

UNIVERSIDADE DE BRASÍLIA – UNB

FACULDADE GAMA

ELETRÔNICA EMBARCADA 120961- TURMA A 2019.2

PONTO DE CONTROLE I

PROJETO: SISTEMA AUXILIAR DE INSERÇÃO DE PRODUTOS NO ESTOQUE PARA LOJAS VIRTUAIS.

Alexandre Torres Kryonidis
Matrícula: 13/0099767

Engenharia de Software

Faculdade Gama - Universidade de Brasília
Área Especial de Indústria Projeção A Brasília,
CEP: 72.444-240

email: alexandreky@gmail.com

Misael de Souza Andrade
Matrícula: 16/0015669

Engenharia Eletrônica

Faculdade Gama - Universidade de Brasília
Área Especial de Indústria Projeção A Brasília,
CEP: 72.444-240

email: misas.andrade@gmail.com

RESUMO

Este relatório tem como objetivo apresentar a proposta do nosso projeto da disciplina de Sistemas Operacionais Embarcados, um dispositivo com sistema embarcado para controle e manutenção de estoque de lojas. Para tanto usaremos como elemento principal do produto a RASPBERRY PI, da *Fundação Raspberry Pi*, uma série de computadores de placa única do tamanho reduzido. O projeto visa alcançar comerciantes dispostos a ter um dispositivo de baixo custo e eficiente, trazendo segurança no controle de dados e facilidade de uso.

1. JUSTIFICATIVA

Cada vez mais percebe-se a importância e o crescimento de lojas virtuais. Porém, um dos maiores gargalos da operação de um mercado online é o controle de estoque [7].

Ao analisar o funcionamento de um e-commerce, percebe-se que após o cliente realizar uma compra o sistema gera uma ordem em que estão discriminados todos os produtos vendidos. Portanto, quando o cliente confirma o pagamento, o sistema decresce a quantidade de cada produto, automaticamente, do banco de dados.

Com isso, é possível notar que o sistema de saída de produtos do estoque de uma loja online já é algo automatizado e não tem muito o que melhorar.

O processo de incrementar o estoque dos produtos por outro lado, é uma atividade manual que é feita normalmente no mesmo sistema web utilizado para vendas. O processo de atualizar informações dos produtos como a quantidade em estoque é algo bastante simples, porém não é prático, visto que é necessário que o usuário faça uma pesquisa com o nome de cada produto ou o seu código associado.

Daí surge a necessidade de utilizar um leitor de código de barras para incrementar a quantidade de determinado produto no estoque. O problema de utilizar código de barras surge pelo fato da Aplicação Web da loja virtual estar na nuvem. Não temos acesso físico ao servidor remoto onde a aplicação está rodando. É necessário, portanto, uma aplicação local rodando em um hardware local capaz de enviar esses dados para a nuvem.

Uma alternativa seria sempre que escanear um produto incrementar o estoque, isso poderia ser facilmente realizado utilizando um microcontrolador conectado a um scanner, porém há situações em que isso não é viável. Por exemplo, para inserir cem unidades de um mesmo produto seria necessário escaneá-lo cem vezes, dessa forma percebe-se a

necessidade de um teclado. Caso o produto não exista no banco de dados é necessário indicar isso para o usuário, daí pensa-se na ideia de um *display*, que seria atualizado com uma lista de produtos e suas respectivas quantidades. Percebe-se que um dispositivo teria que realizar diversas funcionalidades simultaneamente: atualizar display, ler código de barras, ler entradas do teclado, fazer requisições para o servidor, emitir sons. Para isso, é vantajoso possuir um sistema operacional.

2. OBJETIVOS

A. Objetivos gerais

Projetar um dispositivo embarcado em uma Raspberry Pi [1], que auxilie no controle de estoque de lojas online, de modo que facilite o usuário a repor o estoque de sua loja.

Deseja-se construir um sistema desacoplado do sistema principal (aplicação web de loja online), de tal forma que a loja virtual não precise desse sistema auxiliar, e esse sistema auxiliar funcione com qualquer aplicação web que implementar os endpoints necessários seguindo as interfaces definidas.

B. Objetivos específicos

Neste sistema auxiliar o usuário deve ser a:

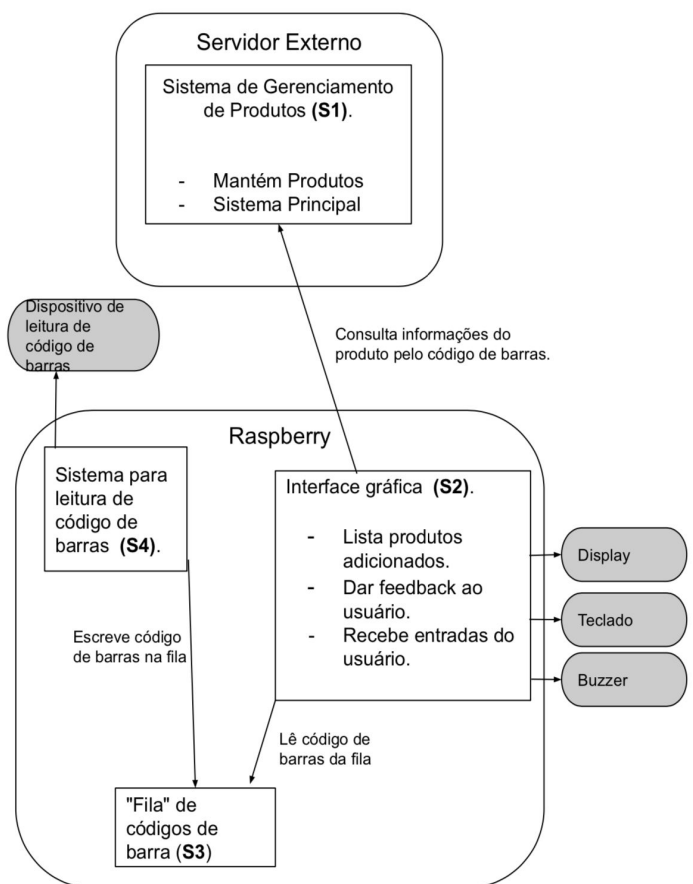
- Inserir produtos na base de dados do estoque por meio de um sistema externo baseado em código de barras;
- Consultar informações dos produtos no servidor web onde são armazenados os produtos;
- Por meio de leitura de código de barras identificar os produtos que não existem na loja virtual e avisar ao usuário, para que ele realize o cadastro;
- Interagir com o sistema por meio dos periféricos de entradas (leitor de código de barras e teclado) e de saída (monitor);
- Ser um sistema seguro do ponto de vista de autenticação dos usuários de tal forma que cada usuário deve ser autenticado para

poder realizar ações de escrita no banco de dados.

- Ter um dispositivo de baixo custo e extremamente confiável.

3. VISÃO GERAL DO PROJETO

O sistema geral pode ser decomposto em subsistemas. A partir da imagem a seguir é possível observar os subsistemas que compõem o projeto (S1, S2, S3, S4).



A. Subsistemas

• (S1)

Este é o sistema principal, é uma aplicação web de vendas que possui controle de estoque. Consiste de uma aplicação em que o usuário pode cadastrar, remover, atualizar e visualizar produtos.

Este é um sistema completo por si só, ou seja, não há necessidade de mais nada para realizar o controle de estoque. Apesar disso, esse sistema é remoto e roda no navegador do usuário, portanto, não permite o uso de um scanner de código de barras. Com isso, o processo atualizar a quantidade de produtos no estoque deve ser feito via navegador e, portanto, o usuário deve digitar o nome ou código do produto, tornando esse processo demorado. Na disciplina será criado um MVP funcional desse sistema para podermos testar os demais subsistemas que rodarão na Raspberry.

- **(S2)**

Esse é o subsistema responsável por incrementar o estoque. Ele lê os códigos de barras do subsistema (S3) e consulta o sistema (S1) para obter as informações do produto. Se o produto for encontrado, ele atualiza a interface gráfica inserindo este produto em uma lista na cor verde e faz um som específico usando o Buzzer. Se o produto não estiver cadastrado, ele o mostra em vermelho e faz um som diferente. Esses produtos em vermelho serão ignorados, o usuário precisa primeiramente inseri-los no Sistema (S1).

Além disso, permite a interação do usuário, como remover um item da lista, adicionar código de barras manualmente. O incremento do estoque somente será realizado após o comando de confirmação do usuário, desta forma, todos os produtos referentes aos itens em verde da lista serão incrementados.

- **(S3)**

Consiste apenas uma fila de código de barras que funciona como um intermediador entre (S2) e (S4). Dessa forma, forçamos uma interface comum entre (S2) e (S4) e esses dois sistemas podem ser substituídos por qualquer um outro que siga a interface definida previamente. Esta arquitetura permite inclusive que sejam utilizados mais de um subsistema (S2) escrevendo na fila de códigos de barras.

- **(S4)**

Esse subsistema é Daemon que rodará na Raspberry PI recebendo o código de barras de um dispositivo externo e inserindo ele no sistema (S3).

B. [Comunicação entre os subsistemas \(Ponto de Controle 2...\)](#)

4. REQUISITOS DO PROJETO

As especificações do projeto a ser desenvolvido serão descritas neste tópico, serão apresentados os requisitos que ele deverá atender.

