



PONTIFÍCIA UNIVERSIDADE CATÓLICA DE CAMPINAS  
Centro de Ciências Exatas, Ambientais e de Tecnologias  
Faculdade de Engenharia da Computação

## **Projeto de Desenvolvimento de Sistema de Controle de Estoque**

Gabriel Gonçalves Mattos Santini	RA: 18189084
Gustavo Melo Cacao	RA: 19091057
Gustavo Campos Dias	RA: 19075712
João Lucas Fernandes da Silva	RA: 19088582
Lucas Rodrigues São João Miguel	RA: 19029297

**Campinas  
Maio 2022**



## Sumário

1. Introdução.....	03
2. Apresentação do Tema.....	03
3. Definição do Problema.....	04
4. Projeto de Software.....	06
5. Análise e Teste dos Dados.....	12
6. Conclusão.....	13
7. Referência Bibliográfica.....	13

## 1. Introdução

Este é um projeto proposto pelo professor Leandro Alonso Xastre na matéria de nome “Redes de Computadores A” do curso de Engenharia de Computação da PUC Campinas.

O grupo é composto por 5 integrantes, onde todos estão matriculados no curso e nas matérias citadas acima.

## 2. Apresentação do Tema

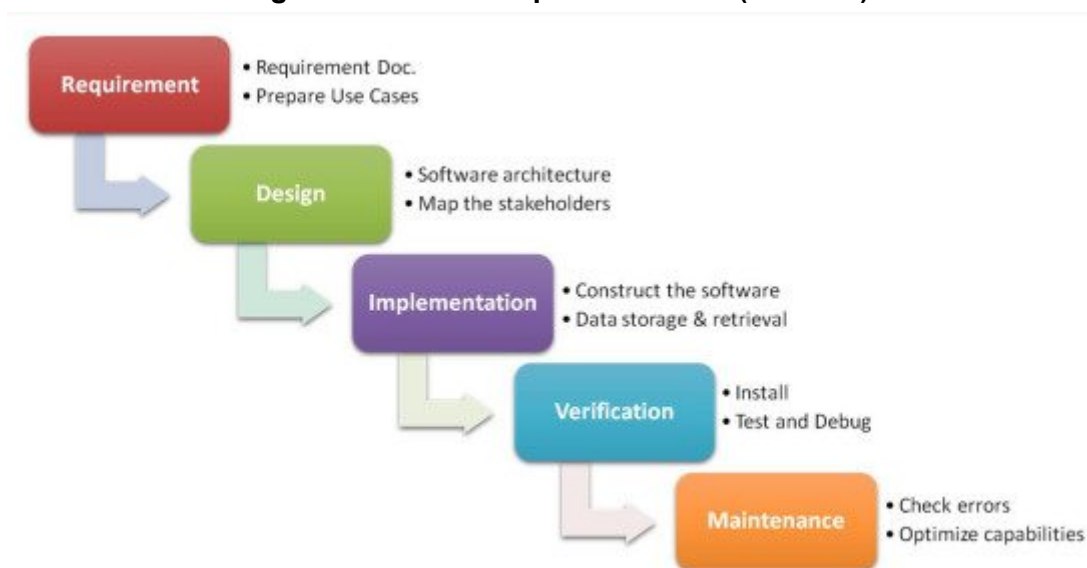
O gerente de um supermercado encontra problemas em saber a quantidade exata de um determinado produto em suas gôndolas e prateleiras, assim não sabendo quando o mesmo deve ser reabastecido. Por esse motivo foi requisitada uma automação nesse processo.

O sistema deve controlar o almoxarifado, o estoque interno, as gôndolas e prateleiras e o caixa. O gerente ainda acrescenta que tem à disposição celulares no almoxarifado e computadores tanto no estoque interno como nos caixas.

Para o desenvolvimento do projeto o Modelo de Ciclo de Vida escolhido foi o Modelo Sequencial Linear, também conhecido como Modelo Cascata.

Um Modelo de Ciclo de Vida descreve as etapas do processo de desenvolvimento de sistemas e as atividades a serem realizadas em cada etapa. O Modelo Cascata (Figura 1) é o modelo mais antigo e o mais usado na engenharia de softwares. Este deve ser usado em situações das quais os requisitos do sistema estão bem definidos e segue uma abordagem sistemática e sequencial.

