

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
CAMPUS CORNÉLIO PROCÓPIO  
ENGENHARIA DE COMPUTAÇÃO

LOUISE PACCOLA PECCIN

**PROJETO DE RESERVAS DE APARTAMENTO PARA POO2**

CORNÉLIO PROCÓPIO

JUNHO, 2024

Louise Paccola Peccin

## **PROJETO DE RESERVAS DE APARTAMENTO PARA POO2**

Projeto elaborado na disciplina de Programação Orientada a Objetos 2 do curso de Engenharia da Computação, do Campus Cornélio Procópio da Universidade Tecnológica Federal do Paraná.

Professora: Gisele Alves Santana

CORNÉLIO PROCÓPIO

JUNHO, 2024

## **RESUMO**

O sistema, realizado na linguagem de programação java e com conexão ao Banco de dados MySQL, tem como objetivo facilitar o gerenciamento de reserva para temporada de apartamentos, oferecendo uma plataforma centralizada para proprietários gerenciarem suas propriedades e facilitar o processo de reserva para os clientes.

Palavras chaves : Reservas, proprietário, cliente, apartamentos.

## SUMÁRIO

1	Introdução	1
2	FERRAMENTAS E TECNOLOGIAS	1
3	DESENVOLVIMENTO	1
3.1	Levantamento dos Requisitos	2
3.2	Diagramas de Casos de Uso	2
3.3	Especificação dos Casos de Uso	2
3.4	Diagrama de Classes	8
3.5	Diagrama de Objetos	9
3.6	Diagramas de Sequência	9
3.7	Banco de Dados (pode ser um print das tabelas ou inserir os códigos SQL)	11
3.8	Telas do sistema	12
4	CONCLUSÕES	16

## **1 INTRODUÇÃO**

O desenvolvimento de sistemas de reserva de apartamentos tem transformado a maneira como propriedades são gerenciadas e reservadas, proporcionando maior eficiência e conveniência tanto para proprietários quanto para clientes. Este projeto foca na criação de um sistema de reserva de apartamentos utilizando Java e MySQL, visando automatizar e facilitar o processo de reserva e gestão de propriedades.

O objetivo deste projeto é desenvolver uma aplicação que permita aos usuários pesquisar, visualizar e reservar apartamentos disponíveis. Utilizando tecnologias modernas como Java para a lógica de negócios e MySQL para armazenamento de dados, o sistema oferecerá uma interface intuitiva para clientes realizarem reservas e para proprietários gerenciarem suas propriedades de forma eficaz.

## **2 FERRAMENTAS E TECNOLOGIAS**

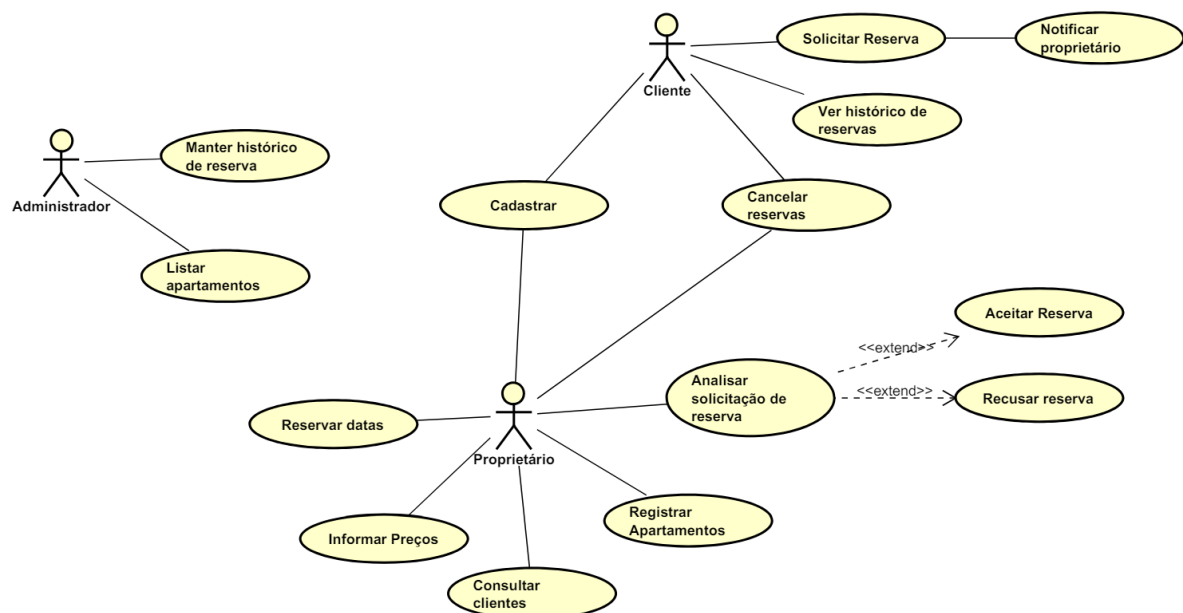
Esse projeto foi desenvolvido utilizando a linguagem Java, pelo IDE NetBeans, MySQL de Banco de dados utilizado para armazenar informações sobre clientes, apartamentos e reservas e o JDBC, API do Java, para conectar e executar consultas SQL no banco de dados MySQL.

