

# Documento de Requisitos

## Dados Gerais:

Projeto: NOQUEUE

## Responsáveis:

Luiz Eduardo Cavaleiro GRR20148879

Julio Back GRR20154015

Laion Luis Ferreira GRR20158911

Duwan Bordinhao Bassani GRR20159185

Henry Jansen GRR20132081

## 1. Introdução

Este documento especifica os requisitos do software NOQUEUE, fornecendo aos desenvolvedores informações necessárias para o projeto e implementação, como a descrição dos requisitos funcionais, requisitos não funcionais e casos de uso.

## 2. Propósito e Descrição do Sistema

O aplicativo Noqueue consiste em ser um facilitador para fazer compras de supermercados, para evitar filas e economizar tempo. O aplicativo teria vários mercados de várias redes integrados a ele, similar aos aplicativos de delivery de comidas, com um mapa mostrando onde fica cada mercado.

O usuário digita o nome de um produto, o app pesquisa entre todos os mercados e ordena como o usuário preferir: maior para o menor, menor para o maior, qualidade, etc. O pagamento pelos carrinhos é sempre efetuado diretamente ao mercado, da maneira que o mercado disponibilizar (ao entregador, no local de retirada, em dinheiro, em cartão, etc).

O usuário pode escolher retirar no local ou mandar entregar em casa. A retirada da compra deve ocorrer em, no máximo, um dia, ou sua compra volta para as prateleiras. A pessoa pode fazer uma compra em um mercado só ou em vários. Quando a pessoa faz a compra, o aplicativo verifica os produtos presentes no carrinho e mostra quanto ficaria pelos mercados com os produtos mais baratos. Caso a pessoa veja que a economia é significativa, ela poderia escolher e ir

passando nos mercados pegando suas compras, visto que muitas pessoas não fazem compras de mês em um único mercado.

Se a pessoa mandar entregar em casa, o aplicativo calcula quanto ficará a compra e caso a forma de pagamento seja em dinheiro e quanto de troco será necessário. A responsabilidade pela entrega será do mercado.

A responsabilidade de integrar o sistema com o estoque do mercado seria dividida entre mercado e grupo do aplicativo, visando facilitar para os dois lados. O sistema disponibiliza uma interface onde o mercado poderá gerenciar o seu estoque, que pode ser implementada na forma de uma API.

Ações do usuário antes da compra devem ser reversíveis: escolha do mercado, escolha dos produtos, escolha das quantidades, opção de entrega, etc.

O custeio e monetização do sistema se dará por cobrança dos mercados de taxas/comissões proporcionais às compras realizadas pelo aplicativo.

### **3. Requisitos Funcionais**

#### **• Barra de busca e de mercados**

- Onde o usuário irá pesquisar o produto ou ver mercados disponíveis.
- Fácil de ser localizado na interface do aplicativo.
- Parâmetros de pesquisa, como ordenação por preço, qualidade, etc.
- Para facilitar o uso, no menu "mercados", teriam duas sessões, com a rede do mercado, e aí após entrar na rede x, o aplicativo mostra quais os mercados dessa rede estão cadastrados. Mostrando no mapa, para evitar que o cliente compre um produto de um mercado que esteja muito longe da onde ele mora. Algo como "Menu->Redes de mercados->Mercados dessa rede".

#### **• Corredores dos mercados**

- O aplicativo separa as categorias de produtos similarmente ao modo que são distribuídos nos corredores dos mercados, visando facilitar o uso do cliente. As categorias são definidas no aplicativo e seguidas pelo mercado quando do cadastro de seus produtos.

#### **• Comparação de preços**

- No carrinho, antes do pagamento, deve haver uma opção para comparar preços com outros mercados: "quanto ficaria sua compra no mercado X?"

#### **• Interface gráfica**

- Visando não utilizar tanto processamento e energia do celular, o aplicativo não terá interfaces gráficas muito “poluídas”, seria algo mais simples com “foto do produto, preço e descrição”.