**Figura 1: Modelo Sequencial Linear(Cascata)**



Fonte: [casadaconsultoria.com.br/modelo-cascata/](http://casadaconsultoria.com.br/modelo-cascata/)

### 3. Definição do problema

Usando como referência a documentação original do projeto e as novas informações passadas posteriormente, o grupo, após algumas reuniões discutindo sobre o assunto, definiu diversos requisitos para o funcionamento do sistema. Estes foram definidos em Requisitos Funcionais e Requisitos Não Funcionais para a melhor organização.

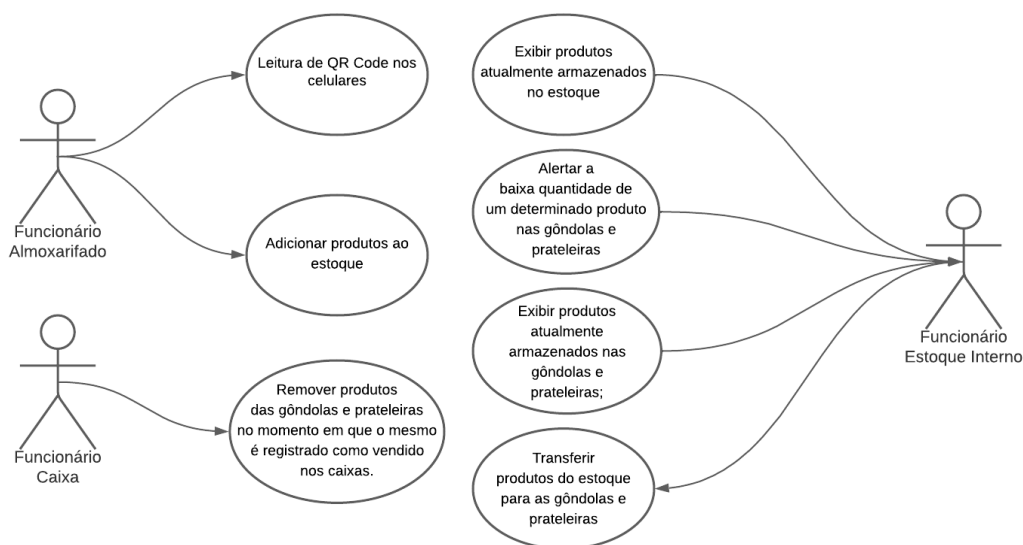
#### 3.1 Requisitos

- **Requisitos Funcionais**

- Leitura de QR Code nos celulares;
- Adicionar produtos ao estoque;
- Exibir produtos atualmente armazenados no estoque;
- Transferir produtos do estoque para as gôndolas e prateleiras;
- Exibir produtos atualmente armazenados nas gôndolas e prateleiras;
- Alertar a baixa quantidade de um determinado produto nas gôndolas e prateleiras;
- Remover produtos das gôndolas e prateleiras no momento em que o mesmo é registrado como vendido nos caixas.

Para uma ilustração melhor dos requisitos funcionais, garantindo assim um entendimento melhor do funcionamento do sistema, foi usada a Modelagem de Caso e Uso da UML(Figura 2).

**Figura 2: Diagrama de Caso e Uso da UML**



Fonte: Autor



- **Requisitos Não Funcionais:**

- Pelo menos um celular Android no Almoxarifado;
- Pelo menos um computador no estoque interno;
- Pelo menos um computador em cada caixa;
- Conexão com a rede em cada um dos computadores que pode ser tanto por cabo quanto Wi-Fi;
- Conexão com a rede em todos os celulares via Wi-Fi;
- Aparelho roteador com conexão à rede.

### **3.2 Regra de Negócios**

Algumas Regras de Negócio também foram consideradas na hora da elaboração do sistema.

Essas regras são normas implementadas no negócio do cliente em questão. Elas sempre variam de negócio a negócio e por isso devem ser analisadas e levadas em consideração no desenvolvimento do sistema para que o mesmo se molde aos termos do cliente e atenda seu propósito com exímio.

- **Regras de Negócio:**

- O estoque interno tem uma quantidade mínima que um determinado produto pode ter;
- As gôndolas e prateleiras tem uma quantidade mínima que um determinado produto pode ter;
- As gôndolas e prateleiras tem uma quantidade máxima que um determinado produto pode ter;
- Os produtos recebidos são identificados exclusivamente por um QR Code, logo o sistema deve conseguir ler e interpretar um;
- Todos os produtos já entregues uma vez e todos que ainda serão entregues já estão cadastrados no Banco de Dados.