## **3 DESENVOLVIMENTO**

### 3.1 Levantamento dos Requisitos

Identificador	Descrição	Prioridade
RF01	O sistema deve registrar Apartamentos	Alta
RF02	O sistema deve registrar Clientes	Alta
RF03	O sistema deve registrar Proprietário	Alta
RF04	O sistema deve reservar datas	Alta
RF05	O sistema deve cancelar reservas	Alta
RF06	O sistema deve manter histórico de reserva	Alta
RF07	O sistema deve informar o preço	Alta
RF08	O sistema deve notificar o proprietário das reservas	Alta
RF09	O sistema deve mostrar as opções de apartamentos	Alta
RF10	O sistema deve permitir o proprietário consultar seus clientes	Alta
RF11	O sistema deve permitir o cliente consultar seu histórico de reserva	Alta

### 3.2 Diagramas de Casos de Uso



### 3.3 Especificação dos Casos de Uso

ID do caso de uso	RF01
Nome do caso de uso	Registrar Apartamentos
Sequência de eventos	<ol style="list-style-type: none"> <li>1) O proprietário faz login no sistema</li> <li>2) O proprietário informa as informações do Apartamento</li> <li>3) O sistema verifica se o Apartamento não foi registrado</li> <li>4) O sistema valida os dados do Apartamento</li> <li>5) O sistema salva as informações do apartamento e gera um identificador para o Apartamento</li> <li>6) O sistema emite uma mensagem de sucesso</li> <li>7) O sistema pergunta se deseja cadastrar mais apartamentos</li> </ol>

ID do caso de uso	RF02
Nome do caso de uso	Cadastrar Clientes

Sequência de eventos	<p>1) O cliente envia seus dados pessoais 2) O sistema verifica se o cliente não existe</p> <p>3) O sistema valida os dados do cliente 4) O sistema salva os dados pessoais do cliente e gera um identificador</p> <p>5) O sistema emite mensagem de sucesso</p> <p>6) O sistema pergunta se deseja fazer login</p>
----------------------	---

ID do caso de uso	RF03
Nome do caso de uso	Cadastrar Proprietário
Sequência de eventos	<p>1) O Proprietário envia seus dados pessoais</p> <p>2) O sistema verifica se o Proprietário não existe</p> <p>3) O sistema valida os dados do Proprietário</p> <p>4) O sistema salva os dados pessoais do Proprietário e gera um identificador</p> <p>5) O sistema emite mensagem de sucesso</p> <p>6) O sistema pergunta se deseja fazer login</p>



