Universidade/Instituição

Curso de ...

Trabalho de Desenvolvimento de Software

Tema do Trabalho

Aluno(s): Erison Oliveira, Marco Rocha e Victor Gabriel

Professor: Nome

Cidade - Ano

Sumário

1 Proposta de Trabalho 2 1.1 Tema do Trabalho . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2 1.2 Contextualização . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2 1.3 A Situação-Problema . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 2 1.4 Breve Descrição da Solução . . . . . . . . . . . . . . . . . . . . . . . . . . 2

2 Planejamento do Projeto 2 2.1 Previsão e Alocação de Recursos (Humanos e Materiais) . . . . . . . . . . 2 2.2 Problemas do Processo do Sistema Atual . . . . . . . . . . . . . . . . . . . 2

3 O Sistema Proposto 2 3.1 Situação Desejada: Objetivos Gerais e Específicos . . . . . . . . . . . . . . 2

4 Desenvolvimento da Solução 3 4.1 Solução Escolhida . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3 4.2 Arquitetura do Sistema . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3

4.2.1 Escopo da Solução . . . . . . . . . . . . . . . . . . . . . . . . . . . 3 4.2.2 Lista de Requisitos do Sistema . . . . . . . . . . . . . . . . . . . . . 3 4.2.3 Diagrama de Casos de Uso . . . . . . . . . . . . . . . . . . . . . . . 3 4.2.4 Especificações Textuais dos Casos de Uso . . . . . . . . . . . . . . . 3 4.2.5 Modelo Conceitual de Classes . . . . . . . . . . . . . . . . . . . . . 4 4.2.6 Modelo Conceitual de Dados . . . . . . . . . . . . . . . . . . . . . . 4

1

1 Proposta de Trabalho

1.1 Tema do Trabalho

# Sistema de Gerenciamento de Reservas em Restaurantes

**1.2 Contextualização**

A gestão de reservas em restaurantes é um processo crítico para garantir a satisfação dos clientes e a eficiência operacional. Com o aumento da demanda por experiências gastronômicas personalizadas e a necessidade de otimizar a ocupação de mesas, muitos estabelecimentos ainda dependem de métodos manuais, como planilhas ou anotações físicas, o que pode levar a erros, sobreposições de reservas e perda de oportunidades de negócio. A relevância deste tema está na modernização desse processo, reduzindo falhas humanas e melhorando a experiência do cliente. A motivação para o desenvolvimento deste projeto surge da necessidade de oferecer uma solução acessível e intuitiva que auxilie restaurantes a gerenciar reservas de forma eficiente, garantindo maior organização e satisfação tanto para os gestores quanto para os clientes.

**1.3 A Situação-Problema**

Atualmente, muitos restaurantes enfrentam dificuldades no gerenciamento manual de reservas, como: Falta de controle em tempo real das mesas disponíveis, levando a erros na alocação. Dificuldade na comunicação entre a equipe, resultando em reservas duplicadas ou esquecidas. Perda de clientes devido à demora no atendimento ou indisponibilidade não comunicada previamente. Ineficiência operacional, com tempo excessivo gasto no registro e confirmação de reservas. Esses problemas impactam negativamente a experiência do cliente e a lucratividade do estabelecimento, evidenciando a necessidade de um sistema automatizado.

**1.4 Breve Descrição da Solução**

O Sistema de Gerenciamento de Reservas em Restaurantes proposto visa resolver esses desafios por meio de uma plataforma digital que permite: Agendamento online de mesas, com confirmação automática. Visualização em tempo real da disponibilidade de horários e capacidade do restaurante. Notificações para clientes e gestores sobre reservas, cancelamentos e mudanças. Relatórios de ocupação para auxiliar na gestão de estoque e equipe. Dessa forma, o sistema otimizará o processo de reservas, reduzindo erros, aumentando a eficiência e melhorando a experiência do cliente.