### **3.3 Restrições Iniciais**

Foram analisadas, também, possíveis restrições iniciais que poderiam ser consideradas no desenvolvimento. Essas restrições são aquelas que regem não só o negócio em questão, mas sim a comunidade da qual o mesmo faz parte, logo devem ser seguidas mesmo que não seja explicitamente pedido pelo cliente. Em outras palavras, são as restrições legais do local em questão.

Após o grupo ter debatido sobre o assunto foi definido que não há nenhuma regra inicial que deva ser considerada na hora do desenvolvimento do sistema.



## 4. Projeto de Software

### 4.1 Linguagens Utilizadas

Antes do início do projeto, o grupo discutiu quais linguagens seriam usadas para a produção dos algoritmos. Para o website foi utilizado HTML, CSS e JS, pelo fato do grupo ter uma base para trabalhar nesse ambiente. No aplicativo, foi utilizado o Flutter, por ser um conhecimento geral do grupo. Para a mensageria, o escolhido foi o RabbitMQ. Por fim, no Banco de dados relacional foi usado MySQL e no Não relacional, o MongoDB, por ser mais prático.

### 4.2 Arquitetura do Ambiente

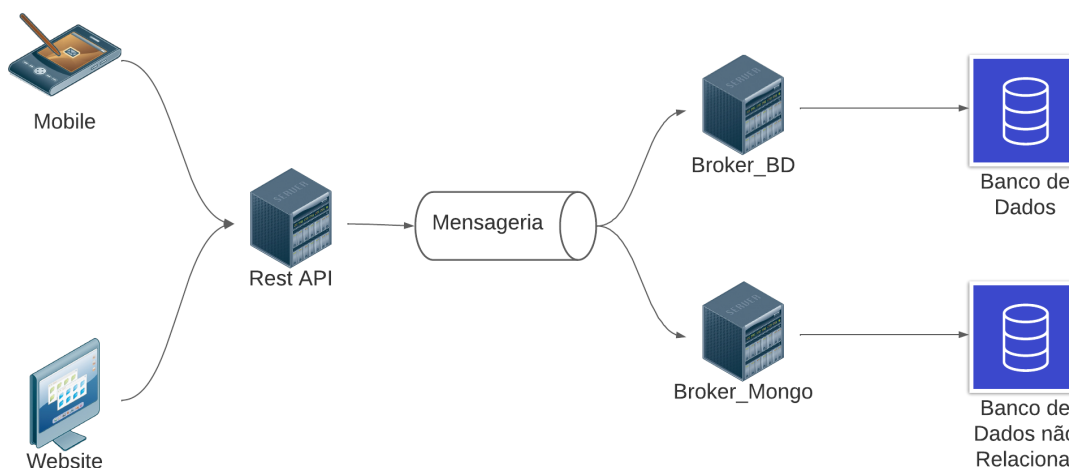
Para o funcionamento do sistema serão necessárias diversas condições e dispositivos. Esse conjunto de requisições é chamado de arquitetura de ambiente. Os itens necessários são:

- **Pelo menos um celular no almoxarifado:** Esse será responsável por escanear o código QR dos produtos que estão chegando para abastecer o estoque. Nele terá instalado um APK que irá escanear o código e já colher todas as informações do produto em questão, após isso o usuário irá digitar a quantidade deste produto que será adicionada no estoque e confirmar, assim o sistema irá automaticamente adicionar este valor no banco de dados do sistema.
- **Pelo menos um computador no estoque interno:** Nesta máquina será acessado o website do sistema, na qual o usuário poderá ver todos os produtos presentes no estoque e nas gôndolas e prateleiras. Nele, também, será possível sinalizar a transferência feita de produtos do estoque para as prateleiras.
- **Pelo menos um computador em cada caixa:** Nestes computadores serão registradas as vendas, assim retirando os produtos em questão das gôndolas e prateleiras.
- **Rede local:** O mercado deverá ter uma rede local, a qual deverá transmitir sinal tanto pelo cabo quanto por Wi-Fi (para os celulares). Isso é essencial, pois toda a comunicação entre os micros serviços do sistema será feita pela rede. Dessa forma, logicamente, todos os aparelhos que compõem o sistema devem estar conectados a esta mesma rede local para o funcionamento do sistema.