ID do caso de uso	RF04
Nome do caso de uso	Fazer reserva
Sequência de eventos	<ol style="list-style-type: none"><li>1) O cliente faz login</li><li>2) O cliente escolhe o Apartamento que deseja fazer sua reserva</li><li>3) O cliente seleciona a data que deseja reservar</li><li>4) O sistema verifica se a data está disponível para reserva daquele apartamento</li><li>5) O sistema emite uma mensagem para confirmar a reserva</li><li>6) O sistema salva as informações do usuário e reserva</li><li>7) O sistema emite mensagem de sucesso</li><li>8) O sistema torna a data selecionada indisponível para outros usuários</li></ol>

ID do caso de uso	RF05
-------------------	------

Nome do caso de uso	Cancelar reservas
Sequência de eventos	<ol style="list-style-type: none"> <li>1) O cliente ou Proprietário faz login no sistema</li> <li>2) O cliente ou proprietário entra em sua área de reservas</li> <li>3) O cliente ou proprietário informa qual reserva já aceita deseja cancelar</li> <li>4) O sistema localiza qual a reserva foi sendo solicitado o cancelamento</li> <li>5) O sistema cancela a reserva e exclui do histórico de reserva</li> <li>6) O sistema emita uma mensagem informando o cancelamento com sucesso</li> </ol>

ID do caso de uso	RF06
Nome do caso de uso	Histórico de reserva
Sequência de eventos	<ol style="list-style-type: none"> <li>1) O sistema deve salvar a reserva realizada pelo cliente</li> <li>2) O sistema deve manter as reservas salvas</li> <li>3) O sistema deve mostrar as reservas salvas para os clientes</li> </ol>
ID do caso de uso	RF07

Nome do caso de uso	Informar o preço
Sequência de eventos	<ol style="list-style-type: none"> <li>1) O sistema deve salvar o preço indicado pelo proprietário na hora do registro de apartamento</li> <li>2) O sistema deve informar esse preço aos clientes na hora da reserva</li> </ol>

ID do caso de uso	RF08
Nome do caso de uso	Notificar o proprietário
Sequência de eventos	<ol style="list-style-type: none"> <li>1) O sistema deve salvar a reserva realizada pelo cliente</li> <li>2) O sistema deve notificar o proprietário pelo Email cadastrado por ele</li> </ol>

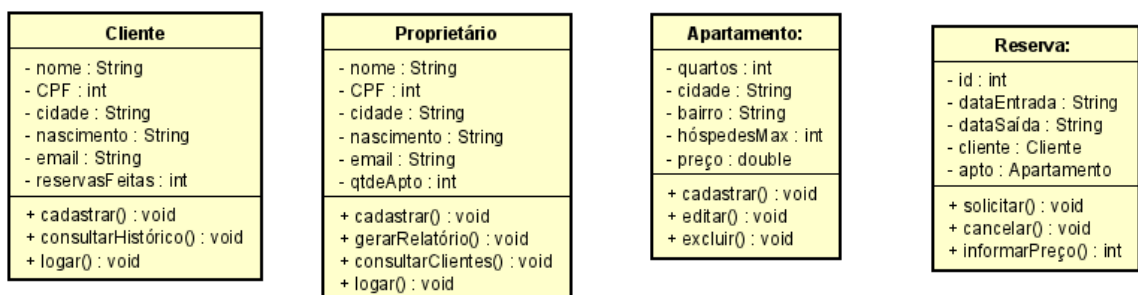
ID do caso de uso	RF09
Nome do caso de uso	Opções de apartamentos
Sequência de eventos	<ol style="list-style-type: none"> <li>1) O sistema deve salvar os apartamentos registrados pelos proprietários</li> <li>2) O sistema deve mostrar as opções de apartamentos aos clientes no momento de reserva</li> </ol>

