

Mooca’s Pizza

Especificação de Sistema

Easy Queue

Versão 1.0

Raí Jóia miquilino valencio, Matheus Henrique Oliveira Chuang, Bruno Godoy Dias

Histórico da Revisão

| **Data** | **Versão** | **Descrição** | **Autor** |
| --- | --- | --- | --- |
| 23/04/2025 | 1.0 | Reunião inicial dos requisitos coletados. | Bruno Dias |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Índice Analítico

[**1. Identificação 5**](#_heading=h.oezjs91p2ie0)

[1.1 Nome aluno 01: Raí jóia miquilino valencio 5](#_heading=h.l47r5b58u8r)

[1.2 RA aluno 01: 82318841 5](#_heading=h.2r88g8b7a9ad)

[1.3 Nome aluno 02: Matheus Henrique Oliveira Chuang 5](#_heading=)

[1.4 RA aluno 02: 823165173 5](#_heading=)

[1.5 Nome aluno 03: Bruno Godoy Dias 5](#_heading=)

[1.6 RA aluno 03: 82311358 5](#_heading=)

[1.7 Turmas: CCP1AN-MCD3-25085187 / CCP1AN-MCD3-25070036 5](#_heading=h.jz3k42dhcxs6)

[1.8 Professores(as): Carlos Enrique Lopez Noriega e Robson Calvetti 5](#_heading=h.jwubkwm7ef3q)

[1.9 Disciplinas envolvidas: 5](#_heading=h.xuqwhz3jkuz1)

[1.9.1 Disciplina 1: Gestão e qualidade de Software 5](#_heading=h.sfuwihgxuuib)

[1.9.2 Disciplina 2: Modelos, métodos e técnicas da engenharia de software 5](#_heading=h.dfrpwni7bpmr)

[**2. Especificação Geral do Sistema. 5**](#_heading=h.tsauzdifuxcw)

[2.1 Escopo](#_heading=h.bsogxq4gj9l3) [5](#_heading=h.s0lnog8ft9dg)

[2.2 Descrição dos requisitos de usuário. 6](#_heading=h.mockmhtp57ca)

[2.2.1 Breve Descrição 6](#_heading=h.5ws5b5c91gk0)

[2.2.2 Requisitos do produto 6](#_heading=h.9iwrbnsa4coz)

[**3. Distribuição das Funcionalidades (Modelagem). 7**](#_heading=h.e4bncuyl3gt8)

[3.1 Funcionalidades do Sistema 7](#_heading=h.hctszfhk6gvj)

[3.1.1 Descrição da Funcionalidade-1 do Sistema 7](#_heading=h.jpdgldua9ylp)

[3.1.2 Descrição da Funcionalidade-2 do Sistema 7](#_heading=h.g10q4sl041aw)

[**4. Representação Gráfica do Software. 7**](#_heading=h.n0to5a5qlu15)

[4.1 Diagrama de Casos de Uso 7](#_heading=h.8psda0tso3x7)

[4.1.1 Especificação do Caso de Uso-1 7](#_heading=h.s67ow8hlgyzz)

[4.1.2 Especificação do Caso de Uso-2 7](#_heading=h.ya0m1ez9y1xe)

[4.2 Diagrama de Classes. 7](#_heading=h.r69916flwytx)

[4.3 Diagrama de Sequência. 7](#_heading=h.xo4uocsompd0)

[4.3.1 Diagrama de Sequência-1. 8](#_heading=h.ub3jv0obqybr)

[4.3.2 Diagrama de Sequência-2. 8](#_heading=h.kzzlxitoo7i7)

[4.4 Diagrama de Comunicação/Colaboração. 8](#_heading=h.x2f3eftq0cda)

[4.5 Diagrama de Atividades. 8](#_heading=h.s5ac1z9kso15)

[4.6 Diagrama de Estados. 8](#_heading=h.npn3syyjfom5)

[4.6.1 Diagrama de Estados da classe-x atributo-x. 8](#_heading=h.uc9l4s6id3fe)

[4.6.2 Diagrama de Estados da classe-y atributo-y. 8](#_heading=h.2v4knun1587g)

[4.7 Diagrama de Implementação. 8](#_heading=h.ykk2dyyhg7xh)

[4.7.1 Diagrama de Pacotes. 8](#_heading=h.5zpdxbjaq6v0)

[4.7.2 Diagrama de Componentes. 8](#_heading=h.hrkbav9crj5l)