### 4.3 Funcionamento do Sistema

O sistema é composto por alguns micro serviços que cumprem cada um o seu papel individual. Para a melhor visualização, foi montado um Fluxograma (Figura 3) com o escopo completo do projeto e um para cada micro serviço individualmente.

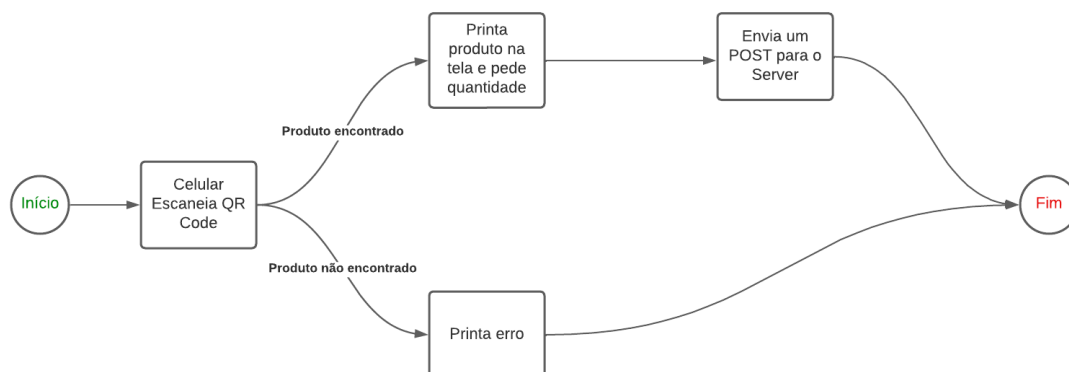
**Figura 3: Fluxograma do sistema completo**



Fonte: Autor

Seguindo essa lógica de funcionamento do sistema, a primeira parte a ser abordada será o APP Mobile. Esse terá a função de escanear um QR Code dos produtos novos e, após solicitar a coerência dos dados, adicionar uma quantidade que será definida pelo usuário caso não haja problema na leitura do QR Code. Segue Fluxograma (Figura 4).

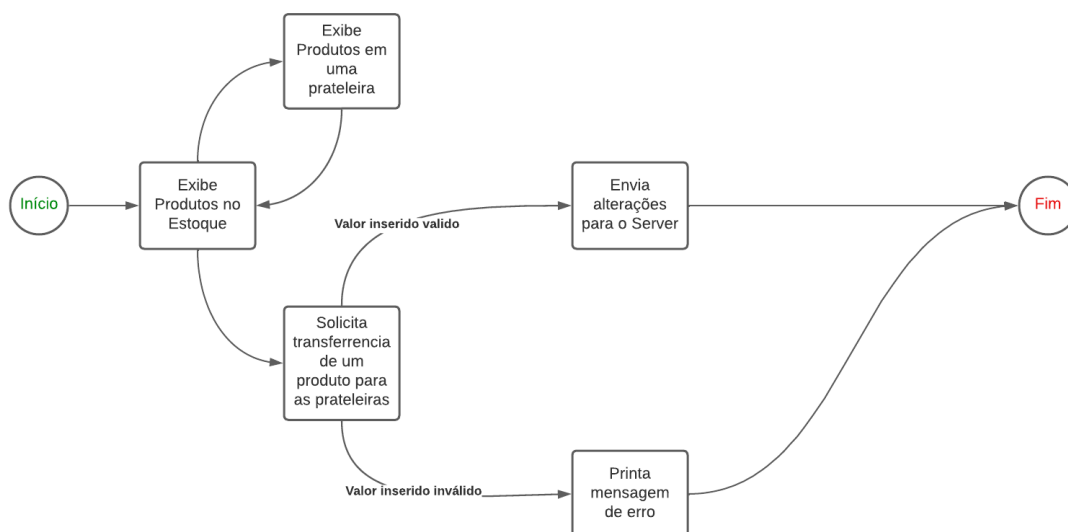
**Figura 4: Fluxograma do APP Mobile**



Fonte: Autor

O próximo a ser abordado será o website. Nele poderá ser conferido os produtos que estão no estoque interno e os que estão nas gôndolas e prateleiras além de transferir os produtos do estoque para as prateleiras. Segue seu Fluxograma (Figura 5).