A. Requisitos globais

- O sistema deve facilitar a inserção de produtos em sistemas de lojas virtuais.
- O usuário deve poder inserir produtos no sistema usando código de barras.
- O usuário deve poder escanear ou digitar manualmente o código de barras.
- O usuário deve ser capaz de desfazer uma inserção feita incorretamente.
- O usuário deve ser capaz de alterar a quantidade de determinado produto a ser inserido.
- O sistema deve autenticar o usuário para realizar ações que envolvem escrita no banco de dados.
- O sistema auxiliar de controle de estoque deve ser desacoplado da loja virtual.

B. Requisitos do subsistemas

- **(S1)**

- Deve possuir endpoints para:
 - Consulta de produto pelo código de barras.
 - Receber uma lista de produtos e incrementar os estes produtos no estoque.
- Deve exigir a autenticação do cliente para inserir os produtos.

- Deve armazenar os dados em um Banco de Dados para garantir a persistência e integridade dos mesmos.
 - Deve "Manter Produtos":
 - Criar
 - Visualizar
 - Atualizar
 - Deletar
 - Cada produto deve ser relacionado a um ou mais códigos de barra
 - Deve "Manter Usuários":
 - Cada usuário administrador terá a sua senha para inserir produtos no estoque.
 - Deve manter o histórico de inserções, salvando horário, a lista de produtos e suas quantidades, e o usuário que realizou a operação.
- S2)
 - Deve permitir que o usuário defina o IP e Porta dos demais subsistemas.
 - Deve permitir a inserção manual de código de barras.
 - Dado um código de barras, deve gerar uma requisição para o sistema (S1) consultando as informações do produto.
 - Ao enviar a lista de produtos para o sistema (S1) com o objetivo de incrementar o estoque, o usuário deve digitar sua senha de autenticação.
 - Deve ler a fila (S3) para verificar se tem um novo produto a ser consultado.
 - S3)
 - Deve funcionar como uma fila de código de barras.
 - Outros sistemas podem inserir o código de barras.
 - Um sistema "consumidor" deve ser capaz de obter os códigos de barras para processamento.
 - S4)

- Deve ser capaz de receber sinais externos que representam o código de barras e inseri-los em (S3).
- Pode ou não processar os dados recebidos de dispositivos externos.

4. BENEFÍCIOS

Através do uso do dispositivo proposto e de suas ferramentas dispostas, o usuário terá uma gestão moderna, eficaz e barata do seu estoque. Haja vista os que o problema de controle de estoque é extremamente prejudicial para as empresas [4], o presente projeto traz um alto custo-benefício, onde sendo de baixo custo pelo uso da Raspberry PI e por ser um sistema dedicado, irá suprir as necessidades do cliente.

Há inúmeros benefícios em se automatizar ao máximo o controle de estoque de uma empresa. No caso de lojas online, o controle de saída de mercadorias é bastante simples, porém a reposição de estoque é algo mais demorado por ser uma atividade predominantemente manual.

Porém, com esse projeto tentamos diminuir o tempo de execução dessa atividade manual. A manipulação de tais produtos por meio de códigos de barra, do teclado e monitor gráfico traz comodidade para o usuário, que não precisa manualmente editar o produto em aplicação web, ele poderá fazer isso por uma aplicação externa que tem como benefício a utilização de um leitor de código de barras.

5. REFERÊNCIAS

- 1.RASPBERRYPI.ORG. **What is a Raspberry Pi?** Disponível em: <<https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/>>. Acesso em: 28 ago. 2019.
- 2.SEBRAE NACIONAL. **Entenda a importância da gestão de estoque.** Disponível em: <<http://www.sebrae.com.br/sites/PortalSebrae/artigos/como-melhorar-a-gestao-de-produtos-no-varejo,6ed4524704bdf510VgnVCM1000004c00210aRCRD>>. Acesso em: 29 ago. 2019.

3.GS1 Brasil. **Leitor de código de barras: entenda como funciona..** Disponível em:
<<https://blog.gs1br.org/leitor-de-codigo-de-barras-como-funciona/>>.Acesso em: 29 ago. 2019.

4.Dino. EXAME. **Problemas na gestão de estoques podem causar falência de empresas.** Disponível em:
<<https://exame.abril.com.br/negocios/dino/problemas-na-gestao-de-estoque-podem-causar-falencia-de-empresas/>>.Acesso em: 28 ago. 2019.

5.Mais Empresas. **Seu controle de estoque está armazenado na nuvem?** Disponível em:
<<http://www.maisempresas.com/2018/04/20/seu-controle-de-estoque-esta-armazenado-na-nuvem/>>.Acesso em: 29 ago. 2019.

6.PDVEND BLOG **Conheça as 8 causas mais comuns de problemas no estoque.** Disponível em:
<<https://blog.pdvend.com.br/problemas-no-estoque/>>.Acesso em: 28 ago. 2019.

7.INFOVAREJO **5 Gargalos da operação de um supermercado online.** Disponível em:
<<https://www.infovarejo.com.br/gargalos-do-supermercado-online/>>.Acesso em: 28 ago. 2019.