ID do caso de uso	RF10
Nome do caso de uso	Consulta de cliente

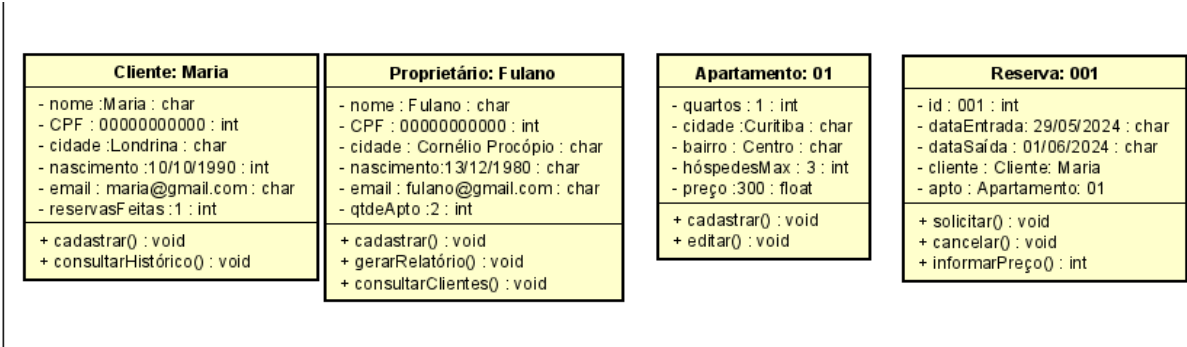
Sequência de eventos	<ol style="list-style-type: none"> <li>1) O proprietário faz login</li> <li>2) O proprietário entra na área de seus clientes</li> <li>3) O sistema deve mostrar os clientes que já reservaram seus apartamentos</li> </ol>
----------------------	--

ID do caso de uso	RF11
Nome do caso de uso	Consulta de reserva
Sequência de eventos	<ol style="list-style-type: none"> <li>1) O cliente faz login</li> <li>2) O cliente entra na área de reservas realizadas</li> <li>3) O sistema deve mostrar todas as reservas já feitas pelo cliente logado</li> </ol>