**Figura 5: Fluxograma do website**



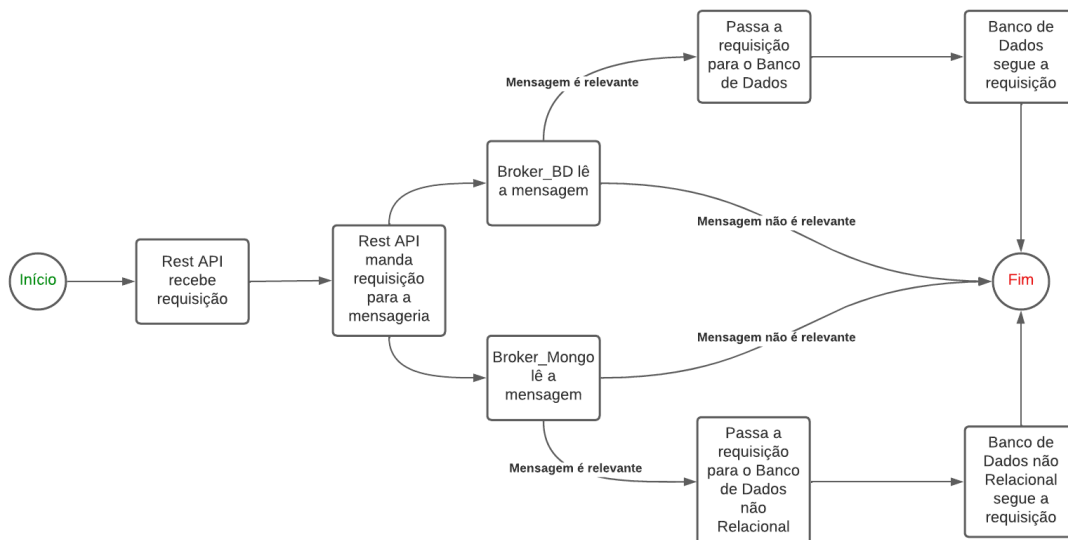
Fonte: Autor

O funcionamento dos diversos módulos são dependentes um do outro para o entendimento completo de seus funcionamentos, portanto serão explicados juntos.

Após o APP Mobile ou o website fazer uma requisição, essa vai para o Server (Rest API) que enviará uma mensagem para os diversos módulos usando um sistema de mensageria. Estas mensagens são interceptadas pelo Agente que se comunica com o Banco de Dados Relacional (Broker\_BD) e pelo Agente que se comunica com o Banco de Dados Não Relacional (Broker\_Mongo) e esses por sua vez, caso a mensagem seja de interesse ao módulo em questão, passam as requisições para os respectivos Bancos de Dados, que caso necessário retornam uma mensagem fazendo o caminho inverso. Segue o Fluxograma para melhor entendimento (Figura 6).



**Figura 6: Fluxograma dos Servidores e Bancos de Dados**

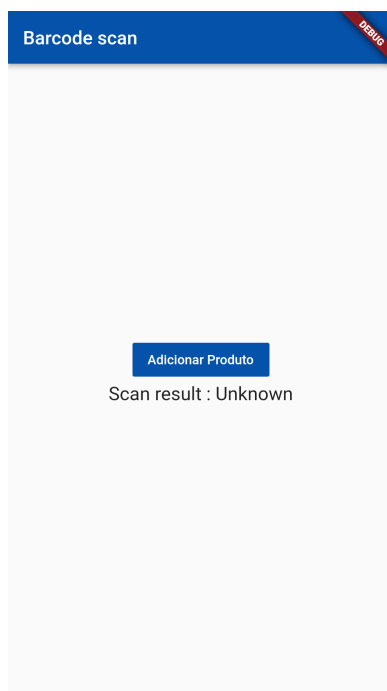


Fonte: Autor

## 4.4 Interface do Sistema

Esta tela é referente ao APP Mobile (Figura 7). Mais precisamente caso fosse escanear um QR Code não cadastrado no Banco de Dados.

**Figura 7: Tela de Produto escaneado não identificado**



Fonte: Autor



Caso o QR Code escaneado seja identificado, uma tela com uma imagem do produto (caso o produto a tenha) e com a opção de escolher a quantidade de produtos que será apresentada (Figura 8).

