**EFOOD**

Documento de Visão

Histórico de Revisões

| **Data** | **Versão** | **Descrição** | **Autor** |
| --- | --- | --- | --- |
| 16/10/2022 | 1.0 | Primeira versão do documento | João Paulo Maués |
| 17/10/2022 | 1.1 | Revisão do documento | Bruno Dantas |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Índice

[**1**](#_heading=h.1fob9te) **INTRODUÇÃO 4**

[**1.1**](#_heading=h.3znysh7) **Objetivo 4**

[**1.2**](#_heading=h.tyjcwt) **Escopo 4**

[**1.3**](#_heading=h.3dy6vkm) **Fora do Escopo 4**

[**1.4**](#_heading=h.1t3h5sf) **Definições, Acrônimos, e Abreviações 4**

[**2**](#_heading=h.2s8eyo1) **CONTEXTO DO NEGÓCIO 4**

[**2.1**](#_heading=h.17dp8vu) **Relato do Problema 4**

[**2.2**](#_heading=h.3rdcrjn) **Processo de Negócio 4**

[**2.3**](#_heading=h.26in1rg) **Riscos Identificados 5**

[**3**](#_heading=h.lnxbz9) **STAKEHOLDERS DO PROJETO 5**

[**3.1**](#_heading=h.35nkun2) **Papéis e Responsabilidades 5**

[**3.2**](#_heading=h.44sinio) **Relação dos Stakeholders 5**

[**4**](#_heading=h.2jxsxqh) **CARACTERÍSTICAS DO PRODUTO 6**

[**5**](#_heading=h.4i7ojhp) **RESTRIÇÕES 6**

[**5.1**](#_heading=h.2xcytpi) **Linguagens de Programação 6**

[**5.2**](#_heading=h.1ci93xb) **Padrões de Desenvolvimento 7**

[**5.3**](#_heading=h.2bn6wsx) **Requisitos Não Funcionais 7**

# INTRODUÇÃO

## Objetivo

O objetivo do software pode ser resumido a desempenhar o papel de um sistema cujo objetivo é permitir que os usuários consigam realizar seus pedidos de entrega de comida e que o pagamento seja efetuado por esta mesma plataforma.

## Escopo

Este projeto irá automatizar apenas a parte de pedido de comida e acompanhamento desses pedidos.

## Fora do Escopo

Desenvolver algoritmos para escolher o melhor caminho para as entregas de comida e garantir a entrega do alimento não são previstos para a solução que o projeto propõe.

## Definições, Acrônimos, e Abreviações

**Cliente** é quem realiza o pedido do alimento.

**Fornecedora** será aquela que irá realizar a entrega do alimento.

# CONTEXTO DO NEGÓCIO

## Relato do Problema

O cliente deseja pedir uma comida para sua refeição, contudo, devido a suas condições (independente de quais sejam, climáticas, pessoais, disponibilidade de tempo) precisa recebê-la sem se deslocar até o prestador. Além disso, o prestador pode não ter disponibilidade de oferecer serviços de entrega.

## Processo de Negócio

Este processo consiste no recebimento de pedidos de alimentos em que são necessárias avaliações a respeito da existência ou não no estabelecimento, além do pedido de pagamento deste produto requisitado. Além disso, possuem rotas alternativas de modo que o usuário possa optar por se alimentar no estabelecimento ou não. Desta forma, o fluxo principal consiste nas seguintes etapas:

Recebimento do pedido -> Verificação de o alimento existe no estabelecimento -> Análise se o estabelecimento entrega na região do cliente -> Geração do Pedido -> Recebimento do Pagamento -> Acompanhamento do pedido para o cliente.