**2 Planejamento do Projeto**

2.1 Previsão e Alocação de Recursos (Humanos e Materiais)

• Recursos Humanos: 1 pessoa para o front-end, 1 para o back-end e uma para o banco de dados

• Recursos Materiais: computadores

2.2 Problemas do Processo do Sistema Atual

O sistema atual de reservas, feito de forma manual ou em planilhas simples, apresenta falhas de comunicação entre funcionários, dificuldade em atualizar a disponibilidade em tempo real e pouco controle sobre os dados de clientes. Além disso, não há relatórios que auxiliem na tomada de decisão. Esses gargalos geram atraso no atendimento, reservas duplicadas e perda de clientes.

**3 O Sistema Proposto**

3.1 Situação Desejada: Objetivos Gerais e Específicos • Desenvolver um sistema digital de gerenciamento de reservas para restaurantes, visando otimizar o processo de agendamento de mesas, melhorar a comunicação entre equipe e clientes e aumentar a eficiência operacional do estabelecimento.

2 Objetivos Específicos

* Permitir que clientes realizem reservas de forma online, com feedback imediato.
* Oferecer visualização em tempo real da disponibilidade de mesas.
* Disponibilizar relatórios gerenciais de ocupação e preferências de clientes.
* Proporcionar facilidade de uso com uma interface intuitiva.
* Reduzir erros de comunicação e falhas no processo de reservas manuais.

**4 Desenvolvimento da Solução**

4.1 Solução Escolhida

Descrever brevemente a abordagem tecnológica escolhida –

Usaremos Python como linguagem de programação também usaremos o Django que é um framework web de alto nível e código aberto escrito em Python. Seu objetivo principal é simplificar e acelerar o desenvolvimento de aplicações web, especialmente aquelas baseadas em banco de dados.

O banco de dados usado será o SQlite usando a biblioteca sqlite3.

Arquitetura usada será a MVC, o Controller se comunica com a View e com o Model para gerar a requisição. Mas o Model nunca se comunica com a View, cabendo a função de renderização e entrega ao Controller.

4.2 Arquitetura do Sistema

Arquitetura Escolhida: MVC (Model-View-Controller) dentro de uma aplicação monolítica.

Justificativa: Escolhemos a arquitetura MVC porque ela permite que o sistema cresça sem bagunçar o que já existe. Também facilita a manutenção, já que separa bem a parte da lógica, da interface e do controle. O custo é baixo, pois usaremos o Django, que é gratuito e aberto. Além disso, o sistema pode ser expandido no futuro, como para apps mobile ou pagamentos online, e o Django garante um bom desempenho no uso com banco de dados.

4.2.1 Escopo da Solução

**Dentro do Escopo:** cadastro e login de clientes, reservas online, cancelamento/edição de reservas, visualização de cardápio, relatórios básicos de ocupação.

**Fora do Escopo:** integração com meios de pagamento online, aplicativo mobile, integração com sistemas de delivery.

.

4.2.2 Lista de Requisitos do Sistema

**• Requisitos Funcionais:** funcionalidades que o sistema deve possuir.

### RF01. Cadastro de Clientes

* O sistema deve permitir que novos clientes se cadastrem com nome, e-mail, telefone e senha.

### RF02. Autenticação de Clientes

* O sistema deve permitir login de clientes com e-mail e senha.

### RF03. Visualização de Cardápio

* O sistema deve exibir o cardápio atualizado, com categorias (entradas, pratos principais, bebidas, sobremesas), descrição e preço.

### RF04. Realização de Reservas

* O sistema deve permitir que o cliente escolha:  
  + Data e horário
  + Número de pessoas
  + Preferência de local (interna/externa)
* O sistema deve confirmar a reserva por notificação.

### RF05. Cancelamento e Edição de Reserva

* O cliente poderá alterar ou cancelar uma reserva com antecedência mínima definida (ex: 2 horas).