## • Tutorial de uso

- Quando aberto pela primeira vez, o aplicativo mostra um tutorial de como usar o aplicativo.  
- Após o cadastro o usuário será apresentado por telas de “swipe” apresentando o usuário o fluxo de uso nos casos que ele esteja procurando um produto em si “Menu Inicial -> Produto -> Busca Produto -> Mercado que tem o produto -> Método de pagamento -> Método de entrega” ou por Mercado em si “Menu Inicial -> Mercado -> Opções de busca de mercado -> Produtos desse mercado -> Método de pagamento -> Método de entrega”.

## • Feedback

- Usuário pode avaliar os produtos e os mercados;  
- Mercados podem aferir “strikes” a usuários por problemas como efetuar compras e não buscá-las.  
- E eventuais disputas devem ser resolvidas por abertura de ticket, pela administração do aplicativo.

## • Georeferenciável

- Escolher e mostrar mercados próximos a localização do usuário.

## • Perfil de usuário

-Cadastrar, editar, visualizar

## • Perfil do Mercado

- Cadastrar, editar, visualizar

# 4. Requisitos Não Funcionais

- Flexível
- Usuário consegue realizar a tarefa por diferentes caminhos.
  - Acessível, Interativo
- A interface foi projetada para acessível e intuitiva para o usuário.
  - Seguro
- Restrições de acesso, privilégios.

- Usabilidade, Portabilidade
- A aplicação deve rodar em vários tipos de aplicativos: móveis, desktop, note.
- Comerciável

## 5. Cenário ilustrativo de uso

Dona Maria necessita de ingredientes para fazer uma receita para as suas visitas que chegarão daqui algumas horas. Como ela possui pouco tempo e já tem alguns dos ingredientes iniciais, decidiu começar a sua receita e fazer o pedido dos produtos que faltavam via aplicativo.

O aplicativo possui categorias em forma de prateleiras de escolha como num cenário real (um super mercado). A cozinheira foi colocando item a item em seu carrinho de compra. Como ela tem preferência pela marca x, o aplicativo pesquisou entre os mercados para encontrar o produto em estoque e o que possuir o menor preço. No final da compra solicitou que fosse realizada a entrega em sua casa, tendo em vista que seu tempo era reduzido.

## 6. Diagrama de casos de uso



## ***7. Descrição dos casos de uso e Fluxo de Eventos***

### **7.0.Tutorial de Uso**

**-Descrição:** Este caso de uso especifica a ação de aprender a usar o sistema, para o comprador vai ensinar como fazer compras facilmente, e para o mercado vai ensinar a parte de gerenciamento.

**-Atores:** Comprador, Mercado.

**-Pré-condições:** O ator deve iniciar o app.

**-Pós-condições:** O ator poderá fazer uso do app.

**-Fluxo Básico:**

- 1- O ator esta usando o aplicativo pela primeira vez.
- 2- Ao iniciar o aplicativo o sistema pergunta se o usuário necessita de tutorial.
- 3- Se o ator quiser o tutorial: começa tutorial.
- 4- Ao acabar o tutorial inicia o aplicativo normalmente.

O caso de uso se encerra.

**-Fluxo Alternativo A:**

- 1- No passo 3 do fluxo básico, se o ator não desejar o tutorial: o sistema pula o tutorial.
- 2- Inicia o aplicativo normalmente.

### **7.1.Login**

**-Descrição:** Este caso de uso especifica a ação de autenticação que um usuário executa no sistema, com objetivo de se conectar na aplicação.

**-Atores:** Comprador.

**-Pré-condições:** O ator já deve ter um cadastro ou se não deve se inscrever-se.

**-Pós-condições:** O ator poderá fazer uso do app.

**-Fluxo Básico:**

- 1- O ator decide se autenticar no sistema.
- 2- O sistema solicita as informações obrigatórias para o login.
- 3- O ator informa os dados.
- 4- O sistema valida os dados de autenticação.
- 5- O aplicativo vai para sua tela inicial.

O caso de uso se encerra.