[4.7.3 Diagrama de Implantação/Distribuição 8](#_heading=h.71pwv4h9laty)

[4.8 Diagrama de Objetos. 8](#_heading=h.zf6fktijxdt2)

[4.9 Diagrama de Estrutura Composta. 8](#_heading=h.bmix1gwcg0px)

[4.10 Diagrama de Interação Geral. 9](#_heading=h.qr71mgegxk8v)

[4.11 Diagrama de Tempo ou Temporal. 9](#_heading=h.mm1vqyefhmvu)

Easy Queue

# Identificação

## Nome aluno 01: Raí jóia miquilino valencio

## RA aluno 01: 82318841

## Nome aluno 02: Matheus Henrique Oliveira Chuang

## RA aluno 02: 823165173

## Nome aluno 03: Bruno Godoy Dias

## RA aluno 03: 82311358

## Turmas: CCP1AN-MCD3-25085187 / CCP1AN-MCD3-25070036

## Professores(as): Carlos Enrique Lopez Noriega e Robson Calvetti

## Disciplinas envolvidas:

### Disciplina 1: Gestão e qualidade de Software

#### Professor responsável: Robson Calvetti

#### Ponto a ser verificado:

*[Solicitação do professor da disciplina para ser desenvolvida no projeto]*

### Disciplina 2: Modelos, métodos e técnicas da engenharia de software

#### Professor responsável: Carlos Enrique Lopez Noriega

# Especificação Geral do Sistema.

## Escopo

## Este documento de Especificação de Sistema descreve o escopo, funcionalidades e diretrizes para o desenvolvimento do sistema Easy Queue, uma solução digital voltada à modernização do atendimento e da gestão operacional da pizzaria Mooca’s Pizza, sem comprometer a experiência acolhedora e tradicional que caracteriza o estabelecimento.

## O escopo cobre todos os processos relacionados ao atendimento ao cliente — desde a recepção até o pagamento — bem como a comunicação entre salão, cozinha e administração. O sistema afetará positivamente os fluxos de trabalho internos (atendentes, garçons, pizzaiolos e caixa), os dispositivos utilizados (tablets, celulares e computadores), e estará integrado a soluções de pagamento e notificações móveis.

## Este sistema deverá operar em múltiplas plataformas (Android, iOS, navegadores modernos), exigindo conexão com a internet, servidores para controle de dados e interface amigável para os usuários internos e clientes. Serão considerados aspectos de segurança, usabilidade e performance para assegurar estabilidade mesmo em horários de pico.

## 

## Descrição dos requisitos de usuário.

### Breve Descrição

Do ponto de vista dos clientes da **Mooca’s Pizza**, o sistema **Easy Queue** tem como principal finalidade oferecer uma experiência mais ágil, organizada e moderna sem abrir mão da essência familiar e do atendimento personalizado da casa.

A solução digital permitirá que os clientes entrem na fila de espera de maneira prática via QR Code, sejam notificados sobre a disponibilidade de mesa, e recebam atendimento personalizado com o garçom os chamando pelo nome. Durante sua permanência, os pedidos são feitos e entregues com mais rapidez e precisão, mantendo a qualidade do serviço.

Além disso, o cliente poderá visualizar sua conta, dividir o pagamento com os amigos e finalizar tudo diretamente pela mesa, sem enfrentar filas. O sistema reforça a proposta de unir o melhor da tecnologia com o charme da tradição italiana, tornando a visita à pizzaria mais confortável, interativa e memorável.

### Requisitos do produto

Ter uma recepção digital rápida e prática, com cadastro simples de nome e telefone.

Permitir que o cliente entre na fila de espera escaneando um QR Code.

Ser avisado no celular quando a mesa estiver pronta, com imagem de referência da mesa.

Manter o atendimento acolhedor, com o garçom chamando o cliente pelo nome.

Ter um garçom fixo responsável por no máximo 5 mesas, para preservar o relacionamento.

Eliminar papel nos pedidos, garantindo mais agilidade e menos erros.

Permitir que o cliente chame o garçom com um botão pelo app.

Visualizar e dividir a conta de forma simples, com possibilidade de pagar direto pelo app.

Aceitar múltiplas formas de pagamento: Pix, VR, cartão, aproximação e QR Code.

Possibilidade de dar gorjeta pelo próprio app.