### RF06. Relatórios Gerenciais

* O sistema deve gerar relatórios exportáveis com:  
  + Taxa de ocupação por período
  + Número total de reservas
  + Preferências de horário

**• Requisitos Não Funcionais:** desempenho, segurança, usabilidade, disponibilidade, etc.

### RNF01. Usabilidade

* A interface deve ser intuitiva

### RNF02. Disponibilidade

* O sistema deve estar disponível 24h, exceto durante manutenções programadas.

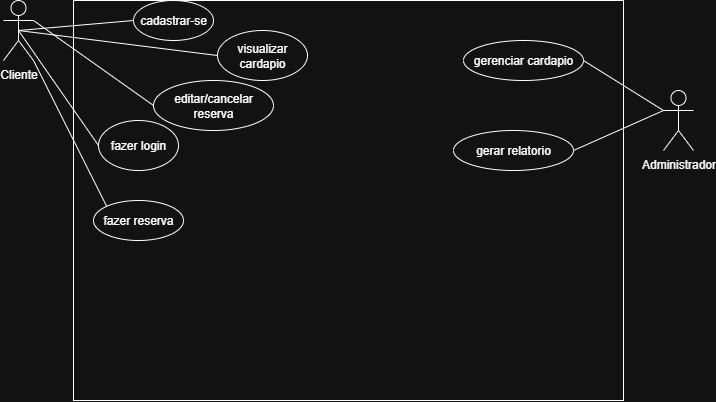
### RNF03. Segurança

* O sistema deve obedecer à LGPD (Lei Geral de Proteção de Dados).

### RNF04. Desempenho

* O tempo de resposta das páginas não deve ultrapassar 3 segundos, mesmo em horários de pico.

4.2.3 Diagrama de Casos de Uso



Inserir diagrama UML com principais interações do sistema. *(Utilizar pacotes ou incluir imagem com \includegraphics{})*

4.2.4 Especificações Textuais dos Casos de Uso

### Caso de Uso: Fazer Reserva

* Ator: Cliente
* Pré-condição: Cliente logado no sistema.

Passos principais:

1. Cliente clica em "Fazer Reserva".
2. Informar data, horário, número de pessoas e local.
3. Sistema verifica se há mesa disponível.
4. Se disponível → reserva é criada no banco e o cliente recebe confirmação.

Alternativa:

* Se não tiver mesa disponível → sistema sugere outros horários.

Exceções:

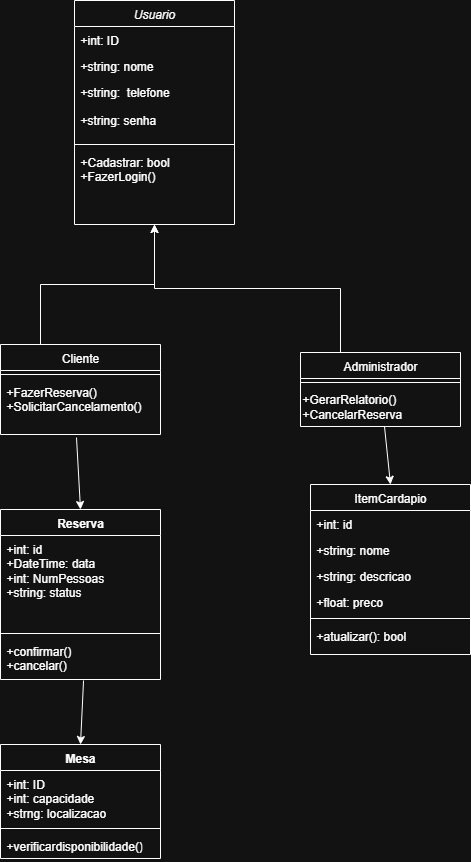
* Erro de conexão ou dados inválidos (ex: número de pessoas maior que o permitido).

Resultado final:

* Reserva salva no banco e aparece para o cliente e o administrador.