**-Fluxo Alternativo A:**

- 1- No passo 4 do fluxo básico, caso haja algum erro nos dados informados pelo ator..
- 2- O sistema informa o erro ao ator.
- 3- O fluxo retorna ao passo 2 do fluxo básico.

### **7.2.Gerenciar Usuário**

**-Descrição:** Personalizar/editar informações pessoais.

**-Atores:** Comprador.

**-Pré-condições:** O ator deve ter estar logado.

**-Pós-condições:** O ator fica habilitado a gerenciar suas informações.

**-Fluxo Básico:**

1- O ator decide editar suas informações como endereço ou cartão de crédito.

2- O ator muda suas informações.

3- O sistema verifica essas informações.

4- Ator confirma as alterações.

5- Sistema informa que alterações foram feitas com sucesso.

O caso de uso se encerra.

**-Fluxo Alternativo A:**

1- No passo 3 do fluxo básico, se informações que precisão de verificações estiverem erradas o sistema informa sobre o erro.

2- Volta para o passo 2 do fluxo básico.

### 7.3.Ver Mapa

**-Descrição:** Este caso de uso especifica a ação de exibir um mapa mostrando mercados próximos ao usuário. Para esse caso de uso existe um extend de um outro caso de uso chamado obter localização que nada mais é de obter a localidade de certo mercado ou do próprio usuário.

**-Atores:** Comprador.

**-Pré-condições:** O ator tem que estar logado.

**-Pós-condições:** O ator tem a localização do que deseja.

**-Fluxo Básico:**

1- O ator deseja verificar mercados por perto.

2- Sistema mostra no mapa localização do ator e dos mercados.

3- Ator escolher mercado.

4- Sistema usa gps para fazer uma rota.

O caso de uso se encerra.

### 7.3.Avaliar Produto/Mercado

**-Descrição:** Este caso de uso especifica a ação de avaliação do produto que o usuário comprou ou do mercado, podendo avaliar o atendimento, higiene do local, etc.

**-Atores:** Comprador.

**-Pré-condições:** Ter comprado o produto de mercado x.

**-Pós-condições:** Ator pode ver avaliação dada e reavaliar.

**-Fluxo Básico:**

1- O ator deseja avaliar um produto ou um mercado.

2- Sistema informa produtos que ator comprou ou mercados que ator já fez compra, ou ator pesquisa pelo produto ou mercado.

3- Ator avalia e confirma.

4- Sistema informa conclusão.

O caso de uso se encerra.

**-Fluxo Alternativo A:**

- 1- Ator quer reavaliar produto.
- 2- Mesmo fluxo do fluxo básico.

**7.4.Pesquisar Produto/Mercado**

**-Descrição:** Especifica a ação de procurar algum produto específico, podendo utilizar também o corredor (categorias dos produtos), ou pesquisar um mercado ou redes de mercado específico.

**-Atores:** Comprador.

**-Pré-condições:** O ator deve estar logado e clicar na barra de pesquisa ou clicar na categoria e pesquisar.

**-Pós-condições:** O ator terá uma lista do produto pesquisado.

**-Fluxo Básico:**

- 1- O ator está procurando por algum produto ou mercado.
- 2- Ator escolhe corredor/categoria ou pesquisa diretamente na barra de busca.
- 3- Sistema faz a busca e mostra na tela.

O caso de uso se encerra.

**-Fluxo Alternativo A:**

- 1- No passo 3 do fluxo básico, o sistema não encontra produto ou mercado.
- 2- Sistema informa ao usuário que não encontrou.
- 3- Volta ao passo 2 do fluxo básico.

**7.5.Adicionar/Remover itens do carrinho**

**-Descrição:** Depois de escolher o produto o usuário pode adicionar o produto no carrinho, podendo remover o mesmo a qualquer momento. O comprador pode adicionar produtos de diferentes mercados no mesmo carrinho. Aqui o usuário pode escolher que o aplicativo procure os produtos com menor preço entre os mercados.

