentario, valoracion)
  PK(reservaId)
  FK(huespedId) / Huespedes
  FK(alojamientoId) / Alojamientos
Fotos(fotoId, alojamientoId, titulo, fotoURL)
  PK(fotoId)
  FK(alojamientoId) / Alojamientos
Servicios(servicioId, alojamientoId, tipoServicio, disponible)
  PK(servicioId)
  FK(alojamientoId) / Alojamientos
```

Extensión (fragmento)

```
ZonasTuristicas = { (z1, "Costa del Sol") }  
Alojamientos = { (a1, p1, z1, "Calle Mayor 10", 3, 2, 6) }  
Reservas = { (r1, h1, a1, "2024-10-01", "2024-10-10", "Excelente estancia",  
1) }  
Servicios = { (s1, a1, "Wifi", true), (s2, a1, "Piscina", true) }  
Fotos = { (f1, a1, "Salón", "Siempre-viva-salon.jpg"), (f2, a1, "Cocina",  
"Siempre-viva-cocina.jpg") }
```

Álgebra relacional

- Alojamientos con sus propietarios y zonas:

$$APZ \leftarrow \text{Alojamientos} \bowtie \text{Propietarios} \bowtie \text{ZonasTuristicas}$$

- Alojamientos con fotos:

$$AF \leftarrow \text{Alojamientos} \bowtie \text{Fotos}$$

- Alojamientos con servicios:

$$AS \leftarrow \text{Alojamientos} \bowtie \text{Servicios}$$

- Alojamientos con reservas:

$$AR \leftarrow \text{Alojamientos} \bowtie \text{Reservas}$$

- Alojamientos con propietarios, zona, fotos, servicios y reservas:

$$APZFRS \leftarrow APZ \bowtie \text{Fotos} \bowtie \text{Servicios} \bowtie \text{Reservas}$$

- Alojamientos con piscina:

$$APiscina \leftarrow \text{Alojamientos} \bowtie \left(\sigma_{\text{tipoServicio} = \text{"Piscina"} \wedge \text{disponible} = \text{true}} (\text{Servicios}) \right)$$

- Reservas en la Costa del Sol:

$$\text{ReservasCS} \leftarrow \Pi_{\text{alojamientoId}} \left(AR \bowtie \sigma_{\text{zona} = \text{"Costa del Sol"}} (\text{ZonasTuristicas}) \right)$$

- Dormitorios y baños de alojamientos con Wifi:

$$\Pi_{numDormitorios, numBanos} \left(\sigma_{tipoServicio= Wifi \wedge disponible= true} (AS) \right)$$

- Propietarios en una zona concreta:

$$\Pi_{nombre} \left(\sigma_{zona= Playa} (APZ) \right)$$

- Valoración media por alojamiento:

$$AVG_{\gamma}(valoracion) \text{ alojamientoId}(AR)$$

- Número de servicios por alojamiento:

$$COUNT_{\gamma}^{(*)} \text{ alojamientoId}(AS)$$

- Número de fotos por alojamiento:

$$COUNT_{\gamma}^{(*)} \text{ alojamientoId}(AF)$$

- Número de reservas por huésped:

$$COUNT_{\gamma}^{(*)} \text{ huespedId}(Reservas)$$

Modelo Tecnológico (MariaDB)

Modelo tecnológico (MariaDB)

Script SQL para crear la base de datos

Apartamentos/createDB.sql

i Info

Código SQL disponible en: <https://raw.githubusercontent.com/ISSI-US/silence-db/main/Apartamentos/sql/createDB.sql>

Script SQL para la carga inicial de datos

Apartamentos/populateDB.sql

i Info

Código SQL disponible en: <https://raw.githubusercontent.com/ISSI-US/silence-db/main/Apartamentos/sql/populateDB.sql>

Consultas

Apartamentos/queries.sql

i Info

Código SQL disponible en: <https://raw.githubusercontent.com/IISSE-US/silence-db/main/Apartamentos/sql/queries.sql>

i Info

Versión PDF disponible