**Figura 8: Tela de Adicionar Produtos ao Estoque**

← Adicionar: Melão Rei - Kilo

Melão Rei - Kilo

Quanto a adicionar  
24

Adicionar Cancelar

Fonte: Autor

No website será apresentado os produtos que estão nas prateleiras, assim como as outras informações referentes a esses (Figura 9).

**Figura 9: Tela de produtos nas prateleiras**

Prateleiras							Caixa	Prateleiras	Source
ID da Prateleira	Quantidade Total	Quantidade Mínima	Quantidade Máxima	Nome do Produto	ID do Produto	Preço do Produto			
1	22	10	40	Azeite Andorinha Extra Virgem 500mL	2345	25.9			
2	26	10	40	Melão Rei - Kilo	5678	14.98			
3	37	10	40	Coca-Cola Original 2L	9012	8.95			

Adicionar à Prateleira



**PUC**  
CAMPINAS

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE CAMPINAS**  
**Centro de Ciências Exatas, Ambientais e de Tecnologias**  
**Faculdade de Engenharia da Computação**

Fonte: Autor

Também no website será possível ver os produtos que estão no estoque e transferi-los para as prateleiras (Figura 10).

**Figura 10: Tela de produtos no estoque**

ID do Produto	Nome do Produto	Preço do Produto	Quantidade Total	Quantidade Mínima	Adicionar para Prateleira
2345	Azeite Andorinha Extra Virgem 500mL	R\$ 25.90	7	5	+
5678	Melão Rei - Kilo	R\$ 14.98	45	8	+
9012	Coca-Cola Original 2L	R\$ 8.95	78	10	+

Fonte: Autor

Por último, nos caixas são registrados os produtos que estão sendo comprados, assim como suas informações, e após isso esses serão retirados das prateleiras(Figura 11).

**Figura 11: Tela do caixa**

ID do Produto	Nome do Produto	Quantidade	Preço do Produto	Preço Total
9012	Coca-Cola Original 2L	1	R\$ 8.95	R\$ 66.66
5678	Melão Rei - Kilo	1	R\$ 14.98	R\$ 45.66
2345	Azeite Andorinha Extra Virgem 500mL	3	R\$ 25.90	R\$ 63.79

Professor Leandro Alonso Xastre da Pontifícia Universidade Católica de Campinas ganha competição de sistemas com mensageria.