### 3.4 Diagrama de Classes

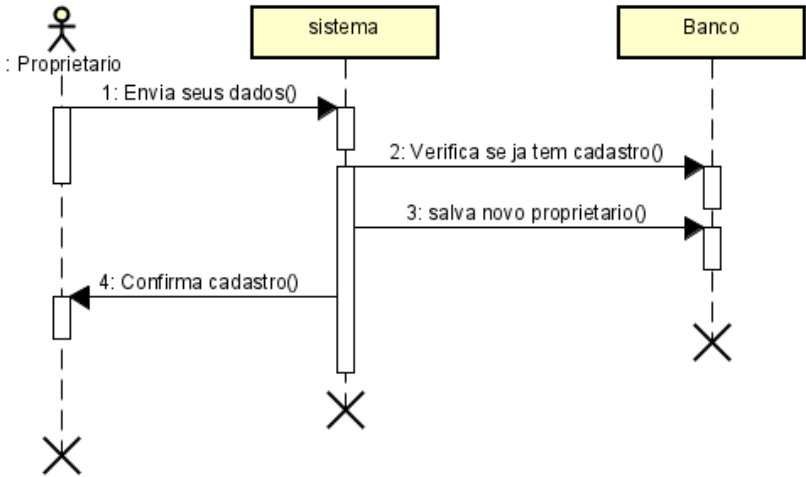


3.5 Diagrama de Objetos

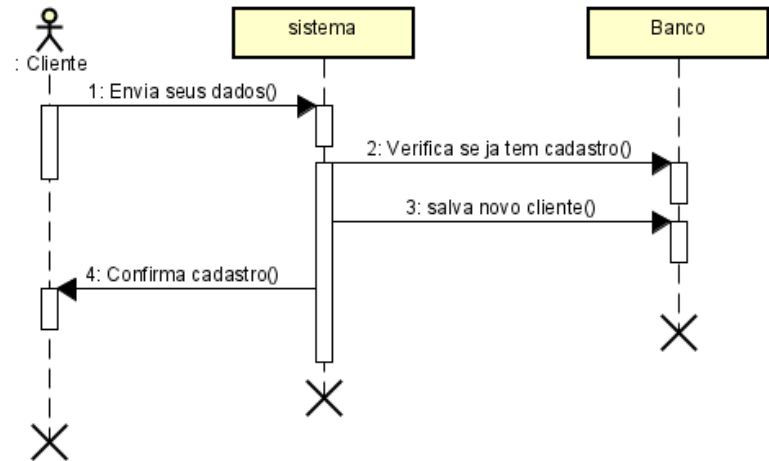


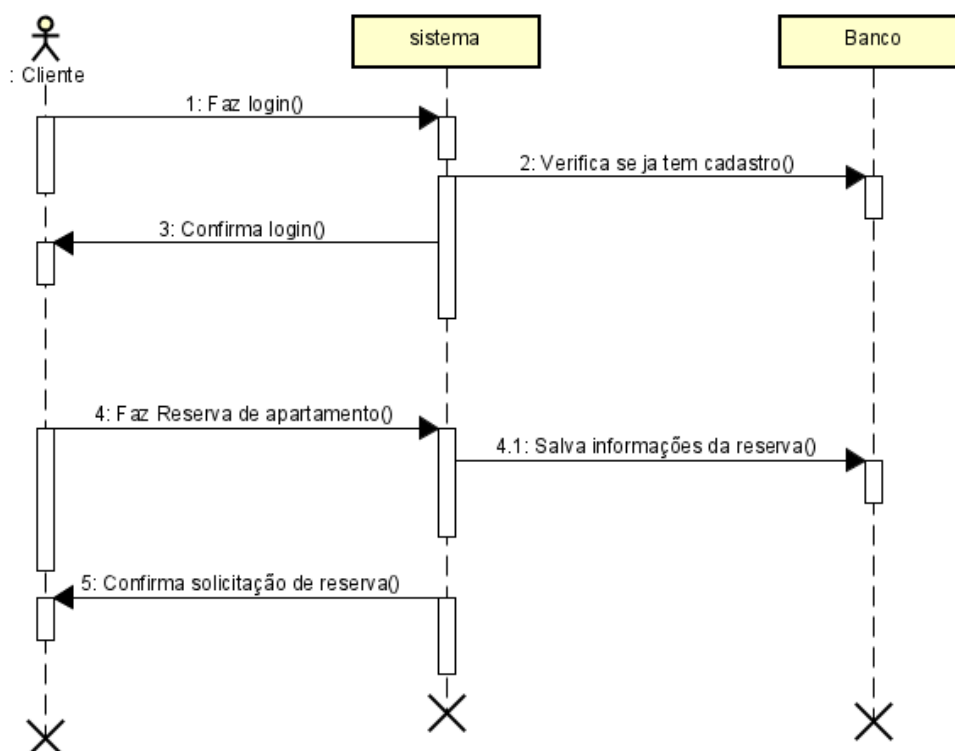
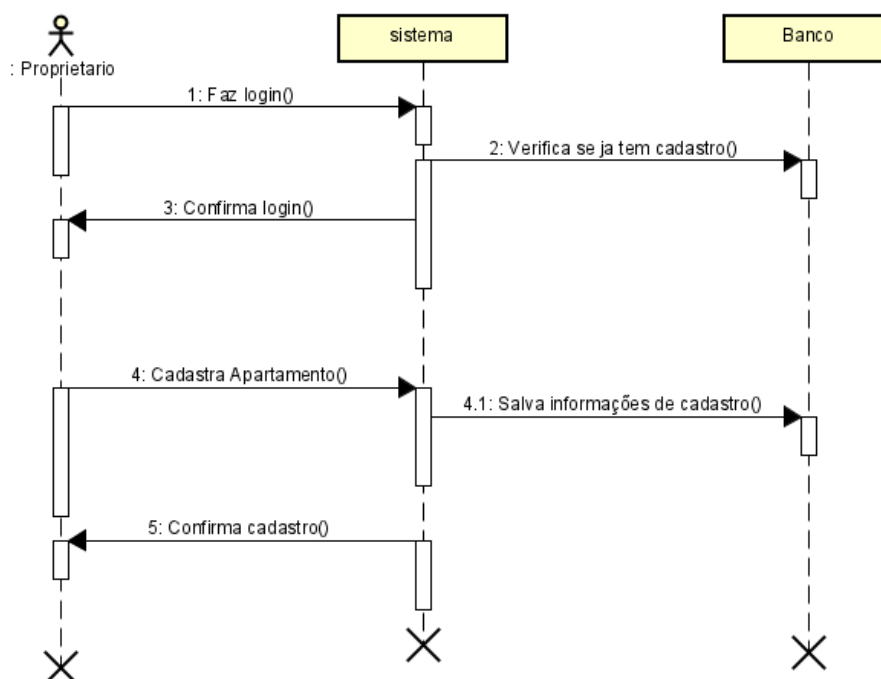
3.6 Diagramas de Sequência