## Riscos Identificados

A tabela a seguir apresenta os riscos identificados nessa etapa. Vale ressaltar que a relação de riscos do projeto deve ser revisitada durante todas as etapas do mesmo.

| **Descrição do Risco** | **Possível Solução** | **Probabilidade** | **Impacto** |
| --- | --- | --- | --- |
| Acesso à localização do cliente via GPS | Pedir para o cliente digitar o local de entrega ou solicitar ao cliente permitir (no caso do uso por celular) que o software tenha acesso a esse dado | alta | grave |
| Precisa-se do cardápio ou do pedido do cliente como string de entrada | Pedir ao cliente o acesso ao teclado ou ao dono do estabelecimento que disponibilize o cardápio | alta | médio |
| Alternativas para a entrega | Diante do dado do estabelecimento sobre ter serviço próprio de entrega, o cliente pode escolher alternativamente que a entrega seja feita pelo entregador do próprio estabelecimento (caso haja essa opção). | baixa | leve |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# STAKEHOLDERS DO PROJETO

## Papéis e Responsabilidades

| **Papel** | **Responsabilidades** |
| --- | --- |
| Cliente | - Avaliar a solução;  - Fornecer informações sobre o processo de negócio. |
| Product Owner | - Definir o escopo do sistema;  - Representar o cliente no desenvolvimento do projeto;  - Priorizar ou não as atividades em processo;  - Auxiliar na definição da sprint. |
| Scrum Master | - Guiar os desenvolvedores;  - Remover impedimentos ao longo do desenvolvimento;  - Fornecer status de atualização do projeto ao longo do desenvolvimento. |
| Desenvolvedores | - Utilizar as ferramentas sugeridas e outras ferramentas a fim de gerar a melhor solução de Software;  - Levantar os requisitos, escrever o Documento de Visão e participar das reuniões;  - Implementar os requisitos propostos. |

## Relação dos Stakeholders

|  |  |
| --- | --- |
| **Nome do Profissional** | **Papel** |
| Maués | Desenvolvedor; Product Owner |
| Toacy | Cliente |
| Dantas | Desenvolvedor; Scrum Master |
| Palmeira | Desenvolvedor |
| Guedes | Desenvolvedor |

# CARACTERÍSTICAS DO PRODUTO

**<Controle / Área de Pedidos>**

* **<Listagem de Alimentos>**: A área de pedidos deve ser capaz de mostrar ao cliente toda a listagem de alimentos disponível para o pedido tal como qual a fornecedora está disponibilizando este produto.
* **<Acompanhamento do Pedido>**: A área de pedidos deve ser capaz de mostrar ao cliente o status do seu último pedido caso ele tenha sido efetivado. No qual os estados serão (Realizado, Em produção, Em rota de entrega, Entregue).

**<Controle / Central de Pagamento>**

* **<Validar o método de pagamento>**: A central de pagamento deve ser capaz de verificar se o método de pagamento utilizado pelo cliente é capaz de pagar o valor do pedido ou não.
* **<Input de método de pagamento>**: A central de pagamento deve ser capaz de receber o input do método de pagamento do cliente.

**<Controle / Central de Validação>**

* **<Validar se o alimento existe>**: A central de validação deve garantir que, antes do cliente realizar o pedido, este alimento requisitado exista no cardápio da fornecedora.
* **<Validar se a fornecedora entrega>**: A central de validação deve garantir que, antes do cliente realizar o pedido, a fornecedora do alimento requisitado realize o processo de entrega.

**<Controle / Central da Fornecedora>**

* **<Input do cardápio>**: A central da fornecedora deve ser capaz de receber o input da fornecedora com o seu cardápio com os alimentos disponibilizados por esta.
* **<Input dos serviços>**: A central da fornecedora deve ser capaz de receber o input da fornecedora a fim de obter informações se esta irá entregar seus produtos ou promover descontos na compra local.

# RESTRIÇÕES

## Linguagens de Programação

A linguagem de programação utilizada será o JAVA e o sistema deverá rodar nos principais navegadores do mercado.

## Padrões de Desenvolvimento

A aplicação será desenvolvida de acordo com a arquitetura da plataforma AgileKIP.

## Requisitos Não Funcionais

Como o pagamento do pedido será realizado pela plataforma, é necessário que a segurança dessas informações seja garantida. Além disso, é necessário que haja a manutenção dos cardápios sempre que haja modificação.