Adicionar ao Carrinho

**R\$ 176.11**

**Finalizar Venda**

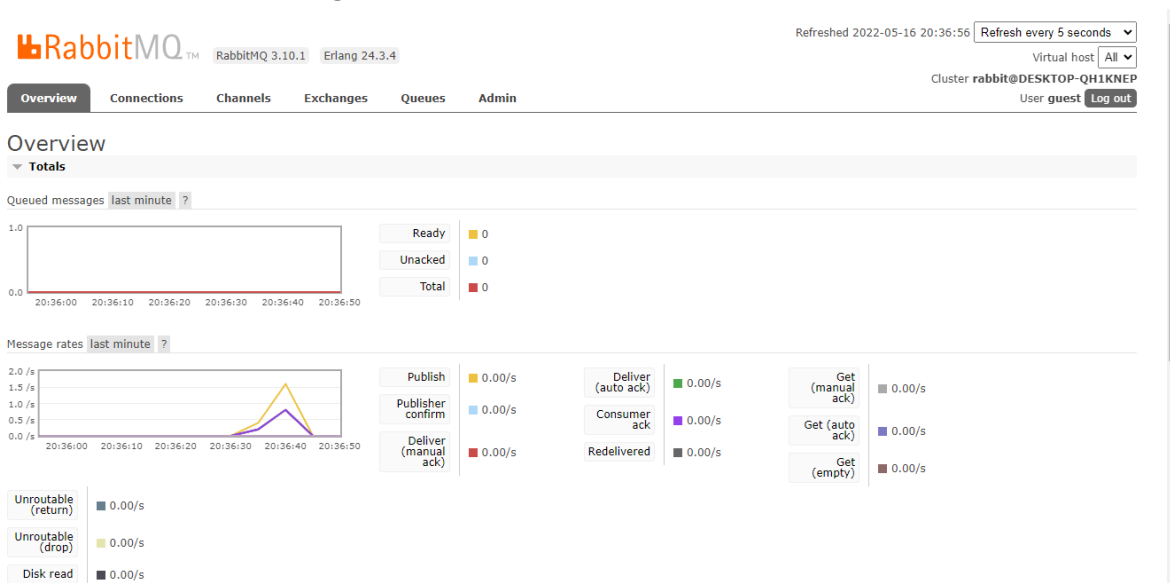
Fonte: Autor



## 5. Análise e Teste dos Dados

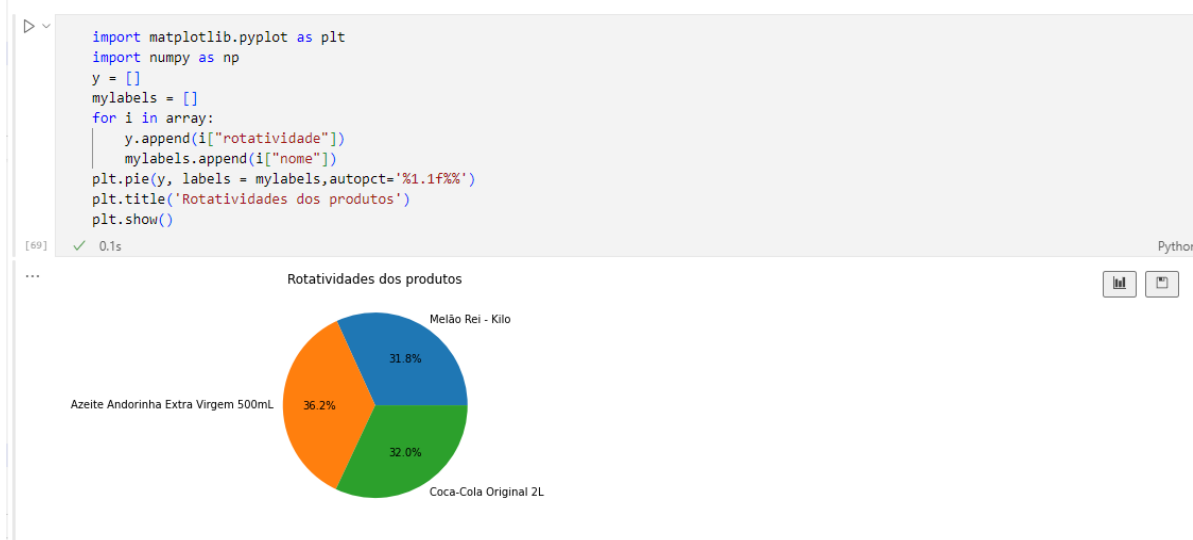
Para fazer uma análise mais precisa dos dados, foi gerado um gráfico, a partir do RabbitMQ, que mostra o fluxo de dados do sistema, como vista na Figura 12. E, também, é produzido gráficos em formatos de pizza, mostrando as quantidades alteradas durante o uso do sistema, ou seja, informações de qual produto foi mais vendido e a quantidade de produtos presentes nos estoques e prateleiras. Um exemplo da visualização desta funcionalidade está presente na Figura 12 e 13 a seguir.

Figura 12: Análise e teste no RabbitMQ



Fonte: Autor

Figura 13: Gráfico das quantidades de produtos



Fonte: Autor



## 6. Conclusão

Em suma, ao finalizar o projeto deu para ter um conhecimento base de como uma rede assíncrona funciona. Dentro do que foi pedido, houve um entendimento das funcionalidades e requisitos de uma mensageria, na qual tornou-se um pequeno desafio por conta do grupo não possuir conhecimento quando teve o início do projeto, as conexões entre os servidores, sendo eles o Rest\_API, o Broker\_DB e Broker\_Mongo, ambos brokers sendo agentes que interpreta a mensageria, e os Bancos de dados Relacional e Não Relacional.

Dessa forma, foi uma boa oportunidade de ter uma noção de como todas essas áreas conseguem se comunicar entre elas. Assim como, aprender a fazer essa comunicação funcionar de uma forma simples e ágil sem que haja perda de dados.

## 7. Referência Bibliográfica

- <https://levelup.gitconnected.com/introduction-to-rabbitmq-with-nodejs-61e2aec0c52c>
- <https://developer.mozilla.org/en-US/docs/Web/API/History/go>
- <https://developer.mozilla.org/pt-BR/docs/Web/API/Document/createElement>
- <https://developer.mozilla.org/pt-BR/docs/Web/HTML/Element/form>
- <https://docs.flutter.dev>