Acompanhar relatórios de vendas, desempenho da equipe e controlar estoque básico.

Usar uma identidade visual com cores que remetam à tradição italiana e ao ambiente familiar.

## 

# Distribuição das Funcionalidades (Modelagem).

*[ (explicação deste tópico 4) E a atividade de construção de modelos que explique as características e o comportamento da parte software e da parte hardware do Sistema a ser desenvolvido. Em sua construção os modelos devem ser usados na elaboração e representação da arquitetura das características e funcionalidades que o Sistema deve prover e contribui para o seu planejamento e na sua construção. Em geral implica na construção de modelos gráficos que simbolizam os artefatos dos Sistemas de software e hardware utilizados e os seus inter-relacionamentos. A escolha da técnica e da ferramenta a ser utilizada, deve ser escolhida de acordo com as características do Sistema a ser desenvolvido com as características do ambiente de desenvolvimento e dos envolvidos. Pode-se utilizar representação através de fluxogramas, linguagens gráficas, ferramentas e técnicas direcionadas para o Sistema a ser produzido. ]*

## Funcionalidades do Sistema

*[Listar as funcionalidades do sistema identificadas no enunciado]*

### Descrição da Funcionalidade-1 do Sistema

*[Descrever cada funcionalidade do sistema pela visão do software]*

### Descrição da Funcionalidade-2 do Sistema

*[Descrever cada funcionalidade do sistema pela visão do software]*

# Representação Gráfica do Software.

## Diagrama de Casos de Uso

*[Representar o conjunto de comportamentos de alto nível do sistema - A Representação gráfica das funcionalidades do sistema]*

### Especificação do Caso de Uso-1

*[Especificação/detalhamento de cada caso de uso do sistema, o detalhamento deve ser feito de forma textual e pode ser demonstrado através de outros diagramas]*

### Especificação do Caso de Uso-2

*[Especificação/detalhamento de cada caso de uso do sistema, o detalhamento deve ser feito de forma textual e pode ser demonstrado através de outros diagramas]*

## Diagrama de Classes.

*[Representar a coleção de classes do sistema e seus inter-relacionamentos]*

## Diagrama de Sequência.

*[Representar uma perspectiva, orientada por tempo, da colaboração/troca de mensagens entre os objetos do sistema, para cada instancia/cenário diferentes do sistema]*

### Diagrama de Sequência-1.

*[Representação do cenário-1]*

### Diagrama de Sequência-2.

*[Representação do cenário-2]*

## Diagrama de Comunicação/Colaboração.

*[Representar o envio ou o recebimento de dados entre um ator e um caso de uso]*

## Diagrama de Atividades.

*[Representar o fluxo de tarefas que podem ser executadas pelo sistema ou por um ator]*

## Diagrama de Estados.

*[Representar um conjunto de estados dos objetos das classes]*

### Diagrama de Estados da classe-x atributo-x.

*[Representar um conjunto de estados que um objeto pode estar e os “gatilhos” que estimulam a transição do objeto de um estado para outro]*

### Diagrama de Estados da classe-y atributo-y.

*[Representar um conjunto de estados que um objeto pode estar e os “gatilhos” que estimulam a transição do objeto de um estado para outro]*

## Diagrama de Implementação.

*[Representar a Arquitetura Física do sistema]*

### Diagrama de Pacotes.

*[Representar os pacotes ou pedaços do sistema divididos em agrupamentos lógicos mostrando as dependências entre eles, representaros módulos físicos que o desenvolvedor deve utilizar]*

### Diagrama de Componentes.

*[Representar a coleção de componentes de software e seus inter-relacionamentos]*

### Diagrama de Implantação/Distribuição

*[Representar a arquitetura dos processadores, dispositivos e os componentes de software que rodam na arquitetura utilizada no sistema, a modelagem das plataformas físicas e conexões de rede que do sistema]*

## Diagrama de Objetos.

*[Representar os objetos que foram instanciados das classes]*

## Diagrama de Estrutura Composta.

*[Representar os relacionamentos entre os elementos/colaboração interna de classes, interfaces ou componentes para especificar uma funcionalidade do sistema]*

## Diagrama de Interação Geral.

*[Representar o envio ou o recebimento de dados entre um ator e um caso de uso]*

## Diagrama de Tempo ou Temporal.

*[Representar o comportamento dos objetos e sua interação em uma escala de tempo, focalizando as condições que mudam no decorrer desse período]*