**-Atores:** Comprador.

**-Pré-condições:** O ator deve ter pesquisado ou clicado em algum produto.

**-Pós-condições:** O produto estará no carrinho para compra.

**-Fluxo Básico:**

- 1- O ator quer fazer compra, pesquisa pelos produtos.
- 2- Ator adiciona itens no carrinho.
- 3- Sistema calcula preço total que está no carrinho.
- 4- Ator pode visualizar todos os carrinhos e produtos dos carrinhos.
- 5- Ator pode remover itens dos carrinhos.
- 6- Ao remover sistema recalcula preço total.

O caso de uso se encerra.

**7.6.Comprar Itens do Carrinho**

**-Descrição:** Este caso especifica a ação de escolher o carrinho para fazer a compra e escolher o tipo de pagamento.

**-Atores:** Comprador.

**-Pré-condições:** O ator não pode estar com carrinho vazio, deve ter adicionado produtos no carrinho.

**-Pós-condições:** O ator poderá escolher forma de pagamento e entrega.

**-Fluxo Básico:**

- 1- O ator decide finalizar a compra de algum carrinho.
- 2- Sistema mostra todos os carrinhos.
- 3- Usuário escolhe carrinho.
- 4- Sistema pergunta forma de pagamento, cartão de crédito ou dinheiro caso usuário vai retirar no local.
- 5- Se for cartão sistema verifica dados do cartão.
- 6- Ator confirma.

O caso de uso se encerra.

**-Fluxo Alternativo A:**

- 1- No passo 4 do fluxo básico, ator pode querer trocar de carrinho
- 2- Volta para o passo 3 do fluxo básico.

**-Fluxo Alternativo B:**

- 1- No passo 5 do fluxo básico, se informações do cartão estiverem desatualizadas informa ao usuário.
- 2- Volta para o passo 4 do fluxo básico.

## 7.7. Entrega

**-Descrição:** Este caso de uso especifica a ação em que o comprador quer que as compras sejam entregues onde ele quiser, a responsabilidade da entrega é do mercado.

**-Atores:** Comprador, Entregador.

**-Pré-condições:** O ator selecionou carrinho.

**-Pós-condições:** Confirmar compra.

**-Fluxo Básico:**

- 1- O ator informa ao sistema que quer sua compra entregue em tal local.
- 2- Sistema pede confirmação.
- 3- Usuário confirma.
- 4- Ao confirmar compra, sistema informa ao entregador dados da entrega.
- 5- Mercado se responsabiliza pela entrega.

O caso de uso se encerra.

**-Fluxo Alternativo A:**

- 1- No passo 4 do fluxo básico, pode não haver entregador disponível.
- 2- Sistema informa que entrega pode atrasar por não ter entregador disponível.
- 3- Volta ao passo 1 do fluxo básico.

## 7.8. Retirar no(s) local(is)

**-Descrição:** Caso de uso onde o comprador escolheu que quer retirar a compra no mercado, ou nos mercados.

**-Atores:** Comprador, Mercado.

**-Pré-condições:** O ator selecionou carrinho.

**-Pós-condições:** Confirmar compra.



**-Fluxo Básico:**

- 1- O ator decide que quer ir pegar produtos nos locais dos mercados.
  - 2- Sistema pede confirmação do usuário.
  - 3- Usuário confirma.
  - 4- Sistema informa ao mercado.
  - 5- Mercado se responsabiliza por arrumar produtos e esperar pelo cliente para pegar os produtos.
- O caso de uso se encerra.

### **7.9.Confirmar Compra**

**-Descrição:** Ação de confirmar a compra do carrinho escolhido, aqui o comprador já deve ter escolhido o tipo de pagamento e se escolher que a mercadoria seja entregue para ele ou se vai retirar no local. Aqui o usuário pode avaliar a compra realizada.

**-Atores:** Comprador, Mercado, Empresa do cartão de crédito.

**-Pré-condições:** O ator deve ter selecionado carrinho, tipo de pagamento e escolhido se quer que mercadoria seja entregue ou vai retirar do local.