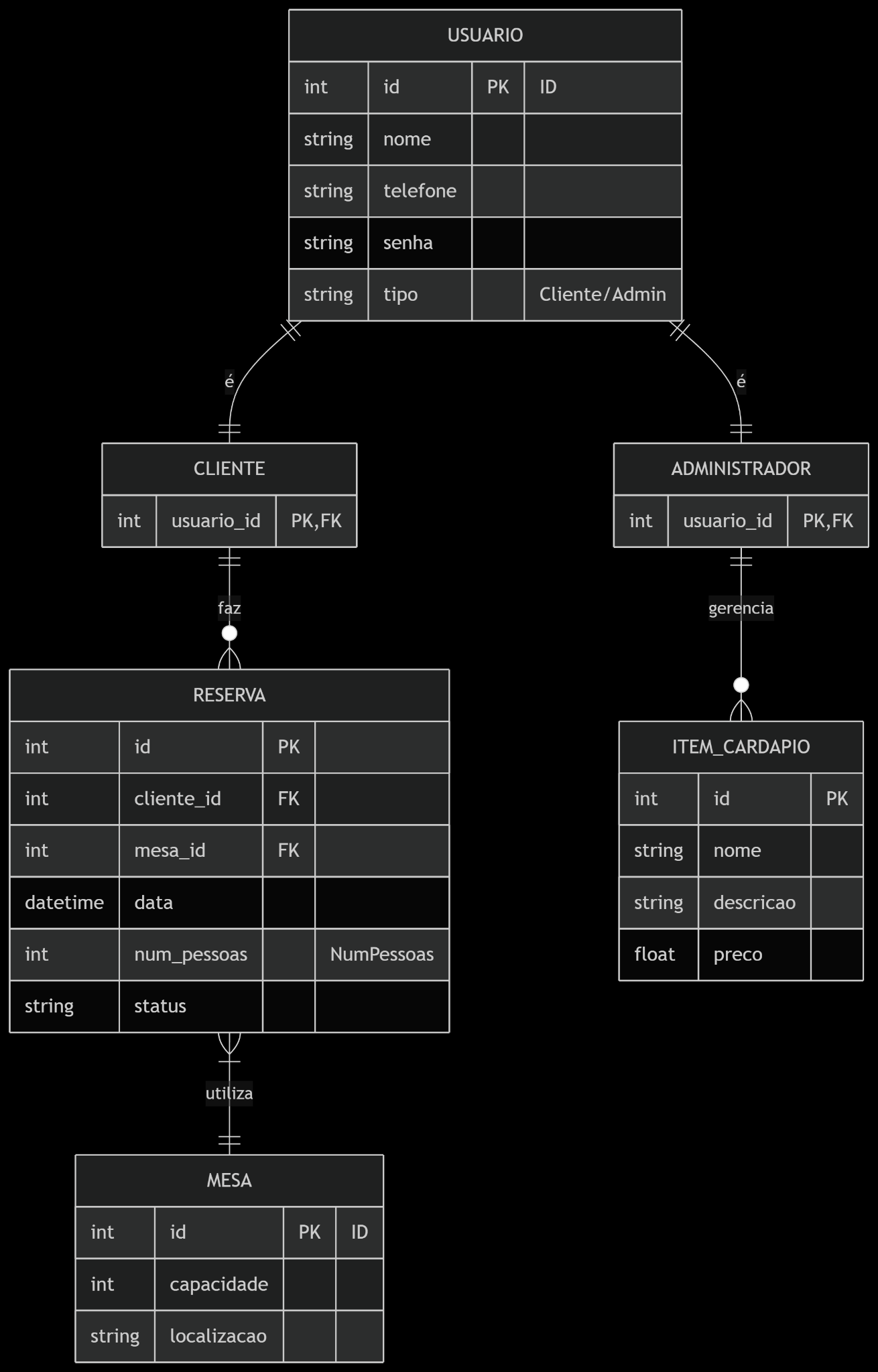
3

4.2.5 Modelo Conceitual de Classes



Inserir diagrama UML de classes, atributos e relacionamentos.

4.2.6 Modelo Conceitual de Dados



Inserir modelo entidade-relacionamento ou diagrama lógico do banco de dados. 4