Cadastro de Proprietario:

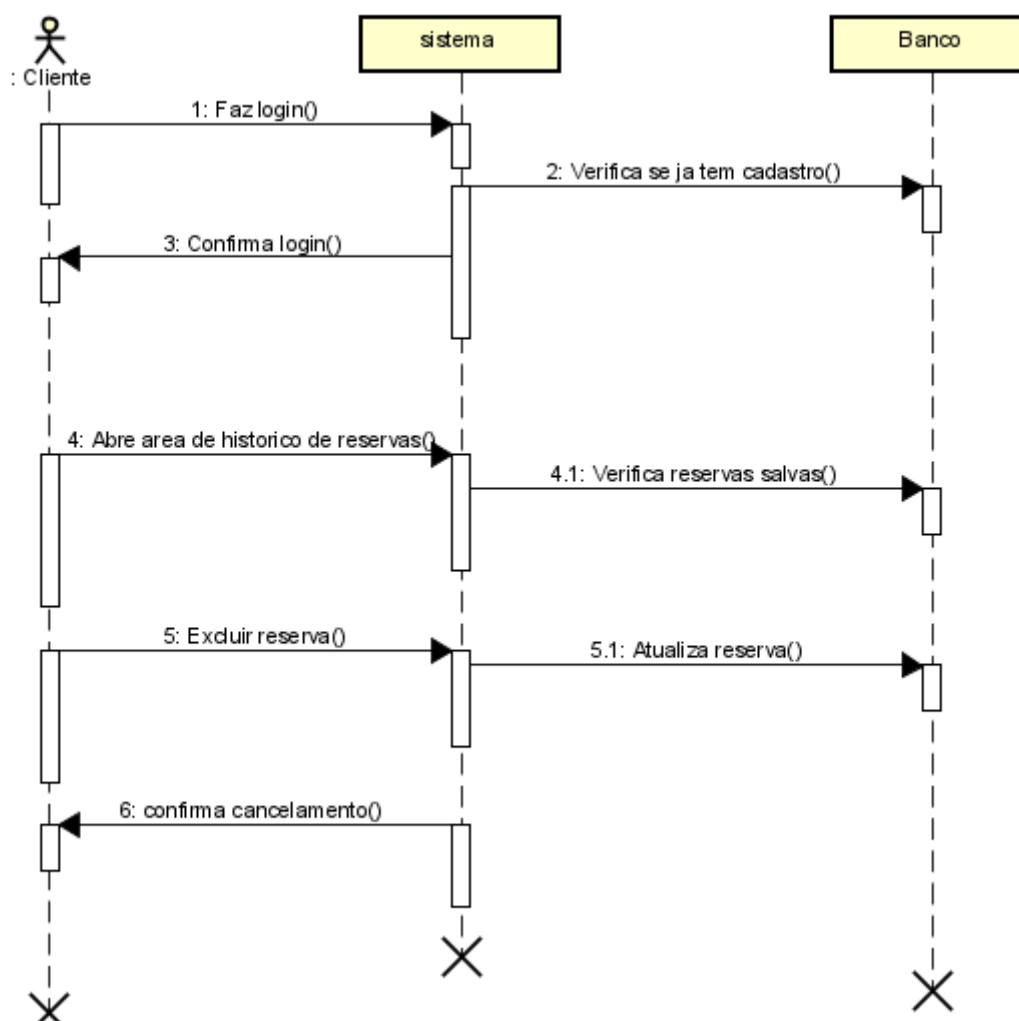


Cadastro de cliente:



**Solicitação de reserva pelo cliente:****Cadastro de Apartamentos:**

### Cancelar reservas:



### 3.7 Banco de Dados

#### Apartamento:

	Field	Type	Null	Key	Default	Extra
►	id	int	NO	PRI	NULL	auto_increment
	cidade	varchar(100)	NO		NULL	
	bairro	varchar(100)	NO		NULL	
	quartos	int	NO		NULL	
	hospedes_max	int	NO		NULL	
	preco	float	NO		NULL	
	descricao	varchar(255)	YES		NULL	
	proprietario_id	int	YES	MUL	NULL	

**Cliente:**

	Field	Type	Null	Key	Default	Extra
►	id	int	NO	PRI	NULL	auto_increment
	nome	varchar(100)	NO		NULL	
	cpf	varchar(14)	NO		NULL	
	cidade	varchar(50)	NO		NULL	
	nascimento	date	NO		NULL	
	email	varchar(100)	NO		NULL	

**Proprietario:**

	Field	Type	Null	Key	Default	Extra
►	id	int	NO	PRI	NULL	auto_increment
	cpf	varchar(14)	YES		NULL	
	nome	varchar(100)	NO		NULL	
	cidade	varchar(100)	YES		NULL	
	nascimento	date	YES		NULL	
	email	varchar(100)	YES		NULL	
	qtde_aptos	int	YES		NULL	

**Reserva:**

	Field	Type	Null	Key	Default	Extra
►	id	int	NO	PRI	NULL	auto_increment
	id_apartamento	int	YES	MUL	NULL	
	id_cliente	int	YES	MUL	NULL	
	data_inicio	date	NO		NULL	
	data_fim	date	NO		NULL	

**3.8 Telas do sistema****Principal:**

## Reservas

Áreas do Cliente

Áreas do Proprietário



**Cadastro clientes:**

## Cliente

### Cadastro

Nome:

CPF:

Cidade:

Nascimento (yyyy-mm-dd):

Email:

**Cadastro Proprietários:**

## Proprietário

### Cadastro

Nome:

CPF:

Cidade:

Nascimento (yyyy-mm-dd):

Email:

Login clientes:

## Cliente

### Login

Email:

Login Proprietários:

## Proprietário

### Login

Email:

Área do cliente:

## Cliente

Reservas realizadas:

Nº da Reserva	Data de Entrada	Data de Saída	Cidade	Bairro

### Área para fazer reservas:

## Cliente

Apartamentos disponiveis

ID	Cidade	Bairro	N° de qu...	Hóspedes	Preço
----	--------	--------	-------------	----------	-------

Data de Entrada (yyyy-mm-dd):

Data de Saída (yyyy-mm-dd):

Quantidade de Pessoas:

1

▼

Confirmar Reserva

Sair

### Área dos Proprietários:

## Proprietário

Cadastro de Apartamentos

Cidade:

Bairro:

Quartos:

1

▼

Hóspedes:

1

▼

Preço diária:

Descrição:

Limpar

Salvar

Seus Apartamentos

ID	Cidade	Bairro	Quartos	Hóspe...	Preço	Descri...
----	--------	--------	---------	----------	-------	-----------

Apagar

Sair

## **4 CONCLUSÕES**

Foi desenvolvido um sistema de reserva de apartamentos utilizando Java e MySQL, focado em facilitar a gestão e reserva de unidades habitacionais de forma eficiente.

Houve dificuldades com a conexão ao banco e modelagem inicial, em partes foram superados.

Este projeto foi base em um grande aprendizado em Java, JDBC e MySQL, melhorando minhas habilidades em POO e Desenvolvimento de Software.