**-Pós-condições:** Esperar ou ir buscar mercadoria.

**-Fluxo Básico:**

- 1- O ator já escolheu forma de entrega e pagamento.
  - 2- Sistema pede confirmação da compra.
  - 3- Ator confirma.
  - 4- Sistema informa validação da compra.
  - 5- Ator pode avaliar sua compra.
- O caso de uso se encerra.

### **7.10.Gerenciar Produto**

**-Descrição:** Este caso de uso especifica a ação do mercado de gerenciar os estoques, quantidade, nome dos produtos.

**-Atores:** Mercado.

**-Pré-condições:** O ator deve estar logado.

**-Pós-condições:** O ator poderá gerenciar os produtos.

### **7.11.Gerenciar Mercado**

**-Descrição:** Este caso de uso especifica a ação de gerenciar o mercado, como melhorias, verificar funcionários, etc.

**-Atores:** Mercado.

**-Pré-condições:** O ator deve estar logado.

**-Pós-condições:** O ator poderá gerenciar a área restrita do mercado.

### **7.12.Gerenciar Franquia**

**-Descrição:** Este caso de uso especifica a ação do mercado de gerenciar a rede de mercados da franquia, como localização, se um mercado fechar precisa ser informado ao aplicativo, etc.

**-Atores:** Mercado.

**-Pré-condições:** O ator deve estar logado.

**-Pós-condições:** O ator poderá gerenciar a franquia de mercados.

### 7.13. Selecionar tipo de Pagamento

**-Descrição:** Este caso de uso especifica a ação do escolher como vai pagar pela compra, que pode ser cartão de crédito, paypal ou em dinheiro se escolher retirar no local.

**-Atores:** Comprador, Empresa do cartão de crédito.

**-Pré-condições:** O ator deve ter escolhido carrinho que queira finalizar a compra.

**-Pós-condições:** O ator vai poder confirmar a compra.

**-Fluxo Básico:**

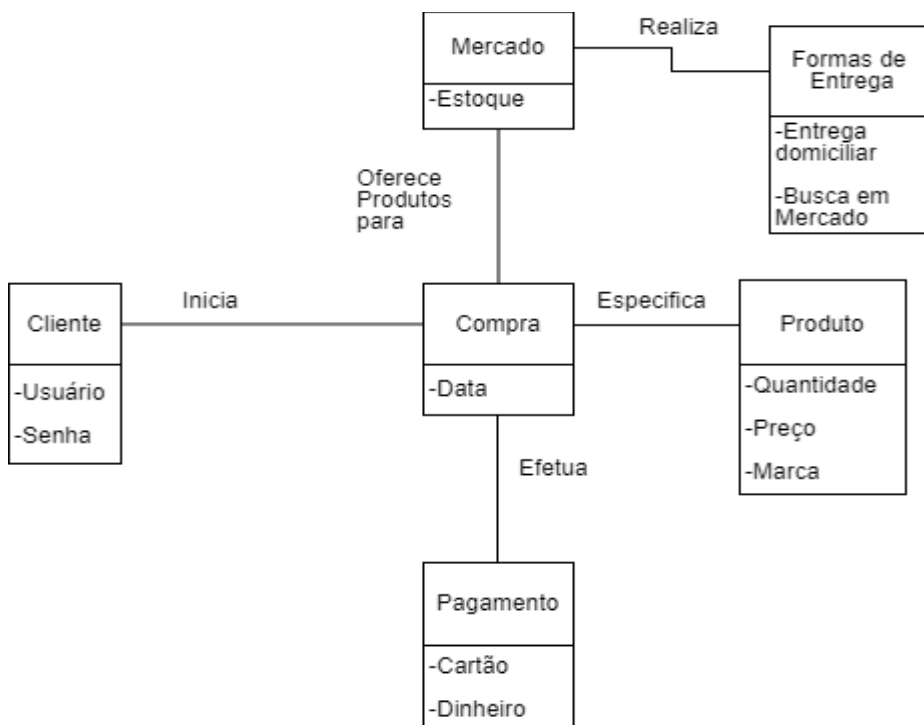
- 1- O ator escolhe carrinho que queira finalizar a compra.
- 2- Sistema informa qual tipo de pagamento ator pode escolher.
- 3- Ator escolhe tipo de pagamento.
- 4- Sistema verifica informações e pede confirmação.
- 5- Ator verifica informações e confirma.

O caso de uso se encerra.

**-Fluxo Alternativo A:**

- 1- No passo 4 do fluxo básico, pode acontecer algum erro da empresa do cartão ou algum dado desatualizado.
- 2- Sistema informa ao usuário.
- 3- Volta ao passo 3 do fluxo básico.

## 8. Diagrama Conceitual



## 9. Prototipação

1- O protótipo ajuda a dar uma clareza maior de como será o aplicativo.

- Link para o protótipo

<https://marvelapp.com/5086j88/screen/32732596>

2- Telas do protótipo:

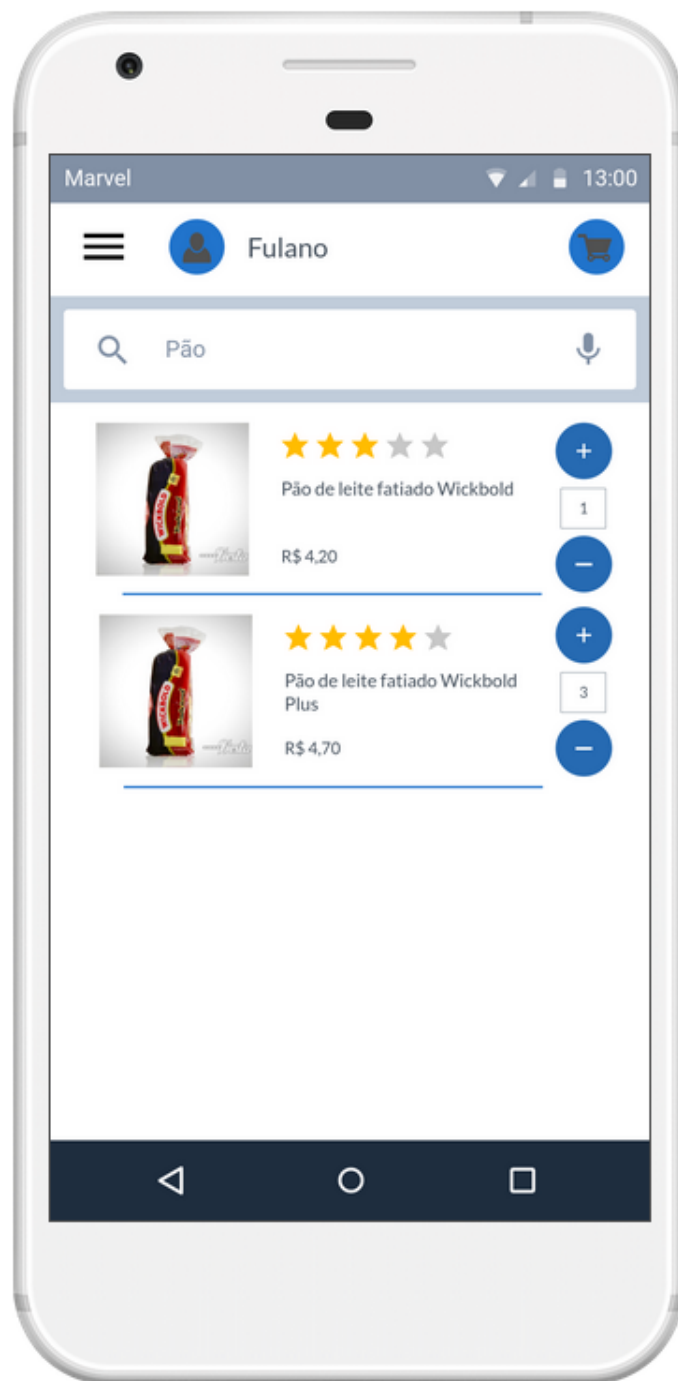
**Tela inicial:**



## Tela de categorias:



## Tela de produtos:



## Tela de Mercados:

