

**UNIVERSIDADE FEDERAL FLUMINENSE**  
**ANDRÉ LUIZ CHAVES FERNANDES**

**UMA ABORDAGEM DOS CONCEITOS DE DESENVOLVIMENTO DE**  
**UM SISTEMA WEB PARA CONTROLE DE ESTOQUE**

**Niterói**  
**2019**

**ANDRÉ LUIZ CHAVES FERNANDES**

**UMA ABORDAGEM DOS CONCEITOS DE DESENVOLVIMENTO DE  
UM SISTEMA WEB PARA CONTROLE DE ESTOQUE**

Trabalho de Conclusão de Curso  
submetido ao Curso de Tecnologia em  
Sistemas de Computação da  
Universidade Federal Fluminense como  
requisito parcial para obtenção do título  
de Tecnólogo em Sistemas de  
Computação.

**Orientador:**

**Jean de Oliveira Zahn**

**NITERÓI**

**2019**

Ficha catalográfica automática - SDC/BEE  
Gerada com informações fornecidas pelo autor

F363a    Fernandes, André Luiz Chaves  
          UMA ABORDAGEM DOS CONCEITOS DE DESENVOLVIMENTO DE UM SISTEMA  
WEB PARA CONTROLE DE ESTOQUE : INTRODUÇÃO DOS FUNDAMENTOS DE  
CONSTRUÇÃO DE PÁGINAS WEB / André Luiz Chaves Fernandes ;  
Jean de Oliveira Zahn, orientador. Niterói, 2019.  
72 f. : il.

Trabalho de Conclusão de Curso (Graduação em Tecnologia  
de Sistemas de Computação)-Universidade Federal Fluminense,  
Instituto de Computação, Niterói, 2019.

1. Desenvolvimento Web. 2. PHP. 3. Controle de Estoque. 4.  
Projeto de Software. 5. Produção intelectual. I. Zahn, Jean  
de Oliveira, orientador. II. Universidade Federal Fluminense.  
Instituto de Computação. III. Título.

CDD -

**ANDRÉ LUIZ CHAVES FERNANDES**

**UMA ABORDAGEM DOS CONCEITOS DE DESENVOLVIMENTO DE  
UM SISTEMA WEB PARA CONTROLE DE ESTOQUE  
INTRODUÇÃO DOS FUNDAMENTOS DE CONSTRUÇÃO DE  
PÁGINAS WEB**

Trabalho de Conclusão de Curso  
submetido ao Curso de Tecnologia em  
Sistemas de Computação da  
Universidade Federal Fluminense como  
requisito parcial para obtenção do título  
de Tecnólogo em Sistemas de  
Computação.

Niterói, 02 de junho de 2019.

Banca Examinadora:

---

Prof. Jean de Oliveira Zahn, MSc. – Orientador  
UFF – Universidade Federal Fluminense

---

Prof. Rafael Burlamaqui Amaral, Dr. – Avaliador  
CEFET/RJ – Centro Federal de Educação Tecnológica Celso Suckow da Fonseca

Dedico este trabalho a minha esposa Larissa,  
meu filho Davi e aos meus pais.

## **AGRADECIMENTOS**

Agradeço primeiramente a Deus por ter me abençoado grandemente nesta caminhada e a minha está firmada Nele.

A minha família, a minha esposa Larissa, o meu filho Davi e aos meus pais, por ter me apoiado nos meus estudos.

Ao orientador Jean Zahn pela sua colaboração em elaborar este trabalho.

Aos Colegas do curso que me ajudaram muito nas disciplinas do curso.

“A nossa cidadania, porém, está nos céus, de onde esperamos ansiosamente o Salvador, o Senhor Jesus Cristo”.

Filipenses 3:20

## RESUMO

O controle de estoque tem um papel importante dentro da organização, pois traz uma forma de atender as necessidades de consumo, sem precisar elevar ou diminuir o estoque. O sistema de informação ajuda o gestor a visualizar os produtos que precisam comprar e a tomar a decisão de gerenciar o estoque para o correto nível de armazenamento dos itens, e assim sendo, tendo aumento nos resultados financeiros da empresa e redução dos custos de estoque. O presente trabalho consiste em desenvolver um sistema web para o controle de estoque de uma empresa, utilizando a linguagem de programação PHP. A organização terá um software capaz de gerenciar o seu estoque que trará resultados financeiros sustentáveis e redução dos custos de estoque. Para a construção do sistema será utilizado as linguagens HTML5, CSS3 e Javascript, *frameworks* Bootstrap e a biblioteca JQuery. O armazenamento dos dados será utilizado em um banco de dados chamado MySQL. Para construção do sistema web iremos utilizar a arquitetura em 3 camadas, com o objetivo organizar as partes do projeto. E por fim, iremos utilizar testes para corrigir possíveis erros de funcionalidade.

**Palavras-chaves:** Controle de Estoque; Desenvolvimento Web; Projeto de Software; Sistema de Informação; Engenharia de Software.



## **ABSTRACT**

Stock control plays an important role within the organization, because it provides a way to meet consumption needs, without having to raise or lower inventory. The information system helps the manager to visualize the products that need to be bought and to make the decision to manage the inventory to the correct level of storage of the items, and thus, increasing the company's financial results and reducing inventory costs. The present work consists of developing a web system for the inventory control of a company, using the PHP programming language. The organization will have software capable of managing its inventory that will bring sustainable financial results and reduce inventory costs. For the construction of the system will be used the HTML5, CSS3 and Javascript languages, Bootstrap frameworks and the JQuery library. The data storage will be used in a database called MySQL. For building the web system we will use the architecture in 3 layers, with the objective to organize the parts of the project. And finally, we will use tests to correct possible functionality errors.

**Keywords: Stock Control; Web development; Software Project; Information system; Software Engineering.**

## LISTA DE ILUSTRAÇÕES

Figura 1: Exemplo de tag HTML .....	22
Figura 2: Exemplo de conteúdo CSS .....	23
Figura 3: Exemplo de código em Javascript .....	24
Figura 4: Exemplo de código utilizando JQuery .....	25
Figura 5: Exemplo de código utilizando PHP .....	27
Figura 6: Exemplo de requisição-resposta HTTP .....	27
Figura 7: Construção do Software .....	30
Figura 8: Ciclo de vida clássico .....	30
Figura 9: Depuração .....	36
Figura 10: Diagrama de caso de uso .....	39
Figura 11: Diagrama de Entidade e Relacionamento .....	41
Figura 12: Tela de cadastro de usuário .....	42
Figura 13: Tela de cadastro de usuário faltando informação .....	43
Figura 14: Tela de login .....	44
Figura 15: Tela de alterar senha .....	45
Figura 16: Tela de cadastro de fornecedor .....	46
Figura 17: Tela de cadastro de produto .....	47
Figura 18: Tela relação de produto .....	47
Figura 19: Tela de cadastro de entrada de produto .....	48
Figura 20: Tela de relação de entrada de produto .....	49
Figura 21: Tela de estoque de produto .....	50
Figura 22: Tela de cadastro de saída de produto .....	51
Figura 23: Tela de relação de saída de produto .....	51
Figura 24: Tela de cadastro de pedido de produto .....	52
Figura 25: Tela de relação de pedido de produto .....	53
Figura 26: Tela de relatório dos produtos em estoque .....	54
Figura 27: Tela de logout .....	54

Figura 28: Inicia sessão .....	55
Figura 29: Encerra sessão por inatividade .....	55

## LISTA DE TABELAS

Tabela 1: Comparação das ferramentas em cada projeto .....	21
Tabela 2: Caso de Uso Efetuar Login .....	61
Tabela 3: Caso de Uso Usuário .....	62
Tabela 4: Caso de Uso Fornecedor .....	63
Tabela 5: Caso de Uso Produto .....	65
Tabela 6: Caso de Uso Entrada de Produto .....	66
Tabela 7: Caso de Uso Saída de Produto .....	67
Tabela 8: Caso de Uso Pedido de Compra .....	68
Tabela 9: Caso de Uso Relatório .....	70
Tabela 10: Caso de Uso Logout .....	70
Tabela 11: Caso de Uso Alteração de Senha .....	71

## LISTA DE ABREVIATURAS E SIGLAS

CNPJ – Cadastro Nacional de Pessoa Jurídica

CPF – Cadastro de Pessoas Físicas

CSS – *Cascading Style Sheets*

DER – Diagrama de Entidade e Relacionamentos

ER - Entidade e Relacionamentos

HTML – *Hyper Text Markup Language*

HTTP – *Hypertext Transfer Protocol*

MVC – *Model View Controller*

PHP – *Personal Home Page*

SGBD – Sistema de Gerenciamento de Banco de Dados

SQL – *Structured Query Language*

WCF – *Windows Communication Foundation*

## SUMÁRIO

1	INTRODUÇÃO .....	16
1.1	PROBLEMA .....	17
1.2	JUSTIFICATIVA .....	17
1.3	METODOLOGIA .....	18
1.4	OBJETIVO GERAL .....	18
1.5	OBJETIVO ESPECÍFICO .....	18
1.6	DESCRIÇÃO DA ESTRUTURA DO TRABALHO .....	19
2	TRABALHOS RELACIONADOS .....	20
3	DESCRIÇÃO DAS ETAPAS DO PROJETO .....	22
3.1	HTML 5 .....	22
3.2	CSS 3 .....	23
3.3	JAVASCRIPT .....	24
3.4	JQUERY .....	25
3.5	BOOTSTRAP .....	26
3.6	PHP .....	26
3.7	MYSQL .....	28
3.8	ENGENHARIA DE SOFTWARE .....	29
3.8.1	GERENCIAMENTO DE PROJETOS .....	31
3.8.2	PROJETO DE SOFTWARE .....	32
3.9	ARQUITETURA EM 3 CAMADAS .....	33
3.10	TESTE DE SOFTWARE .....	34
3.10.1	VERIFICAÇÃO E VALIDAÇÃO .....	35
3.10.2	REVISÃO DE SOFTWARE .....	35
3.10.3	DEPURAÇÃO .....	36
4	DESENVOLVIMENTO DO PROJETO .....	37
4.1	LEVANTAMENTO DOS REQUISITOS .....	37

4.2 CASOS DE USO .....	38
4.3 CRIAÇÃO DO BANCO DE DADOS .....	39
4.4 APRESENTAÇÃO DO PROJETO .....	41
4.5 TESTES E MANUTENÇÃO DE SOFTWARE .....	56
CONCLUSÕES E TRABALHOS FUTUROS .....	57
CONCLUSÕES .....	57
TRABALHOS FUTUROS .....	58
REFERÊNCIAS BIBLIOGRÁFICAS .....	59
APÊNDICE .....	61
APÊNDICE A – DESCRIÇÃO DO CASO DE USO .....	61

# 1 INTRODUÇÃO

A utilização de sistemas corporativos tem auxiliado as empresas na tomada de decisão e no armazenamento de dados para o seu controle. Em algumas organizações precisa ser armazenado uma grande quantidade de dados e isso seria possível com o auxílio de um software especializado.

A elaboração de um sistema, dependendo do tamanho de componentes que farão parte da aplicação web, pode ser muito complexo. Para diminuir a complexidade o desenvolvedor precisa colocar no papel os requisitos que farão parte do sistema e depois passar como forma de código. A etapa de coletar os requisitos precisa estar alinhado com o que o cliente está precisando. A seguir, tem-se um conceito de como analisar os requisitos de software.

Para entender a natureza do(s) programa(s) a ser(em) construído(s), o engenheiro (“analista”) de software deve compreender o domínio da informação para o software (descrito no Capítulo 6), bem como a função, desempenho e interfaces exigidos. Os requisitos, tanto para o sistema como para o software, são documentados e revistos com o cliente [12, p. 33].

O entendimento daquilo que o cliente está buscando é importante nesta fase para que a entrega do produto final esteja coerente com o que foi pedido. Durante a construção do software, é fundamental a revisão dos requisitos coletados com o cliente.

A engenharia de software que usa técnicas para o desenvolvimento de sistemas, com o intuito de entregar softwares de qualidade.

Entretanto, ao combinarmos métodos abrangentes para todas as fases de desenvolvimento do software, melhores ferramentas para automatizar esses métodos, blocos de construção mais poderosos para a implantação do software, melhores técnicas para a garantia da qualidade do software e uma filosofia de coordenação predominante, controle e administração, podemos



conseguir uma disciplina para o desenvolvimento do software – disciplina esta chamada engenharia de software [12, p. 30 e 31].

A entrega do software com qualidade é um diferencial no desenvolvimento do produto final, trazendo confiabilidade do sistema desenvolvido e facilidade de manutenção para possíveis alterações que venham a acontecer.

O sistema web presente neste trabalho possuirá um cadastro de materiais, dos itens que entram e saem do estoque, um relatório dos produtos que estão no estoque, uma tela de login do usuário que está fazendo a retirada e cadastro de fornecedores.

As etapas de desenvolvimento serão mostradas para uma melhor compreensão do desenvolvimento do sistema web de controle de estoque.

## **1.1 PROBLEMA**

Neste presente trabalho será construído um sistema web para controle de estoque de uma empresa detalhando as etapas de elaboração. A dificuldade entre os desenvolvedores é entender o que o usuário precisa, e depois, passar os requisitos coletados para o processo de desenvolvimento do software. Depois de entregue o software, normalmente existem alterações que o usuário pede para atender as suas necessidades. O desenvolvimento do software precisa estar alinhado as necessidades do cliente para o desenvolvedor entender detalhadamente o processo.

## **1.2 JUSTIFICATIVA**

Atualmente, as organizações têm utilizado software para auxiliar na gestão dos processos corporativos. A construção de um sistema web é importante para que todos da empresa tenham acesso. A elaboração de um software para atender as

necessidades do usuário otimiza o trabalho do empregado, maximiza os processos empresarias e traz competitividade para a organização no mercado.

### 1.3 METODOLOGIA

Para construir um sistema web é preciso escolher a linguagem para este tipo de aplicação que neste atual projeto será a linguagem de programação PHP, como *back-end* e as linguagens HTML, CSS, Javascript, Bootstrap, como *front-end*. E conseqüentemente, modelar um banco de dados para formar as tabelas que serão visualizados e inseridos no sistema, que neste caso é o SGBD MySQL. E fazer testes de verificação e validação para que a aplicação funcione corretamente.

### 1.4 OBJETIVO GERAL

Desenvolver um sistema web para controle de estoque, mostrando as etapas de elaboração para a construção de um software.

### 1.5 OBJETIVO ESPECÍFICO

Os objetivos específicos apresentados a seguir, para o sistema web de controle de estoque, são:

- Levantar os requisitos que farão parte das páginas web;
- Realizar a elaboração de algoritmos para a construção do sistema web de controle de estoque;
- Modelar um banco de dados para a formulação das tabelas;

- Realizar testes para o funcionamento dos componentes que compõem o sistema de controle de estoque.

## **1.6 DESCRIÇÃO DA ESTRUTURA DO TRABALHO**

A seguir, será apresentado detalhes de cada capítulo composto neste trabalho.

- Capítulo 1: É apresentado uma introdução do assunto que será tratado neste trabalho, descrevendo o problema encontrado, a justificativa do tema, a metodologia aplicada, o objetivo geral e específico deste trabalho.
- Capítulo 2: Apresenta os trabalhos relacionados fazendo comparação de projetos que envolvem o desenvolvimento web.
- Capítulo 3: É mostrado as técnicas de desenvolvimento deste trabalho. São descritas as etapas de construção de um sistema web, como: ferramentas utilizadas para o desenvolvimento da aplicação web, conceitos de engenharia de software, e por último, conceitos de realização de teste de software.
- Capítulo 4: São mostradas as telas do sistema web de controle de estoque, descrevendo suas funcionalidades.
- Capítulo 5: Apresenta as conclusões que foram realizadas durante todas as fases de construção do sistema web para controle de estoque.

## 2 TRABALHOS RELACIONADOS

A seguir será mostrado os componentes que farão parte da criação das páginas web do sistema de controle de estoque e uma comparação dos softwares de controle de estoque que existem no mercado atualmente.

O desenvolvimento do sistema de controle de estoque é importante para a empresa, pois traz um gerenciamento dos produtos que precisam manter em estoque.

Existem no mercado sistemas web de controle de estoque com mais funcionalidade do que o proposto neste trabalho. O projeto não irá aprofundar em construir um software complexo, mas em detalhar as etapas de construção e gerenciamento do projeto de acordo com o escopo inicial realizado pela empresa para a elaboração de um sistema web de controle de estoque.

Alguns pontos positivos encontrados em sistema web no mercado são: sistema integrado e online, disponível em mobile, possui notificação em tempo real, possui função de chat interno, monta gráfico detalhando a movimentação e faz estatísticas dos últimos meses. E alguns pontos negativos encontrados são: interface não amigável ao usuário.

O projeto 1 descreve as etapas de desenvolvimento do software web para a gerenciamento de pedidos. Na construção do sistema foram utilizados como ferramentas o HTML5, CSS3, Bootstrap, Javascript, JQuery, PHP 5 (linguagem de programação), MySQL (banco de dados), arquitetura MVC, metodologia de desenvolvimento Scrum e testes de software. [9, p. 38].

Já o projeto 2 apresenta os passos de desenvolvimento do software web para a previsão de estoques. Na elaboração do sistema foram utilizados um *Web Service*, WCF (biblioteca do *framework* .Net), C# (linguagem de programação), SQL Server (banco de dados), arquitetura MVC e validação do sistema. [15, p. 61-74].

O projeto 3 apresenta as ferramentas que foi desenvolvido o sistema web para inscrição em atividades acadêmicas. O software é composto das ferramentas

HTML, CSS, Javascript, JQuery, PHP (linguagem de programação), MySQL (banco de dados) e validação do sistema. [8, p. 20].

Em comparação aos trabalhos desenvolvidos já mencionados antes, o presente trabalho irá mostrar o desenvolvimento de um sistema web para controle de estoque utilizando as ferramentas HTML5, CSS3, Javascript, JQuery, *framework* Bootstrap, PHP (linguagem de programação), MySQL (banco de dados) e teste de software. Essas ferramentas serão descritas passo a passo para a compreensão de como desenvolver um sistema web. A tabela abaixo faz uma comparação das ferramentas utilizadas no projeto atual para cada trabalho citado anteriormente.

Tabela 1: Comparação das ferramentas em cada projeto.

	HTML5	CSS3	Javascript	Bootstrap	MySQL	PHP	MVC
Lima e Silva (2016) – Projeto 1	X	X	X	X	X	X	X
Vieira (2013) – Projeto 2							X
Librelato e Belusso (2013) – Projeto 3	X	X	X		X	X	
Abordagem Corrente (2019)	X	X	X	X	X	X	

### 3 DESCRIÇÃO DAS ETAPAS DO PROJETO

Neste capítulo serão apresentadas as ferramentas e o processo de construção de uma página web para o controle de estoque. Será descrita cada uma das funcionalidades como: HTML 5, CSS 3, Javascript/JQuery e PHP 7. E também o gerenciador de banco de dados MySQL e *framework* Bootstrap 4.3. O presente documento abordará questões sobre engenharia de software. E, por fim, faremos teste de software para que o sistema esteja funcionando corretamente.

#### 3.1 HTML 5

O HTML (*Hypertext Markup Language*), que traduzindo para o português linguagem de marcação de hipertexto, é uma linguagem de marcação para a construção de texto de páginas web e atualmente está na sua quinta versão.

Para o bom entendimento das definições podemos resumir hipertexto como todo o conteúdo inserido em um documento para web e que tem como principal característica a possibilidade de se interligar a outros documentos da web [16, p. 22].

Esse tipo de linguagem faz a construção da página de forma estática, utilizando elementos específicos chamados *tags* para visualização do conteúdo na web.

Um exemplo de *tag* que forma um *site* é relacionado na Figura 1:

```
1
2
3  <p>Forma um parágrafo</p>
4
5
```

Figura 1: Exemplo de tag HTML.

Esse exemplo mostra que é formado um parágrafo quando for visualizar no navegador. A construção é uma *tag* de abertura (<p>) e uma *tag* de fechamento (</p>), no HTML nem todas as *tags* precisa de uma *tag* de fechamento.

### 3.2 CSS 3

O CSS (*Cascading Style Sheet*), que traduzindo para o português, significa folhas de estilos em cascata, é um mecanismo para adicionar estilos no conteúdo na página, como por exemplo, cor, fonte, efeitos de animação, alinhamento, etc.

A CSS funciona com HTML, mas não é HTML. Ela é uma linguagem completamente diferente. Embora HTML forneça a estrutura para um documento organizando informações em títulos, parágrafos, listas com marcadores etc., a CSS funciona lado a lado com o navegador para fazer a HTML ter uma boa aparência [17, p. 1].

O desenvolvimento do CSS é composto de um seletor e um bloco que contém a formatação do conteúdo.

Um exemplo de uma formação de CSS é descrito na Figura 2:

```
1
2
3 <style type="text/css">
4     p {
5         text-align:center;
6     }
7 </style>
8
9 <p>Forma um parágrafo</p>
10
```

Figura 2: Exemplo de conteúdo CSS.

Essa formação de CSS faz com que o seletor p que contém um bloco de propriedades de alinhamento de texto e faz referência a *tag* <p> seja centralizada, ou seja o conteúdo que está entre as *tags* <p> e </p> seja centralizado, quando for visualizado no navegador.

### 3.3 JAVASCRIPT

O Javascript é uma linguagem de programação interpretada de alto nível, ela manipula os elementos do HTML, trazendo interatividade as páginas web. Por exemplo ao clicar no botão da tela do navegador, aparece uma mensagem, solicitando o usuário se deseja ou não fazer aquela operação.

As funcionalidades que podem ser inseridas em páginas web a partir da linguagem JavaScript são inúmeras. Com Javascript, podemos interagir com os visitantes de uma página a partir de campos de formulário, acessar e modificar o conteúdo e as características de uma página, salvar informações no navegador do usuário, auxiliar na construção de layouts complexos, exibir opções de compras e alterar características de produtos de acordo com as escolhas dos clientes ou, ainda, criar jogos interativos que rodam no browser do internauta em qualquer plataforma (Windows, Mac ou Linux) e em qualquer dispositivo (computador, tablet ou celular) [18, p. 23].

A linguagem Javascript é usada tanto no *front-end* (no navegador), como também no *back-end* (no servidor), utilizando a tecnologia Node.js. É uma linguagem popular empregada para o desenvolvimento web, sendo construída juntamente com HTML e CSS. A Figura 3 mostra um trecho de um código em Javascript:

```
1
2 <!DOCTYPE html>
3 <html>
4 <body>
5
6 <ul>
7   <li>1º Item</li>
8   <li>2º Item</li>
9   <li>3º Item</li>
10 </ul>
11
12 <button onclick="captura()">Try it</button>
13
14 <p id="quant"></p>
15
16 <script type="text/javascript">
17 function captura() {
18   var x = document.getElementsByTagName("LI");
19   document.getElementById("quant").innerHTML = x.length;
20 }
21 </script>
22
```

Figura 3: Exemplo de código em Javascript.

Esse código significa que o script está acessando a *tag* e trazendo o resultado da quantidade de `<li>` do HTML. O objeto *document* constitui o acesso a toda página web, o método *getElementsByTagName* ao elemento `<li>` e o método



`document.getElementById` referencia o identificador `id="quant"` que está dentro da `tag` `<p>`.

### 3.4 JQUERY

O JQuery é uma biblioteca Javascript com uma maneira simples de escrever o código. “O JQuery fornece uma maneira simples de executar uma variedade de tarefas Javascript comuns rápida e consistentemente, em todos os principais navegadores e sem a necessidade de nenhum código de fallback.” [2, p. 294].

No JQuery se escreve menos código do que em Javascript, pois utiliza seletores para acessar os elementos do HTML e CSS. Para utilizar as funcionalidades do JQuery é preciso baixar o arquivo ou fazer um link direto do servidor.

Um exemplo de um seletor JQuery a seguir:

```

1
2 <!DOCTYPE html>
3 <html>
4 <head>
5 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js"></script>
6 </head>
7 <body>
8
9 <ul>
10 <li>1</li>
11 <li>2</li>
12 <li>3</li>
13 </ul>
14
15 <script>
16 $(document).ready(function() {
17     var elemento = $("li:nth-child(odd)");
18     elemento.css("color", "yellow");
19 });
20 </script>
21
22 </body>
23 </html>

```

Figura 4: Exemplo de código utilizando JQuery.

Esse seletor é definido como um símbolo de `$(cifrão)` que identifica que é um JQuery e em seguida o que está em parênteses e aspas simples é um elemento `<li>` do HTML que vai colorir de amarelo o 1º e 3º elemento com o método CSS.

### 3.5 BOOTSTRAP

O Bootstrap é um *framework front-end* que utiliza classes HTML, CSS e Javascript com modelos de design prontos e responsivo e que se adequa a qualquer tamanho de tela.

Bootstrap é o mais popular framework JavaScript, HTML, e CSS para desenvolvimento de sites e aplicações web responsivas e alinhadas com a filosofia mobile first. Torna o desenvolvimento front-end muito mais rápido e fácil. Indicado para desenvolvedores de todos os níveis de conhecimento, dispositivos de todos os tipos e projetos de todos os tamanhos [19, p. 20].

Para o desenvolvedor otimiza o trabalho de criar uma página web. Neste trabalho será utilizado a versão 4.3.

### 3.6 PHP

O PHP é uma linguagem de script dinâmica de código aberto para o desenvolvimento páginas web. Para que haja dinamismo nas páginas web o PHP é adicionado entre as *tags* HTML. A seguir, uma descrição de como funciona a uma página sem PHP. “Imagine, por exemplo, um *website* que deseja exibir notícias em sua página principal, mostrando a cada dia, ou a cada hora, notícias diferentes. Seria inviável fazer isso utilizando apenas HTML.” [11, p. 21].

A Figura 5 mostra o funcionamento de um código em PHP embutido em uma *tag* HTML que exibe a hora atual.

```

1
2  <html>
3    <body>
4      <p>
5        <?php
6          echo date('d/m/Y');
7        ?>
8      </p>
9    </body>
10  </html>
11

```

Figura 5: Exemplo de código utilizando PHP.

A linguagem PHP é interpretada, pois precisa de um programa para poder interpretar o seu código-fonte. Os arquivos PHP são colocados em um servidor que executa a extensão “.php” e envia para o navegador.

O HTTP é um protocolo de transferência de hipertexto, onde os clientes fazem a requisição de páginas aos servidores e estes enviam as respostas. “Quando um usuário requisita uma página Web (por exemplo, clica sobre um hiperenlace), o navegador envia ao servidor mensagens de requisição HTTP para os objetos da página.” [7, p. 73]. A Figura 6 ilustra um exemplo de solicitação entre o cliente e servidor.

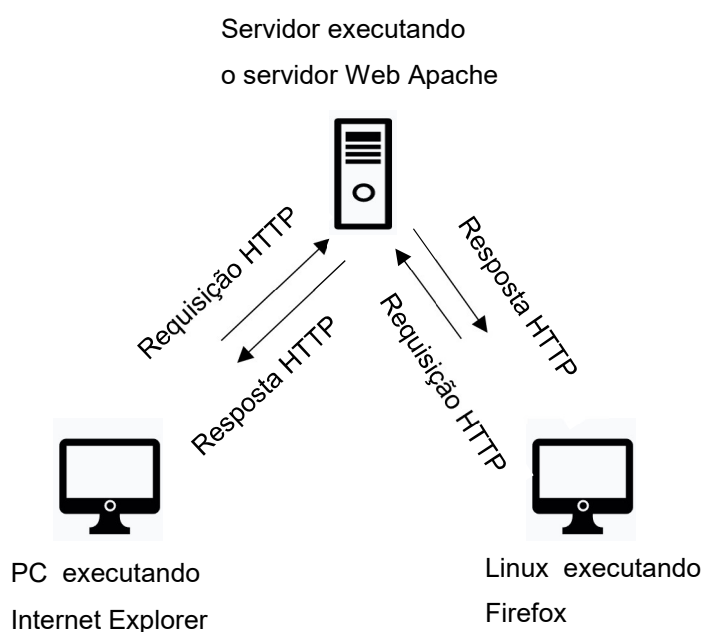


Figura 6: Exemplo de requisição-resposta HTTP. (Fonte: [7, p. 73])

A linguagem de programação PHP se conecta ao banco de dados para que as informações da tabela ou consulta seja mostrada na página web, tornando a página web uma estrutura dinâmica. Neste trabalho será empregado a versão 7.

### 3.7 MYSQL

O MySQL é um SGBD (Sistema de Gerenciamento de Banco de Dados) que utiliza a linguagem SQL para a manipulação de banco de dados.

O MySQL é um servidor e gerenciador de banco de dados (SGBD) relacional, de licença dupla (sendo uma delas de software livre), projetado inicialmente para trabalhar com aplicações de pequeno e médio portes, mas hoje atendendo a aplicações de grande porte e com mais vantagens do que seus concorrentes [10, p. 22].

A velocidade no acesso às informações são uma característica importante para o uso do MySQL mesmo se for uma base muito grande em relação aos outros SGBDs. “Para bases de dados muito grandes, o MySQL faz um acesso mais rápido que o PostgreSQL.” [11, p. 154].

Uma característica importante do MySQL é a sua compatibilidade com outros sistemas operacionais, tornando-se uma aceitação por parte dos desenvolvedores em utilizar um SGBD. Existem outras características que o MySQL fornece como: SGBD possui todas as funcionalidades de um Sistema Gerenciador de Banco de Dados onde é chamado de MySQL Server; portabilidade para as outras linguagens de programação; *multithreads* que aumenta a velocidade do processamento; formas de armazenamento para tipos de tabelas em cada situação diferente; velocidade no acesso ao dados; segurança no acesso aos dados e o uso de criptografia no acesso ao sistema de conexões; o SQL empregado no SGBD é executado em alta velocidade; *fulltext search* é uma funcionalidade busca neste estilo para soluções integradas, e por fim, capacidade de armazenamento e execução extremamente elevado [10, p. 27 – 29]. Com essas funcionalidades que o MySQL traz pode ser utilizado vários recursos para ser aplicado em projetos com grandes volumes

de dados. A seguir, é descrito as vantagens que o SGBD MySQL traz para aplicações em sua interface.

Outras vantagens do MySQL é a possibilidade de realizar exclusões de tabelas diferentes em um único comando DELETE, permitir utilizar tabelas de bancos de dados distintos em uma mesma consulta de união (JOIN), maior capacidade de colunas em uma única tabela, entre outros [10, p. 32].

Neste trabalho será utilizado a ferramenta PhpMyAdmin, que é uma interface Web para a manipulação da base dados do MySQL. O uso do PhpMyAdmin facilita na elaboração das tabelas por sua interface de fácil manuseio, trazendo os principais recursos do MySQL. “O grande objetivo do PhpMyAdmin é tornar mais simples e prática a interação com o MySQL, utilizando os conceitos de programação gráfica.” [10, p. 85].

Neste projeto será utilizado o SGBD MySQL por possuir código aberto e simples de ser manipulado.

### **3.8 ENGENHARIA DE SOFTWARE**

A Engenharia de Software traz um conjunto de métodos, ferramentas e os procedimentos de forma organizada, utilizando princípios de gestão de projetos. O objetivo é elaborar sistema de qualidade e baixo custo. “A engenharia de software compreende um conjunto de etapas que envolve métodos, ferramentas e os procedimentos...” [12, p. 32].

No desenvolvimento de um sistema, é preciso gerenciar o projeto de construção do software para que o prazo de entrega seja cumprido.

A Figura 7 mostra as etapas de construção do software:

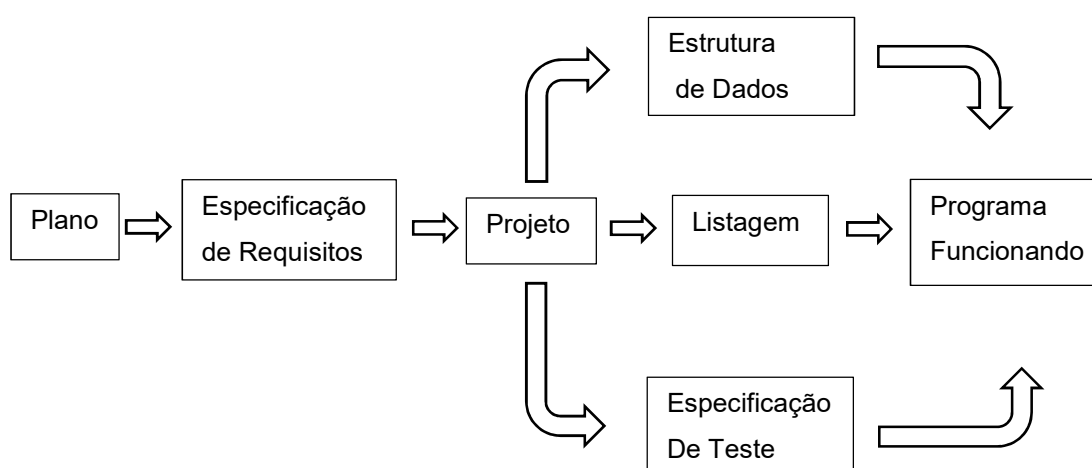


Figura 7: Construção do Software. (Fonte: [12, p. 30])

Para a elaboração do sistema de controle de estoque, vai ser utilizado alguns conceitos da engenharia de software. Um dos paradigmas da engenharia de software é ciclo de vida clássico, um modelo que segue as etapas para a construção do software. Também chamado modelo cascata, pois uma etapa só inicia quando a outra termina e assim até chegar a última fase. A Figura 8 descreve o ciclo de vida clássico.

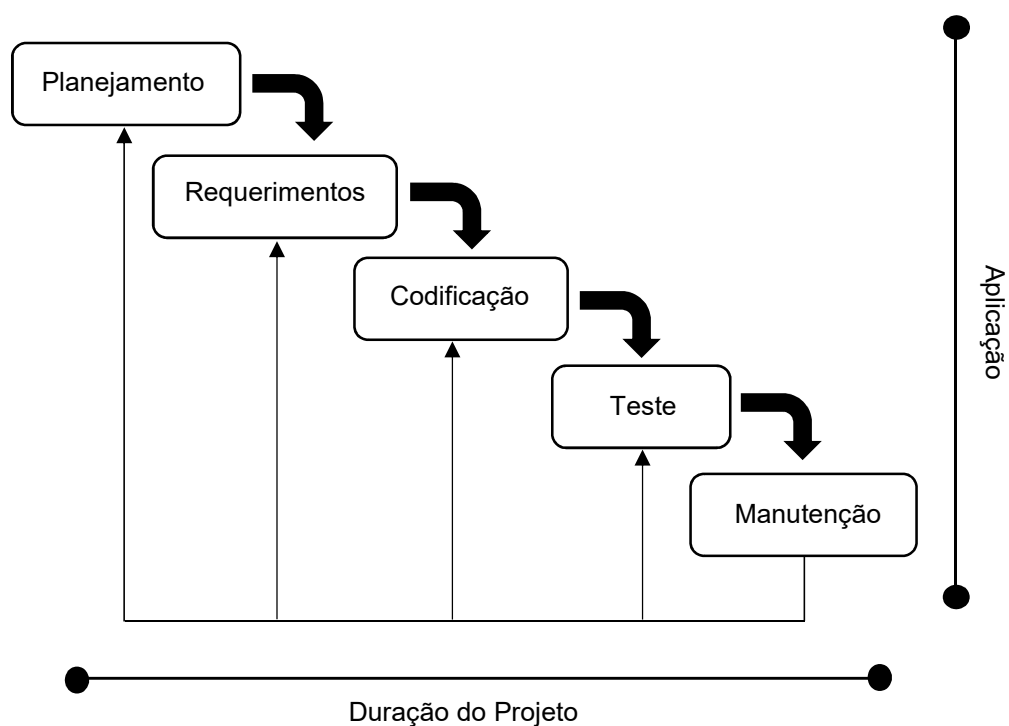


Figura 8: Ciclo de vida clássico. (Fonte: [4, p. 183 e 184])

A seguir será descrito cada uma das etapas do ciclo de vida em cascata e posteriormente aplicar no sistema web controle de estoque.

- 1º: Planejamento constitui a coleta dos requisitos que vai compor o sistema. Nessa fase o responsável pelo projeto busca conhecer a necessidade do seu cliente e adquirir as informações necessárias para o projeto. [4, p. 183].
- 2º: Requerimentos é analisar as informações levantadas pelo cliente anteriormente e entender como será o processo de construção do software. [4, p. 183].
- 3º: Codificação representa os códigos utilizados no sistema para que os requisitos que foi coletado passe para as instruções que o computador entenda. [4, p. 183].
- 4º: Testes é iniciado com testes nos códigos e verificar que todas as funcionalidades do sistema estão de acordo com a etapa do projeto e verificar erros de lógicas ou de digitação nos códigos. [4, p. 183 e 184].
- 5º: Manutenção é onde o software poderá ser encontrado erros ou mudanças quando for entregue ao cliente. Como mostrado na Figura 8, a manutenção é refeita para cada uma das etapas. [4, p. 183 e 184].

Para o sistema web de controle de estoque, será aplicado o ciclo de vida clássico, por ser um modelo simples de seguir e adequado para o projeto deste trabalho.

### **3.8.1 GERENCIAMENTO DE PROJETOS**

Em um projeto no desenvolvimento de um software, o gerenciamento de projetos é uma ferramenta importante para o atingir o projeto final. O gerenciamento de projetos tem como objetivo acompanhar o seu andamento, com o intuito de evitar falhas, prever os recursos necessários, estabelecer prazos, definir custos e minimizar os riscos. No projeto, a equipe precisa está motivada e as tarefas distribuída pelo pessoal.

No gerenciamento de projetos, existem áreas de conhecimento que são gerenciadas pelo gerente para obter sucesso no projeto de software. As áreas são:

gerenciamento de integração, gerenciamento do escopo, gerenciamento do tempo, gerenciamento do custo, gerenciamento da qualidade, gerenciamento dos recursos humanos, gerenciamento das comunicações, gerenciamento das partes interessadas, gerenciamento dos riscos e gerenciamento das aquisições. [14, p. 22].

O ciclo de vida do projeto é composto por fases para que as suas atividades sejam acompanhadas e tenha sucesso no desenvolvimento do software. As fases são: início e concepção do projeto, organização, preparação e planejamento, execução do trabalho e encerramento. [14, p. 19]

“A gerência de projetos é a primeira camada do processo de engenharia de software.” [12, p. 55]. Estabelecer conceitos de gestão de projetos no desenvolvimento de um sistema e seguir todas as etapas de um gerenciamento de projetos para a qualidade e manutenção do software. O sucesso de um projeto de software está na condução dos conceitos de gerenciamento de projetos.

Para conduzir um projeto de software bem-sucedido, devemos compreender o escopo do trabalho a ser feito, os riscos em que incorremos, os recursos exigidos, as tarefas a serem executadas, os marcos de referência a serem acompanhados, o esforço (custo) despendido e a programação a ser seguida [12, p. 55].

As etapas do produto de software são importantes para que o projeto tenha um resultado satisfatório em relação às expectativas do cliente.

### **3.8.2 PROJETO DE SOFTWARE**

O projeto de software é uma parte da engenharia de software que trata de descrever a estrutura do sistema antes do programador realizar seu desenvolvimento. A sua importância traz uma organização para o projeto que será construído e qualidade no produto final. “A importância do projeto de software pode ser estabelecida com uma única palavra – qualidade.” [12, p. 417]. O projeto traz qualidade no produto final, pois isso se deve anteriormente a coleta de requisitos do cliente que é especificado com exatidão na fase inicial do processo.



A necessidade de elaborar um projeto está na manutenção do sistema, pois fica fácil em identificar quando pequenas alterações forem feitas se o software falhar.

### 3.9 ARQUITETURA EM 3 CAMADAS

Neste presente trabalho vai ser utilizada a arquitetura em 3 camadas para o desenvolvimento do sistema web. Em uma aplicação cliente-servidor, essa arquitetura é uma maneira de separar em 3 camadas: cliente, aplicação, banco de dados.

A 1ª camada é a de cliente que consiste na resposta do servidor quando o navegador solicita um pedido. O navegador exibe o conteúdo que foi requerido pelo servidor. Essa camada não precisa de software especializado para executar as aplicações, como HTML, CSS e Javascript.

A 2ª camada é a de aplicação ou também chamada camada intermediária onde a lógica do sistema é processada. A integração do banco de dados e cliente acontece nesta camada. O banco de dados armazena as informações e exibe no cliente. A camada de aplicação é executada no servidor, onde o cliente solicita as informações e devolvido como resposta do servidor. A solicitação pode ser um conteúdo estático como em HTML ou dinâmico que neste caso é a linguagem de programação usada neste projeto o PHP.

A 3ª e última camada é a de banco de dados onde se encontra as informações para a exibição do conteúdo na 1ª camada a do cliente. O banco de dados possui uma segurança dos dados, utilizando sistema de *backup*. Pode ser feita consultas, inserção, alteração ou a remoção de um algum registro ou todos da tabela. O banco de dados tem a capacidade de gerenciar as informações para acesso aos registros. O MySQL é o SGBD empregado neste projeto para o gerenciamento e manipulação do banco de dados.

A arquitetura em 3 camadas separa as camadas para uma melhor compreensão e definição de cada componente no sistema web.

“A criação de camadas é uma das técnicas mais comuns que os projetistas de software usam para quebrar em pedaços um sistema complexo de software”. [5, p. 37]. Um exemplo de como funcionam a arquitetura em 3 camadas é descrito a seguir:

Ao pensar em um sistema em termos de camadas, você imagina os subsistemas principais no software dispostos de forma parecida com camadas de um bolo, em que cada camada repousa sobre uma camada mais baixa. Nesse esquema, a camada mais alta usa vários serviços definidos pela camada mais baixa, mas a camada mais baixa ignora a existência da camada mais alta [5, p. 37].

A arquitetura em 3 camadas é uma maneira de compreender melhor a construção de um projeto de software.

### **3.10 TESTE DE SOFTWARE**

Os testes de software são uma ferramenta importante no desenvolvimento, pois nesta fase tem o objetivo de apresentar erros ou falhas. A importância do teste de software está ligada à sua qualidade para que o sistema seja entregue ao usuário de acordo com as especificações que foi proposto inicialmente.

Quando o desenvolvedor finaliza a construção do software, e se esse for um sistema complexo, pode existir vários erros de lógica ou funcionamento de uma determinada ação. A atividade de teste é um fator importante para descobrir esses erros devido ao resultado da ação humana.

Um teste de software é definido como: “...teste de software é o processo que visa a sua execução de forma controlada, com o objetivo de avaliar o seu comportamento baseado no que foi especificado.” [13, p. 10]. Os testes são importantes para garantir a funcionalidade de todos os componentes do sistema. A verificação e validação são técnicas que foi visto anteriormente e faz parte do teste de software.

### **3.10.1 VERIFICAÇÃO E VALIDAÇÃO**

Quando o software é produzido e entregue precisa atender as especificações e necessidades do usuário. Antes de entregar o sistema, o desenvolvedor precisa fazer a verificação e validação.

A validação faz uma análise do sistema para atender as necessidades dos usuários, enquanto a verificação faz também uma análise do sistema para estabelecer se o que está sendo construído está de acordo com a especificação inicial do software.

“Verificação: neste conceito estão incluídos todos os testes que permitem verificar se o software está sendo construído corretamente, ou seja, teste de integração e teste de sistema.” [13, p. 29]. E também, “Validação: neste conceito estão incluídos os testes que permitem validar se o software está fazendo o que foi definido nos requisitos, ou seja, teste de aceitação.” [13, p. 29].

A verificação e validação tem um papel importante no desenvolvimento do software, por mostrar resultados que agregam tanto a produtividade quanto à qualidade.

### **3.10.2 REVISÃO DE SOFTWARE**

A revisão de software está ligada a uma análise de toda a sua especificação para identificar se os requisitos definidos inicialmente estão de acordo com as necessidades dos usuários. A revisão tem o intuito de realizar a verificação e validação dos artefatos e detectar defeitos que possam existir.

As revisões de software são um “filtro” para o processo de engenharia de software. Ou seja, as revisões são aplicadas em vários pontos durante o desenvolvimento de software e servem para descobrir defeitos que possam ser eliminados. [12, p. 736].

Quando o desenvolvedor faz uma revisão no sistema ele está buscando uma qualidade para a entrega ao usuário. E para manter a qualidade do software, o analista precisa fazer uma busca no código afim de encontrar possíveis defeitos.

### 3.10.3 DEPURAÇÃO

A depuração (*debugging*) é a arte de encontrar erros e corrigir de forma rápida. Quando o sistema é depurado é feito uma análise do código e descobrir problemas de lógica e/ou falta de algum componente. A depuração não é um teste de software, e sim, um processo posterior ao teste.

“... quando um caso de teste revela um erro, a depuração é o processo que resulta na remoção do erro.” [12, p. 865].

O processo de depuração de o desenvolvedor visualizar o que foi feito no sistema, passando por cada comando do código, e analisando se as entradas, processamento e saída estão de acordo com o esperado. “O processo de depuração tenta ligar o sintoma à causa, levando assim à correção do erro.” [12, p. 866]. A Figura 9 demonstra como é realizado um processo de depuração.

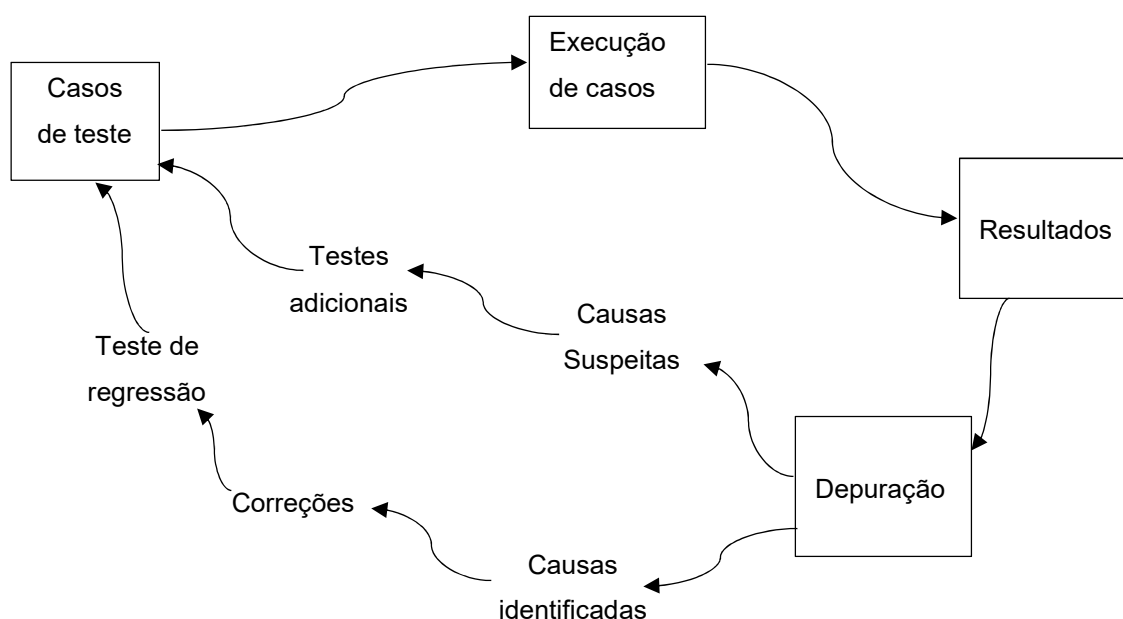


Figura 9: Depuração. (Fonte: [12, p. 866])

## **4. DESENVOLVIMENTO DO PROJETO**

Neste capítulo são apresentadas as etapas de desenvolvimento do sistema web para controle de estoque em uma empresa. Serão abordados os assuntos sobre levantamento dos requisitos, os casos de uso, os diagramas de entidade e relacionamento, apresentação do sistema web e aplicações de teste de software.

### **4.1 LEVANTAMENTO DOS REQUISITOS**

As etapas de construção do software, visto no Capítulo 3, é um processo importante para o desenvolvimento. O primeiro passo é a de Planejamento dos Requisitos que constitui o levantamento dos requisitos descrito pelo cliente e o segundo passo é o de Requerimentos que significa entender os requisitos apresentado na etapa anterior.

A seguir, são descritos os requisitos levantados pelo cliente para atender a sua necessidade da utilização do software.

Os requisitos levantados são:

- O sistema deverá apresentar uma tela de login e senha;
- O sistema deverá apresentar uma tela de alteração de senha;
- O sistema deverá permitir fazer logoff;
- O sistema deverá permitir cadastrar usuário, fornecedor, produto, entrada, saída e pedido de compra;
- O sistema deverá permitir alterar usuário, fornecedor e produto
- O sistema deverá permitir excluir usuário e fornecedor
- O sistema deverá permitir visualizar dados cadastrados dos usuários, fornecedores, produtos, entradas, saídas, estoques, pedido de compra e relatório.

- O sistema deverá permitir gerar o relatório em Excel.

Essa é a lista de requisitos que o usuário precisa para o uso da ferramenta no setor de estoque da organização.

## 4.2 CASOS DE USO

Os casos de uso é uma ferramenta para entender a sequência da funcionalidade do software. O caso de uso é composto por atores e sistema que formam um cenário. Com a formação do cenário é possível obter uma sequência das etapas do projeto.

... os casos de uso descrevem como um sistema interage com atores externos. Cada caso de uso representa uma funcionalidade que um sistema fornece a seus usuários. Os casos de usos são úteis para capturar requisitos informais [1, p. 137].

E ainda: “Um ator é um usuário externo direto de um sistema – um objeto ou conjunto de objetos que se comunica diretamente com o sistema, mas que não é parte dele”. [1, p. 136].

A Figura 10 mostra a elaboração de um diagrama de caso de uso para melhor compreensão das etapas de desenvolvimento e interação entre o usuário e o sistema. A descrição do diagrama de caso de uso está no Apêndice A.

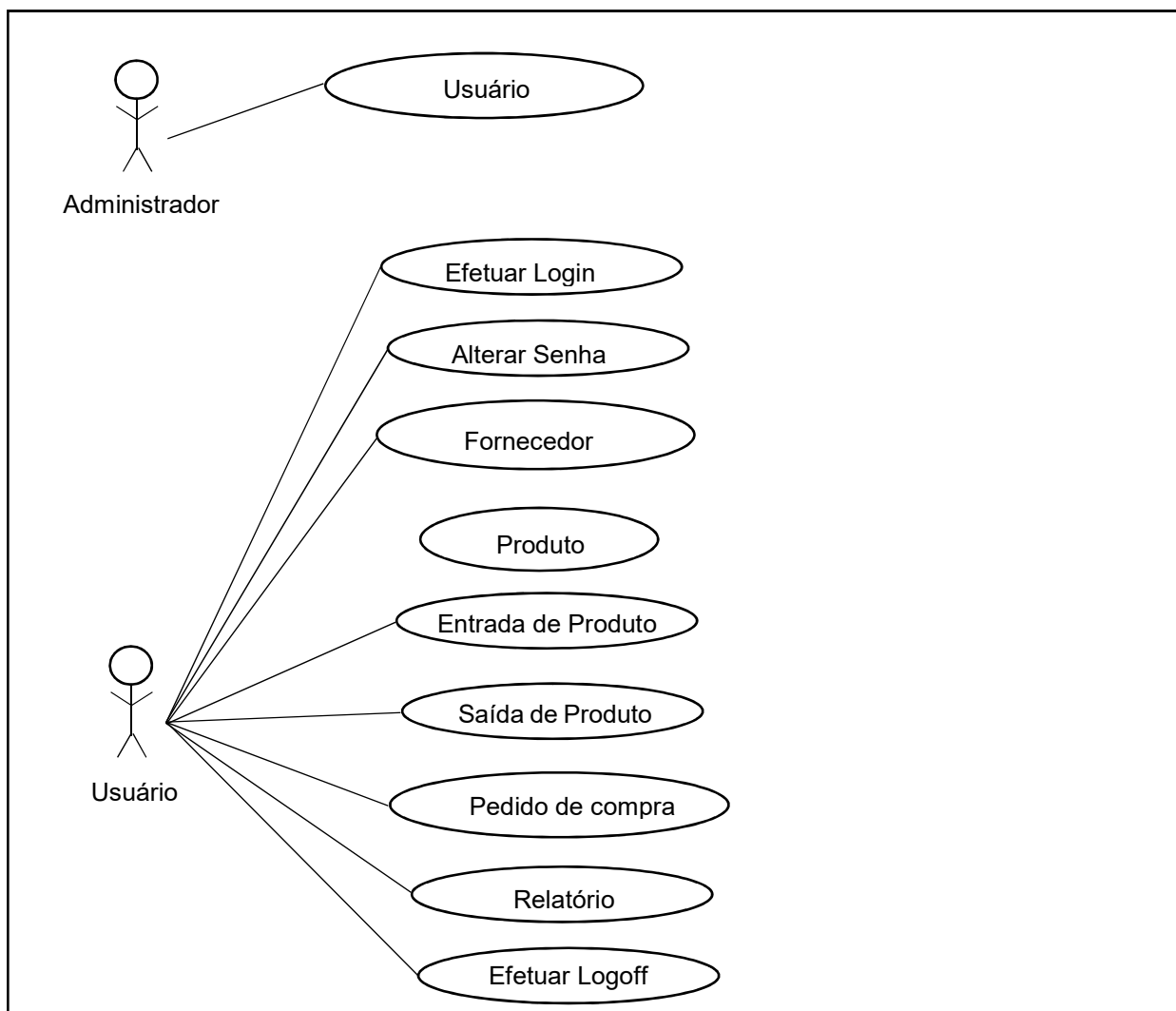


Figura 10: Diagrama de caso de uso. (Fonte: O Autor (2019)).

### 4.3 CRIAÇÃO DO BANCO DE DADOS

Um sistema de banco de dados relacional consiste em um conjunto de tabelas ou relações. O modelo relacional, que se encontra no sistema gerenciador de banco de dados, é uma forma de que os dados se deparam em uma tabela e pode existir relações entre elas. “O modelo relacional representa o banco de dados como uma coleção de relações”. [3, p. 39].

Os elementos que compõem uma tabela são linhas, colunas e os registros. “Na terminologia formal do modelo relacional, uma linha é chamada de tupla, um cabeçalho da coluna é chamado de atributo e a tabela de relação.” [3, p. 39].

Neste presente trabalho será utilizado o modelo de Entidade-Relacionamento (ER) para auxiliar na construção das tabelas do banco de dados. O conceito do modelo ER é descrito como: “Nesta técnica, o modelo de dados é representado através de um modelo entidade-relacionamento (modelo ER). Geralmente, um modelo ER é representado graficamente através de um diagrama entidade-relacionamento (DER).” [6, p. 34].

O conceito de entidade é definido como um conjunto de objetos onde estarão as informações. Dentro do diagrama é representado por um retângulo. Já o relacionamento é a associação entre as entidades. No diagrama é representado por um losango. E na estrutura do modelo ER existem os atributos que são as propriedades das entidades e relacionamentos e é representado por uma elipse. Uma entidade deve possuir uma chave primária em um ou mais atributos para que os valores sejam únicos.

Para construção de banco de dados um importante conceito é a de cardinalidade de relacionamento quantos elementos entre as entidades estão se relacionando. “Há duas cardinalidades a considerar: a cardinalidade máxima e a cardinalidade mínima.” [6, p. 39]. Existem cardinalidade de (0 para 1), (1 para 1), (1 para n) e (n para m). Neste trabalho será empregado o modelo de engenharia de informação (Erwin), onde as entidades são representadas na forma de um retângulo, os atributos é o que estão listados no retângulo e os relacionamentos são os linhas acompanhados dos símbolos de um traço, círculo ou uma forma que lembra um pé de galinha.

No MySQL foi construído as tabelas que fará parte do sistema de controle de estoque.

A Figura 11 mostra o diagrama de entidade e relacionamento.



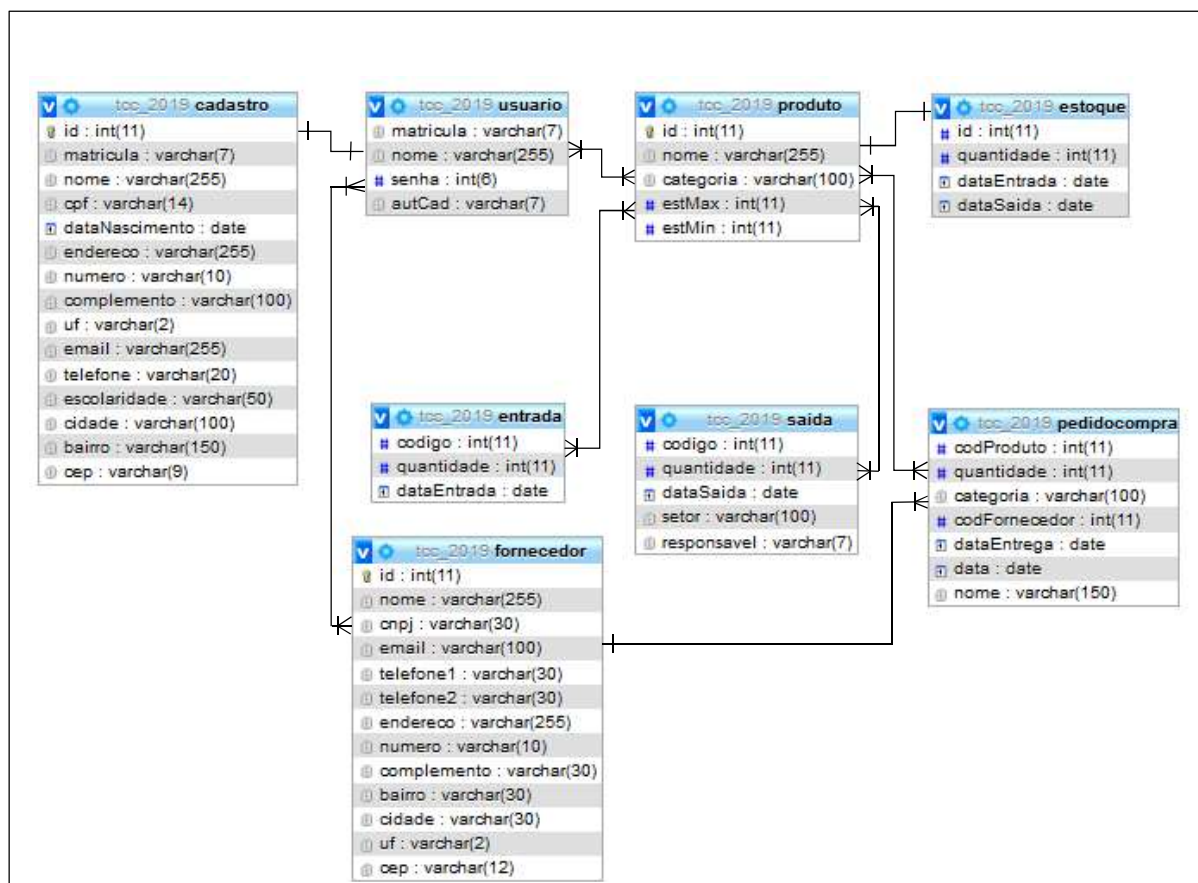


Figura 11: Diagrama de Entidade e Relacionamento. (Fonte: O Autor (2019)).

#### 4.4 APRESENTAÇÃO DO PROJETO

A seguir, serão apresentadas as telas do projeto controle de estoque e as suas principais funcionalidade. Foram inseridas informações fictícias para testar se o sistema está funcionando de acordo com os requisitos levantados pelo cliente. Esta etapa de construção de software, visto no Capítulo 3, é a de codificação que é a forma de passar para o computador através de código os requisitos levantados pelo cliente. Para executar o sistema web foi empregado um servidor Apache para exibição do conteúdo em PHP e para escrita do código foi utilizado um editor de código-fonte *Visual Studio Code*.

A primeira tela a ser apresentada é de cadastro de usuário, onde somente que tem acesso é o administrador do sistema ou o gerente do setor de estoque. As

informações do usuário são cadastradas na tabela para que o mesmo possa ter acesso. Ao cadastrar o funcionário, o sistema verifica se este já foi cadastrado através de uma verificação no banco de dados pelo CPF. Se já existe um cadastro é emitido uma mensagem de alerta que o empregado já foi cadastrado. Se for o primeiro cadastro é inserido os dados e gerado uma matrícula automaticamente e uma senha provisória de 6 números para que possa ser trocada no próprio sistema. Na tela é exibido uma relação de usuários que foram cadastrados e ao lado um botão de alterar e excluir para que possa permitir fazer alteração e exclusão de empregados.

A imagem mostra a interface de usuário para o cadastro de um novo usuário no sistema 'Controle de Estoque'. No topo, há uma barra azul com o ícone de uma prateleira e o texto 'Controle de Estoque' à esquerda, e 'Bem vindo, Usuario A' com um botão 'Sair' à direita. O título principal da seção é 'Cadastro de Usuário'. Abaixo dele, os campos de formulário são organizados da seguinte forma: 'Matricula' (campo único); 'Nome' (campo único); 'CPF' (campo único), 'Data Nascimento' (campo único) e 'Escolaridade' (menu suspenso); 'E-mail' (campo único) e 'Telefone' (campo único); 'Endereço' (campo único) e 'Número' (campo único); 'Complemento' (campo único), 'Bairro' (campo único), 'Cidade' (campo único) e 'UF' (menu suspenso); e 'CEP' (campo único). No final dos campos, há um botão azul 'Cadastrar'.

Figura 12: Tela de cadastro de usuário.

O sistema verifica em todas as telas de cadastro de usuário, fornecedor, produto, entrada, saída, pedido de compra, login e alterar senha se todos os campos foram preenchidos para que não insere dados sem informações. A próxima tela do

cadastro mostra um caso em que o usuário não preenche todos os campos. O sistema exibe uma informação que o campo precisa ser preenchido.

The image shows a web application interface for 'Controle de Estoque'. The header is blue with the text 'Bem vindo, Usuário A' and a 'Sair' button. The main content area is titled 'Cadastro de Usuário'. It contains a form with the following fields: 'Matricula' (text input), 'Nome' (text input with 'Usuario D' as a placeholder), 'CPF' (text input), 'E-mail' (text input), 'Telefone' (text input), 'Endereço' (text input), 'Número' (text input), 'Complemento' (text input), 'Bairro' (text input), 'Cidade' (text input), 'UF' (dropdown menu), and 'CEP' (text input). A blue 'Cadastrar' button is located at the bottom left of the form. A white modal dialog box is centered over the form, displaying the message 'O campo CPF deve ser informado!' and an 'OK' button.

Figura 13: Tela de cadastro de usuário faltando informação.

A página web exibe os usuários cadastrados pelo administrador do sistema e ao lado de cada informação tem dois botões de alterar e excluir. O botão de alterar é acionado se precisar fazer alguma alteração como: escolaridade, endereço, telefone e etc. E o botão excluir é ativado se o usuário não mais fizer parte da lista de funcionários da empresa ou mudar de setor, por exemplo.

As próximas Figuras será mostrado as funcionalidades do sistema criado para o usuário. A Figura 14 mostra a tela de login onde depois de ser cadastrado pelo administrador poderá ter acesso ao sistema de controle de estoque. Quando o usuário entrar no sistema será verificado se a matrícula e senha confere. Se for inserido uma

matrícula que não esteja cadastrada aparecerá uma mensagem que o usuário não cadastrado. Também poderá fazer a troca da senha, na Figura 15 mostra a tela de alterar senha. O sistema faz uma verificação e emite uma mensagem de alerta se for inserido uma matrícula que não seja cadastrada, se a senha informada é diferente da senha cadastrada ou se a nova senha é igual a confirmação de senha. Se todas as informações estiverem corretas, o usuário já pode fazer login com a senha alterada.

Controle de Estoque

## Login

Somente números

Entrar

[Alterar senha](#)

2019 - TCC Tecnologia em Sistemas de Computação

Figura 14: Tela de login.

Controle de Estoque Sair

## Alterar Senha

Matrícula

Senha Atual

Somente números

Senha

Somente números

Confirme a Senha

Somente números


Alterar

2019 - TCC Tecnologia em Sistemas de Computação

Figura 15: Tela de alterar senha.

O usuário cadastrado e senha alterada na sequência poderá navegar pelos menus na barra de navegação vertical na tela principal. Quando o usuário fizer o login a página que irá aparecer é a de produtos em estoque. Primeiramente é preciso cadastrar fornecedor e produto para poder ter a movimentação dos produtos. Depois de cadastrar fornecedores, o usuário poderá visualizar a relação de fornecedores onde também poderá alterar ou excluir. O sistema exibe uma informação se o usuário inserir um fornecedor já cadastrado na tabela do banco de dados pelo CNPJ.

A seguir na Figura 16, é mostrado uma tela de cadastro de fornecedores.

 Controle de Estoque

Bem vindo, Usuario D

Sair

### Cadastro de Fornecedor

Nome

Nome

CNPJ

CNPJ

E-mail

E-mail

Telefone 1

Telefone 1

Telefone 2

Telefone 2

Endereço

Endereço

Número

Número

Complemento

Complemento

Bairro

Bairro

Cidade

Cidade

UF

UF


CEP

CEP

Cadastrar

Figura 16: Tela de cadastro de fornecedor.

As informações sobre o cadastro de produtos são mostradas na Figura 17. Se o produto já tiver sido cadastrado anteriormente, um aviso de alerta é apresentado na página. O sistema permite fazer alteração do produto através do botão alterar. Uma relação de produtos pode ser visualizada na tela. Nas páginas onde consta a relação de fornecedor, produto, entrada, saída, estoque, pedido de compra, relatório e usuário foi posto um *plug-in* JQuery para utilizar nas tabelas em HTML. Esta traz a funcionalidade de ordenar cada coluna, selecionar a quantidade de itens que serão mostradas, um campo de pesquisa instantânea e paginação de página. A Figura 18 mostra a relação de produtos cadastrados.


**Controle de Estoque**

**Bem vindo, Usuario D**

Sair


## Cadastro de Produto

Nome

Categoria
 
 Estoque Máximo
 
 Estoque Mínimo

**Cadastrar**

Figura 17: Tela de cadastro de produto.


**Controle de Estoque**

**Bem vindo, Usuario D**

Sair

Fornecedor  
**Produto**  
Entrada  
Saída  
Estoque  
Pedido de Compra  
Relatório  
Cadastrar Usuário

### Produto

Cadastrar

Mostrando 10 Registros por Pagina
Pesquisar: Cód., Nome, Categ...

Código	Nome	Categoria	Est. Máximo	Est. Mínimo	
1	Produto A1	B	20000	1000	Alterar
2	Produto B1	C	10000	300	Alterar
3	Produto C	A	450	20	Alterar
4	Produto D	A	15000	200	Alterar
5	Produto E	C	7000	180	Alterar
6	Produto F	C	45000	150	Alterar
7	Produto G	B	5000	60	Alterar
8	Produto H	B	14000	160	Alterar

Mostrando Pagina 1 de 1

Anterior
1
Proxima

Figura 18: Tela relação de produto.

A próxima etapa que o usuário terá de fazer depois de cadastrar os produtos é determinar a quantidade de produtos que chegaram no estoque. Para isso uma página de cadastro de entrada de produtos é mostrada na Figura 19. Se o produto não tiver sido cadastrado anteriormente um aviso será mostrado na tela. Se a quantidade for maior que o estoque máximo cadastrado uma mensagem de alerta é exibida na tela.

A interface apresenta um cabeçalho azul com o ícone de uma casa e o texto 'Controle de Estoque' à esquerda, e 'Bem vindo, Usuario D' com um botão 'Sair' à direita. O título principal da seção é 'Cadastro de Entrada de Produto'. Abaixo dele, há dois campos de entrada: 'Cód. Produto' e 'Quantidade', cada um com um input correspondente. Um botão azul com o texto 'Cadastrar' está posicionado abaixo dos campos.

Figura 19: Tela de cadastro de entrada de produto.

A relação de cadastro de produtos é mostrada na Figura 20. Nela foram inseridos os produtos cadastrados e quantidade de cada item que entrou no estoque.



Bem vindo, Usuario D

Sair

Fornecedor
Produto
Entrada
Saída
Estoque
Pedido de Compra
Relatório
Cadastrar Usuário

## Entrada

Cadastrar

Mostrando 10 Registros por Pagina
Pesquisar: Cód., Quantidade ...

Código	Quantidade	Dt. Entrada
1	15000	10/05/2019
2	8500	02/05/2019
3	200	15/05/2019
4	10000	16/05/2019
5	5500	29/05/2019
6	40000	29/05/2019
7	3200	29/05/2019
8	13100	20/05/2019

Mostrando Pagina 1 de 1

Anterior
1
Proxima

Figura 20: Tela de relação de entrada de produto.

Se a entrada da quantidade do item já foi cadastrada, já é possível visualizar a sua movimentação no sistema. A seguir é mostrada o código do produto, o nome, a quantidade, categoria, uma barra de status que mostra a porcentagem de produtos que estão no estoque e um botão de pedido de compra que redireciona para a página de cadastro com o campo código, nome e categoria do produto preenchido automaticamente como mostra a Figura 24. O preenchimento automático é feito passando por parâmetro o código do produto, no momento que é acionado o botão pedido de compra, e fazendo uma consulta na tabela produto do banco de dados e preenchendo os campos. A porcentagem da barra de status é calculada de acordo com a quantidade de estoque em relação ao estoque máximo. A Figura 21 mostra essa funcionalidade.

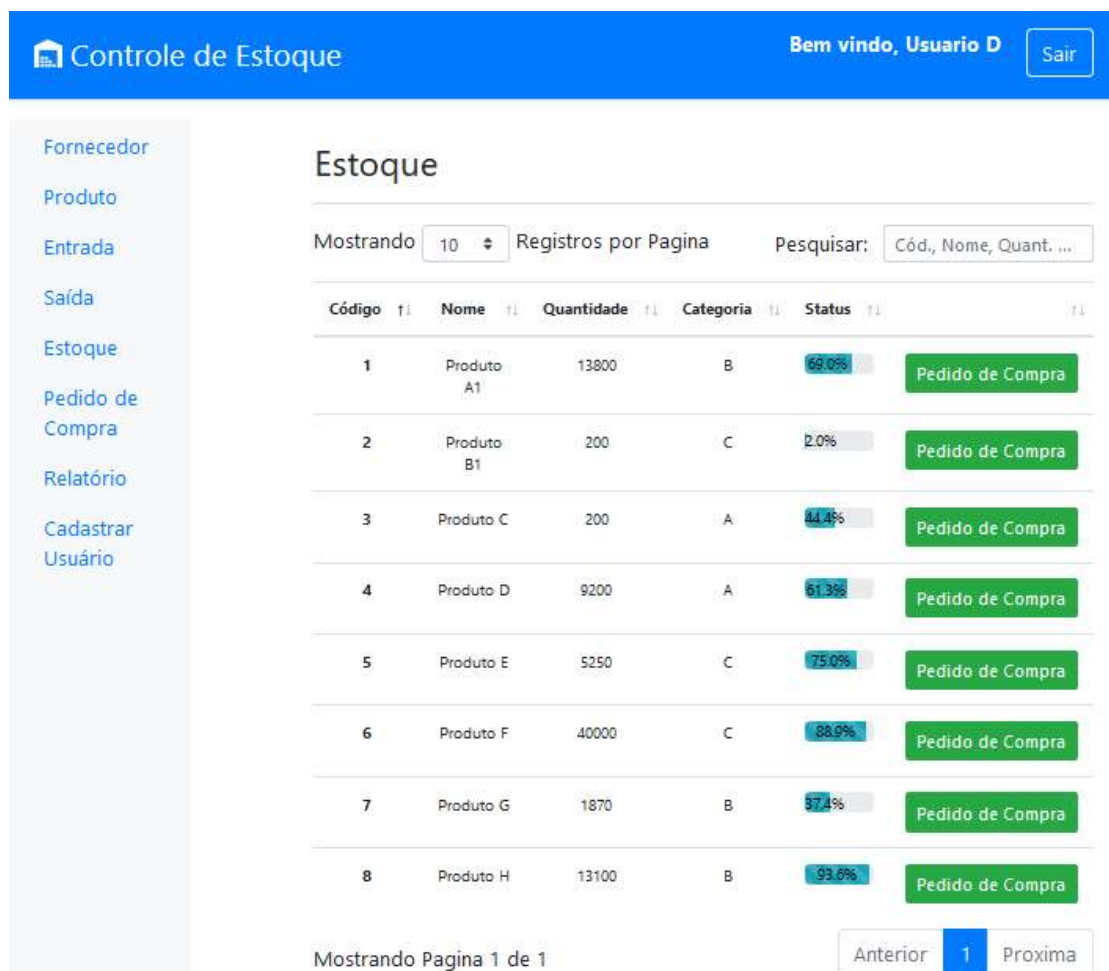


Figura 21: Tela de estoque de produto.

Se algum item sair do estoque uma página para cadastrar a saída de produto. Se o produto não tiver sido cadastrado anteriormente um aviso será mostrado na tela. E se a quantidade do produto a ser retirado for menor a que está no estoque um alerta é exibido na tela. Também o sistema permite enviar um e-mail automaticamente para o gerente do setor, se a quantidade que está no estoque for que o estoque mínimo. Isso faz com que antecipe a compra do item antes que zera o estoque. A Figura 22 mostra o cadastro de saída do produto.

Controle de Estoque

Bem vindo, Usuário D

Sair

## Cadastro de Saída de Produto

Código

Quantidade

Setor

Código

Quantidade

Setor

Cadastrar

Figura 22: Tela de cadastro de saída de produto.

A relação de cadastro de saída de produtos é mostrada na Figura 23. Nela foram inseridos os produtos que saíram dos produtos cadastrados, quantidade de cada item, a data de saída e o setor.

Controle de Estoque

Bem vindo, Usuário D

Sair

Fornecedor  
Produto  
Entrada  
Saída  
Estoque  
Pedido de Compra  
Relatório  
Cadastrar Usuário

### Saída

Cadastrar

Mostrando 10 Registros por Pagina
Pesquisar:

Cód., Quantidade ...

Código	Quantidade	Dt. Saída	Setor
1	1200	21/05/2019	Elétrica
2	7800	31/05/2019	Elétrica
2	500	22/05/2019	Mecânica
4	800	23/05/2019	Elétrica
5	250	29/05/2019	Civil
7	330	29/05/2019	Mecânica
7	1000	29/05/2019	Elétrica

Mostrando Pagina 1 de 1

Anterior
1
Proxima

Figura 23: Tela de relação de saída de produto.

Um pedido de compra poderá ser realizado, solicitando ao gerente do setor do estoque através de e-mail disparado automaticamente pelo sistema para que seja providenciado a compra do produto junto ao setor de compra. Se o produto e/ou fornecedor não tiverem sido cadastrado anteriormente um aviso será mostrado na tela. Se o usuário informar o nome do produto diferente do código, uma mensagem de alerta será mostrada. Se for digitada a categoria diferente que está no cadastrado um alerta é exibido na tela. Se a quantidade pedida for maior que o estoque máximo do produto é apresentado uma mensagem informando. E, por fim, se a data de reposição é menor que a data informada é mostrado um aviso na página. A data de reposição é baseada na data do dia mais 7 dias. Isso para evitar que o usuário informe uma data muito próxima, e consequentemente não daria tempo suficiente para fazer a reposição. A Figura 24 apresenta a página de cadastro de pedido de compra.

Controle de Estoque Bem vindo, Usuario D Sair

### Cadastro de Pedido de Produto

Código do Produto Nome do Produto

2 Produto B1

Quantidade Categoria

Quantidade C

Código do Fornecedor Data limite de Reposição

Código do Fornecedor Data limite de Reposição

Cadastrar

Figura 24: Tela de cadastro de pedido de produto.

A relação de cadastro de pedido de compras é mostrada na Figura 25. A tabela mostra o código do produto, nome do produto, quantidade, categoria, código do fornecedor, data limite de reposição e data do cadastro do pedido de cada item.

Controle de Estoque

Bem vindo, Usuário D

Sair

Fornecedor

Produto

Entrada

Saída

Estoque

Pedido de Compra

Relatório

Cadastrar Usuário

## Pedido de Compra

Mostrando 10 Registros por Pagina      Pesquisar: Cód., Quantidade ...

Cód. Produto	Nome Produto	Quantidade	Categoria	Cód. Fornecedor	Data Limite de Reposição	Data do Pedido
5	Produto E	110	C	1	20/06/2019	30/05/2019
2	Produto B1	150	C	2	18/06/2019	30/05/2019
8	Produto H	200	B	2	20/06/2019	30/05/2019
1	Produto A1	1200	B	3	16/06/2019	30/05/2019
2	Produto B1	2000	C	1	17/07/2019	29/06/2019

Mostrando Pagina 1 de 1

Anterior 1 Proxima

Figura 25: Tela de relação de pedido de produto.

O usuário poderá visualizar o relatório dos itens que estão no estoque. A página fornece um botão para que possa fazer o download do relatório em HTML para Excel. A seguir é mostrada o código do produto, o nome, a categoria, o estoque máximo, estoque mínimo e a quantidade de itens em estoque. Na coluna quantidade, se o valor menor que o estoque mínimo a cor da letra fica vermelho e se for maior, muda para a cor verde. A Figura 26 mostra um exemplo de relatório fornecido pelo sistema.

**Controle de Estoque**

Bem vindo, **Usuario D**

Sair

Fornecedor  
 Produto  
 Entrada  
 Saída  
 Estoque  
 Pedido de Compra  
 Relatório  
 Cadastrar Usuário

## Relatório

Mostrando  Registros por Pagina
 

Pesquisar:

Cód. Produto	Nome Produto	Categoria	Est. Máximo	Est. Mínimo	Quantidade
1	Produto A1	B	20000	1000	13800
2	Produto B1	C	10000	300	200
3	Produto C	A	450	20	200
4	Produto D	A	15000	200	9200
5	Produto E	C	7000	180	5250
6	Produto F	C	45000	150	40000
7	Produto G	B	5000	60	1870
8	Produto H	B	14000	160	13100

Mostrando Pagina 1 de 1
 

Anterior
 

1

 Proxima

Figura 26: Tela de relatório dos produtos em estoque.

Para realizar logoff do sistema, as telas de navegação têm um botão de sair e quando for clicado é finalizada a sessão, aparecendo uma mensagem se deseja sair do sistema ou não. A Figura 27 mostra a tela de logout.

**Controle de Estoque**

## Logout

Deseja realmente sair do sistema?
 

Não

Sim

Figura 27: Tela de logout.

Nas páginas onde consta o conteúdo do sistema foi utilizado sessões para que o usuário não fique logado no sistema por muito tempo sem utilizar e verificar se foi feito login. Nela são utilizados a função `session_start()`; para iniciar a sessão e a variável `$_SESSION['tempo']` para contar o tempo que o usuário ficará inativo. As variáveis `$_SESSION['matricula']` e `$_SESSION['senha']` se estiver vazias o empregado é redirecionado para a tela de login. Esse tipo de método evita que o empregado fique com uma página inativa por um período e também navegue pelas páginas sem efetuar login. “Uma sessão é um período de tempo durante o qual uma pessoa navega pelas páginas de um *site*”. [11, p. 220]. A Figura 28 e 29 mostram, respectivamente, o código utilizado para abrir sessão, as variáveis que são passadas por parâmetros e um arquivo `encerra_sessao.php` que é chamado para calcular o tempo de inatividade do sistema. Se a variável `$tempoLimite` for maior que 600 é redirecionado para página de logoff.

```

2      session_start();
3      $mat = $_SESSION['matricula'];
4      $_SESSION['matricula'] = $mat;
5      $senha = $_SESSION['senha'];
6      $tempo = $_SESSION['tempo'];
7
8      if((empty($mat)) OR (empty($senha))){
9          echo "<script>alert('Voce precisa efetuar login!');location.href='login.php'</script>";
10     }
11     else {
12
13         //Encerra sessão por inatividade
14         require_once("encerra_sessao.php");
15

```

Figura 28: Inicia sessão.

```

1  <?php
2      //Encerra sessão por inatividade
3      $tempoLimite = time() - $tempo;
4      if(($tempoLimite) > 600)
5      {
6          echo "<script>location.href='logout_encerrar.php'</script>";
7      }
8      else
9      {
10         $tempo = time();
11     }
12
13  ?>

```

Figura 29: Encerra sessão por inatividade.

## 4.5 TESTES E MANUTENÇÃO DE SOFTWARE

As duas últimas etapas do ciclo de vida de software utilizado neste projeto são respectivamente, testes e manutenção visto no Capítulo 3. Depois de concluída a etapa de codificação, a próxima fase é realizar teste para que todas as funcionalidades do sistema estejam funcionando corretamente. Os testes foram realizados pelo autor em todas as páginas, fazendo verificação e validação nos formulários, nas ações dos botões de inserção, alteração e exclusão, no acesso ao banco de dados para exibir as informações e se o que foi programado está de acordo com o que foi especificado nos requisitos.

A última fase é a de manutenção onde pode ser encontrados erros não vistos nos testes e possíveis mudanças por parte do cliente. Para facilitar as alterações que possam acontecer, foi inserido comentários em algumas linhas do código para que tenha entendimento o que aquela linha de código está executando. Os comentários são importantes para quando for fazer alguma alteração sabe-se o que está fazendo aquele comando. Na manutenção o desenvolvedor, terá que voltar em todas as etapas do ciclo de vida do software.



## CONCLUSÕES E TRABALHOS FUTUROS

### CONCLUSÕES

Este trabalho teve objetivo abordar os conceitos das etapas de desenvolvimento de páginas web, construindo um controle de estoque para uma empresa. Foi explicado as ferramentas empregadas, bem como, as fases para a construção do software.

O presente trabalho mostrou a coleta do levantamento dos requisitos por parte do cliente e a elaboração de um diagrama e descrição do caso de uso para melhor entendimento da construção do software. No código foram feitos comentários para um melhor entendimento de cada linha, consequentemente para possíveis alterações, isso facilitará a manutenção. Na conclusão do projeto foi inserido mensagens para usuário informando o usuário o motivo do ocorrido.

Foi escolhido um sistema de controle de estoque para mostrar como funciona a construção de um *website* e o ciclo de vida clássico que auxiliou nas etapas de elaboração.

O emprego do *framework* css Bootstrap facilitou na construção do *layout* da página e otimizando o desenvolvimento da parte lógica da programação. Foi utilizado as linguagens HTML5, PHP, SQL, Javascript e banco de dados MySQL para armazenar as informações. Foram realizados testes para corrigir possíveis problemas de lógica ou digitação e entregar ao cliente o correto funcionamento do sistema.

Estes fatores são primordiais para o processo de desenvolvimento de um sistema web, pois traz agilidade na entrega e qualidade para o software. Procurar ferramentas que facilitam na elaboração do sistema é um fator importante para fase de entrega do produto ao cliente.

Este projeto atende ao que foi proposto inicialmente nos requisitos e pode ser utilizado em uma empresa que possua controle de estoque.

## **TRABALHOS FUTUROS**

Com os testes realizado e o sistema funcionando, o usuário poderá apresentar mudanças para atender as suas necessidades. A seguir serão relacionadas algumas melhorias no sistema de controle de estoque.

- Um relatório mais completo dos produtos em estoque;
- Criação de gráficos, por entrada e saída de produto;
- Cálculos para previsão de estoque.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] BLAHA, Michael; RUMBAUGH, James. **Modelagem e projetos baseados em objetos com UML 2**. 2. ed. Rio de Janeiro: Elsevier, 2006.
- [2] DUCKETT, Jon. **Javacript e JQuery: desenvolvimento de interfaces web interativas**. 1. ed. Rio de Janeiro: Alta Books, 2015.
- [3] ELMASRI, Ramez; NAVATHE, Shamkant B. **Sistemas de banco de dados**. 6. ed. São Paulo: Pearson, 2011.
- [4] FEDELI, Ricardo D.; POLLONI, Enrico G. F.; PERES, Fernando E. **Introdução à ciência da computação**. 2. ed. São Paulo: Cengage Learning, 2010.
- [5] FOWLER, Martin. **Padrões de Arquitetura de Aplicações Corporativas**. 1. ed. São Paulo: Bookman, 2006.
- [6] HEUSER, Carlos Alberto. **Projeto de banco de dados**. 6. ed. Porto Alegre: Bookman, 2009.
- [7] KUROSE, James F.; ROSS, Keith W. **Rede de computadores e a internet: uma abordagem top-down**. 6. ed. São Paulo: Pearson Education do Brasil, 2013.
- [8] LIBRELATO, ~,ç.;° H.; BELUSSO, W. V. **Sistema web para inscrição em atividades acadêmicas**. Trabalho de Conclusão de Curso (Tecnologia em Análise e Desenvolvimento de Sistemas. Universidade Tecnológica Federal do Paraná, Pato Branco, 2013. Disponível em: <  
[http://repositorio.roca.utfpr.edu.br/jspui/bitstream/1/2011/1/PB\\_COADS\\_2013\\_2\\_10.pdf](http://repositorio.roca.utfpr.edu.br/jspui/bitstream/1/2011/1/PB_COADS_2013_2_10.pdf)>. Acesso em: 11 mar. 2019.
- [9] LIMA E SILVA, G.B. de. **Desenvolvimento de um sistema web para gerenciamento de pedidos de delivery de comida em Diamantina/MG**. Trabalho de Conclusão de Curso (Graduação em Sistemas de Informação). Universidade Federal dos Vales do Jequitinhonha e Mucuri, Brasil, 2016.

Disponível em: <<http://decom.ufvjm.edu.br/site/index.php/mn-tcc/category/15-tcc-monografias-2016?download=42:tcc-grabriel>>. Acesso em: 4 mar. 2019.

[10] MILANI, André. **MySQL**: Guia do programador. 1. ed. São Paulo: Novatec Editora, 2006.

[11] NIEDERAUER, Juliano. **Desenvolvendo Websites com PHP**: Aprenda a criar Websites dinâmicos e interativos com PHP e banco de dados. 3. ed. São Paulo: Novatec, 2017.

[12] PRESSMAN, Roger S. **Engenharia de Software**. 1. ed. São Paulo: Pearson Makron Books, 1995.

[13] RIOS, Emerson; FILHO, Trayahú Moreira. **Teste de Software**. 3. ed. Rio de Janeiro: Alta Books, 2013.

[14] SANTOS, Carlos Fernando da Rocha. **Gerenciamento de Projetos**: Conceitos e Representações. 1. ed. Rio de Janeiro: LTC, 2014.

[15] VIEIRA W. **Desenvolvimento de um sistema web para previsão de estoque**. Trabalho de Conclusão de Curso (Graduação em Sistemas de Informação). Universidade do Sul de Santa Catarina, Palhoça, 2013. Disponível em: <[http://pergamum.unisul.br/pergamum/pdf/107724\\_Wagner.pdf](http://pergamum.unisul.br/pergamum/pdf/107724_Wagner.pdf)>. Acesso em: 9 mar. 2019.

[16] SILVA, Maurício Samy. **HTML 5**: A linguagem de marcação que revolucionou a web. 2. ed. São Paulo: Novatec, 2014.

[17] MCFARLAND, David Sawyer. **CSS3**: o manual que faltava: o livro que devia vir na caixa. 1. ed. Rio de Janeiro: Alta Books, 2015.

[18] IEPSEN, Edécio Fernando. **Lógica de Programação e Algoritmos com JavaScript**: Uma introdução à programação de computadores com exemplos e exercícios para iniciantes. 1. ed. São Paulo: Novatec, 2018.

[19] SILVA, Maurício Samy. **Bootstrap 3.3.5**: Aprenda a usar o framework Bootstrap para criar layouts CSS complexos e responsivos. 1. ed. São Paulo: Novatec, 2015.

## APÊNDICE

### APÊNDICE A – DESCRIÇÃO DO CASO DE USO

Tabela 2: Caso de Uso Efetuar Login.

Caso de Uso	UC 01 – Efetuar Login
Descrição	Efetua o login para acessar o sistema
Pré-Condições	Matrícula e senha válidos
Pós-Condições	Continuar logado
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O usuário digita a matrícula e senha e clica no botão entrar na tela de Login</li> <li>2. O sistema envia os dados para o servidor</li> <li>3. O sistema verifica se o usuário está cadastrado</li> <li>4. O sistema recebe do servidor os dados de matrícula e senha corretos</li> <li>5. O sistema redireciona para a página de estoque com sessão ativa</li> </ol>
Fluxo Alternativo	<ol style="list-style-type: none"> <li>3.1. O sistema verifica que o usuário não está cadastrado               <ol style="list-style-type: none"> <li>3.1.1. O sistema mostra uma mensagem “Usuário não cadastrado!”</li> <li>3.1.2. O sistema redireciona a página para o passo 1</li> </ol> </li> <li>4.1. O sistema verifica que os dados de matrícula e senha estão incorretos               <ol style="list-style-type: none"> <li>4.1.1. O sistema mostra uma mensagem “Matrícula e/ou Senha incorreto!”</li> </ol> </li> </ol>

	4.1.2. O sistema redireciona a página para o passo 1
--	--

Tabela 3: Caso de Uso Usuário.

Caso de Uso	UC 02 – Usuário
Descrição	Efetua o cadastro, alteração e exclusão de usuário no sistema
Pré-Condições	UC 01 – Efetuar Login
Pós-Condições	Administrar o acesso aos usuários
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O administrador clica no link chamado “Cadastrar Usuário” no menu vertical do lado esquerdo</li> <li>2. O sistema exibe uma relação de usuários</li> <li>3. O sistema fornece três botões de cadastrar, alterar e excluir <ol style="list-style-type: none"> <li>3.1. O usuário clica no botão cadastrar [passo 3.1.1]</li> <li>3.2. O usuário clica no botão alterar [passo 3.2.1]</li> <li>3.3. O usuário clica no botão excluir [passo 3.3.1]</li> </ol> </li> </ol>
Fluxo Alternativo	<ol style="list-style-type: none"> <li>3.1.1. O sistema exibe a página de cadastro de usuário</li> <li>3.1.2. O usuário preenche os campos que são obrigatórios <ol style="list-style-type: none"> <li>3.1.2.1. O usuário não preenche todos os campos</li> <li>3.1.2.2. O sistema redireciona a página para o passo 3.1.1</li> </ol> </li> <li>3.1.3. O servidor recebe a requisição dos dados do formulário <ol style="list-style-type: none"> <li>3.1.3.1. O servidor verifica que o CPF informado é igual ao cadastrado no banco de dados</li> <li>3.1.3.2. O sistema exibe uma mensagem “CPF já cadastrado no sistema!”</li> <li>3.1.3.3. O sistema redireciona para a página o passo 3.1.1</li> </ol> </li> <li>3.1.4. O servidor armazena os dados no banco de dados</li> <li>3.1.5. O sistema exibe uma mensagem de “Cadastro incluído com sucesso!”</li> <li>3.1.6. O sistema redireciona para a página do passo 2</li> </ol>

	<p>3.2.1. O sistema exibe a página de alteração de usuário</p> <p>3.2.2. O sistema exibe os campos dos formulários preenchidos</p> <p>3.2.3. O usuário preenche os campos que precisam ser alterados</p> <p>3.2.4. O servidor recebe a requisição dos dados do formulário</p> <p>3.2.5. O servidor armazena os dados no banco de dados</p> <p>3.2.6. O sistema exibe uma mensagem de “Alteração incluído com sucesso!”</p> <p>3.2.7. O sistema redireciona para a página do passo 2</p> <p>3.3.1. O sistema exibe uma mensagem “Deseja realmente excluir usuário?”</p> <p>3.3.1.1. O usuário clica no botão não</p> <p>3.3.1.1.1. O sistema redireciona para a página do passo 2</p> <p>3.3.1.2. O usuário clica no botão sim</p> <p>3.3.1.2.1. O servidor exclui o usuário</p> <p>3.3.1.2.2. O sistema exibe uma mensagem de “Excluído com sucesso!”</p> <p>3.3.1.2.3. O sistema redireciona para a página do passo 2</p>
--	--

Tabela 4: Caso de Uso Fornecedor.

Caso de Uso	UC 03 – Fornecedor
Descrição	Efetua o cadastro, alteração e exclusão de fornecedor no sistema
Pré-Condições	UC 01 – Efetuar Login
Pós-Condições	Administrar o acesso aos fornecedores
Fluxo Principal	<p>1. O usuário clica no link chamado “Fornecedor” no menu vertical do lado esquerdo</p> <p>2. O sistema exibe uma relação de fornecedores</p> <p>3. O sistema fornece três botões de cadastrar, alterar e excluir</p>

	<p>3.1. O usuário clica no botão cadastrar [passo 3.1.1]</p> <p>3.2. O usuário clica no botão alterar [passo 3.2.1]</p> <p>3.3. O usuário clica no botão excluir [passo 3.3.1]</p>
Fluxo Alternativo	<p>3.1.1. O sistema exibe a página de cadastro de fornecedor</p> <p>3.1.2. O usuário preenche os campos que são obrigatórios</p> <p>3.1.2.1. O usuário não preenche todos os campos</p> <p>3.1.2.2. O sistema redireciona a página para o passo 3.1.1</p> <p>3.1.3. O servidor recebe a requisição dos dados do formulário</p> <p>3.1.3.1. O servidor verifica que o CNPJ informado é igual ao cadastrado no banco de dados</p> <p>3.1.3.2. O sistema exibe uma mensagem “CNPJ já cadastrado no sistema!”</p> <p>3.1.3.3. O sistema redireciona para a página o passo 3.1.1</p> <p>3.1.4. O servidor armazena os dados no banco de dados</p> <p>3.1.5. O sistema exibe uma mensagem de “Cadastro incluído com sucesso!”</p> <p>3.1.6. O sistema redireciona para a página do passo 2</p> <p>3.2.1. O sistema exibe a página de alteração de fornecedor</p> <p>3.2.2. O sistema exibe os campos dos formulários preenchidos</p> <p>3.2.3. O usuário preenche os campos que precisam ser alterados</p> <p>3.2.4. O servidor recebe a requisição dos dados do formulário</p> <p>3.2.5. O servidor armazena os dados no banco de dados</p> <p>3.2.6. O sistema exibe uma mensagem de “Alteração incluído com sucesso!”</p> <p>3.2.7. O sistema redireciona para a página do passo 2</p> <p>3.3.1. O sistema exibe uma mensagem “Deseja realmente excluir fornecedor?”</p> <p>3.3.1.1. O usuário clica no botão não</p> <p>3.3.1.1.1. O sistema redireciona para a página do passo 2</p>



	<p>3.3.1.2. O usuário clica no botão sim</p> <p>3.3.1.2.1. O servidor exclui o fornecedor</p> <p>3.3.1.2.2. O sistema exibe uma mensagem de “Excluído com sucesso!”</p> <p>3.3.1.2.3. O sistema redireciona para a página do passo 2</p>
--	--

Tabela 5: Caso de Uso Produto.

Caso de Uso	UC 04 – Produto
Descrição	Efetua o cadastro, alteração e exclusão de produto no sistema
Pré-Condições	UC 01 – Efetuar Login
Pós-Condições	Administrar o acesso aos produtos
Fluxo Principal	<p>1. O usuário clica no link chamado “Produto” no menu vertical do lado esquerdo</p> <p>2. O sistema exibe uma relação de produtos</p> <p>3. O sistema fornece três botões de cadastrar, alterar e excluir</p> <p>3.1. O usuário clica no botão cadastrar [passo 3.1.1]</p> <p>3.2. O usuário clica no botão alterar [passo 3.2.1]</p>
Fluxo Alternativo	<p>3.1.1. O sistema exibe a página de cadastro de produto</p> <p>3.1.2. O usuário preenche os campos que são obrigatórios</p> <p>3.1.2.1. O usuário não preenche todos os campos</p> <p>3.1.2.2. O sistema redireciona a página para o passo 3.1.1</p> <p>3.1.3. O servidor recebe a requisição dos dados do formulário</p> <p>3.1.3.1. O servidor verifica que o produto informado é igual ao cadastrado no banco de dados</p> <p>3.1.3.2. O sistema exibe uma mensagem “Produto já cadastrado no sistema!”</p> <p>3.1.3.3. O sistema redireciona para a página o passo 3.1.1</p> <p>3.1.4. O servidor armazena os dados no banco de dados</p> <p>3.1.5. O sistema exibe uma mensagem de “Cadastro incluído com sucesso!”</p>

	<p>3.1.6. O sistema redireciona para a página do passo 2</p> <p>3.2.1. O sistema exibe a página de alteração de produto</p> <p>3.2.2. O sistema exibe os campos dos formulários preenchidos</p> <p>3.2.3. O usuário preenche os campos que precisam ser alterados</p> <p>3.2.4. O servidor recebe a requisição dos dados do formulário</p> <p>3.2.5. O servidor armazena os dados no banco de dados</p> <p>3.2.6. O sistema exibe uma mensagem de “Alteração incluído com sucesso!”</p> <p>3.2.7. O sistema redireciona para a página do passo 2</p>
--	--

Tabela 6: Caso de Uso Entrada de Produto.

Caso de Uso	UC 05 – Entrada de Produto
Descrição	Efetua o cadastro da entrada do produto no sistema
Pré-Condições	UC 01 – Efetuar Login
Pós-Condições	Administrar o acesso das entradas dos produtos
Fluxo Principal	<p>1. O usuário clica no link chamado “Entrada” no menu vertical do lado esquerdo</p> <p>2. O sistema exibe uma relação de produtos que entraram no estoque</p> <p>3. O usuário clica no botão de cadastrar</p> <p>4. O sistema exibe a página de cadastro de entrada de produto</p> <p>5. O usuário preenche os campos que são obrigatórios</p> <p>6. O servidor recebe a requisição dos dados do formulário</p> <p>7. O servidor armazena os dados no banco de dados</p> <p>8. O sistema exibe uma mensagem de “Cadastro incluído com sucesso!”</p> <p>9. O sistema redireciona para a página do passo 2</p>

Fluxo Alternativo	<p>5.1. O usuário não preenche todos os campos</p> <p>5.1.1. O sistema redireciona a página para o passo 4</p> <p>6.1. O servidor verifica que o produto não foi cadastrado</p> <p>6.1.1. O sistema exibe uma mensagem “Produto não cadastrado!”</p> <p>6.1.2. O sistema redireciona a página para o passo 4</p> <p>6.2. O servidor verifica que a quantidade total do produto é maior que o estoque máximo cadastrado</p> <p>6.2.1. O sistema exibe uma mensagem “Quantidade atingiu o estoque máximo!”</p> <p>6.2.2. O sistema redireciona a página para o passo 4</p>
-------------------	--

Tabela 7: Caso de Uso Saída de Produto.

Caso de Uso	UC 06 – Saída de Produto
Descrição	Efetua o cadastro da saída do produto no sistema
Pré-Condições	UC 01 – Efetuar Login
Pós-Condições	Administrar o acesso das saídas dos produtos
Fluxo Principal	<p>1. O usuário clica no link chamado “Saída” no menu vertical do lado esquerdo</p> <p>2. O sistema exibe uma relação de produtos que saíram do estoque</p> <p>3. O usuário clica no botão de cadastrar</p> <p>4. O sistema exibe a página de cadastro de saída de produto</p> <p>5. O usuário preenche os campos que são obrigatórios</p> <p>6. O servidor recebe a requisição dos dados do formulário</p> <p>7. O servidor armazena os dados no banco de dados</p> <p>8. O sistema exibe uma mensagem de “Cadastro incluído com sucesso!”</p> <p>9. O sistema redireciona para a página do passo 2</p>

Fluxo Alternativo	<p>5.1. O usuário não preenche todos os campos</p> <p>5.1.1. O sistema redireciona a página para o passo 4</p> <p>6.1. O servidor verifica que o produto não foi cadastrado</p> <p>6.1.1. O sistema exibe uma mensagem “Produto não cadastrado!”</p> <p>6.1.2. O sistema redireciona a página para o passo 4</p> <p>6.2. O servidor verifica que a matrícula não foi cadastrada</p> <p>6.2.1. O sistema exibe uma mensagem “Matrícula não cadastrado!”</p> <p>6.2.2. O sistema redireciona a página para o passo 4</p> <p>6.3. O servidor verifica que a quantidade total do produto é menor que zero</p> <p>6.3.1. O sistema exibe uma mensagem “Quantidade de estoque abaixo de zero!”</p> <p>6.3.2. O sistema redireciona a página para o passo 4</p> <p>7.1. O servidor verifica que a quantidade total do produto é menor ou igual a quantidade de estoque mínimo</p> <p>7.1.1. O sistema envia um e-mail automaticamente para o gerente, solicitando a compra do produto</p> <p>7.1.2. O sistema redireciona a página para o passo 4</p>
-------------------	--

Tabela 8: Caso de Uso Pedido de Compra.

Caso de Uso	UC 07 – Pedido de Compra
Descrição	Efetua o cadastro do pedido de compra no sistema
Pré-Condições	UC 01 – Efetuar Login
Pós-Condições	Administrar o acesso dos pedidos de compra
Fluxo Principal	<p>1. O usuário clica no link chamado “Pedido de Compra” no menu vertical do lado esquerdo</p> <p>2. O sistema exibe uma relação de produtos que fizeram o pedido de compra</p>

	<p>3. O usuário clica no botão de cadastrar</p> <p>4. O sistema exibe a página de cadastro de pedido de compra</p> <p>5. O usuário preenche os campos que são obrigatórios</p> <p>6. O servidor recebe a requisição dos dados do formulário</p> <p>7. O servidor armazena os dados no banco de dados</p> <p>8. O sistema envia um e-mail automaticamente para o gerente, solicitando o pedido de compra</p> <p>9. O sistema exibe uma mensagem de “Cadastro incluído com sucesso!”</p> <p>10. O sistema redireciona para a página do passo 2</p>
Fluxo Alternativo	<p>5.1. O usuário não preenche todos os campos</p> <p>5.1.1. O sistema redireciona a página para o passo 4</p> <p>6.1. O servidor verifica que o produto não foi cadastrado</p> <p>6.1.1. O sistema exibe uma mensagem “Produto não cadastrado!”</p> <p>6.1.2. O sistema redireciona a página para o passo 4</p> <p>6.2. O servidor verifica que o nome do produto é diferente do código do produto informado no formulário</p> <p>6.2.1. O sistema exibe uma mensagem “Nome do produto diferente do código produto cadastrado!”</p> <p>6.2.2. O sistema redireciona a página para o passo 4</p> <p>6.3. O servidor verifica que o fornecedor não foi cadastrado</p> <p>6.3.1. O sistema exibe uma mensagem “Fornecedor não cadastrado!”</p> <p>6.3.2. O sistema redireciona a página para o passo 4</p> <p>6.4. O servidor verifica que a categoria informada é diferente da que foi cadastrada</p> <p>6.4.1. O sistema exibe uma mensagem “Categoria diferente da cadastrada!”</p> <p>6.4.2. O sistema redireciona a página para o passo 4</p> <p>6.5. O servidor verifica que a quantidade total é maior que a estoque máximo cadastrado</p>

	<p>6.5.1. O sistema exibe uma mensagem “Quantidade é maior que o estoque máximo!”</p> <p>6.5.2. O sistema redireciona a página para o passo 4</p> <p>6.6. O servidor verifica que a data de reposição é menor que a data limite de entrega estabelecido</p> <p>6.6.1. O sistema exibe uma mensagem “Data de Reposição é menor que data informada!”</p> <p>6.6.2. O sistema redireciona a página para o passo 4</p>
--	--

Tabela 9: Caso de Uso Relatório.

Caso de Uso	UC 08 – Relatório
Descrição	Visualiza o relatório dos produtos que estão no estoque
Pré-Condições	UC 01 – Efetuar login
Pós-Condições	Visualizar relatório
Fluxo Principal	<p>1. O usuário clica no link chamado “Relatório” no menu vertical do lado esquerdo</p> <p>2. O sistema exibe uma relação de produtos cadastrados e a quantidade em estoque</p>
Fluxo Alternativo	2.1. O usuário pode baixar o relatório no formato excel

Tabela 10: Caso de Uso Logout.

Caso de Uso	UC 09 – Fazer Logout
Descrição	Sair da sessão
Pré-Condições	UC 01 – Efetuar login
Pós-Condições	Sessão é finalizada
Fluxo Principal	1. O usuário clica no botão sair na barra de menu do lado superior direito da tela

	<p>2. O sistema exibe uma mensagem de “Deseja realmente sair do sistema?” com um botão de não ou sim</p> <p>3. O usuário aperta sim</p> <p>4. O sistema encerra a sessão</p> <p>5. O sistema redireciona para a página de Login</p>
Fluxo Alternativo	<p>2.1. O usuário clica no botão não</p> <p>2.2. O sistema redireciona para a página de estoque</p>

Tabela 11: Caso de Uso Alteração de Senha.

Caso de Uso	UC 10 – Alterar Senha
Descrição	Efetua a alteração da senha
Pré-Condições	Matrícula e senha válidos
Pós-Condições	Entrar no sistema
Fluxo Principal	<p>1. O usuário clica no link chamado “Alterar Senha” na página de login</p> <p>2. O sistema exibe a página de alterar senha</p> <p>3. O usuário preenche os campos que são obrigatórios</p> <p>4. O servidor recebe a requisição dos dados do formulário</p> <p>5. O servidor armazena os dados no banco de dados</p> <p>6. O sistema exibe uma mensagem de “Senha alterada com sucesso!”</p> <p>7. O sistema redireciona para a página no passo 1</p>
Fluxo Alternativo	<p>3.1. O usuário não preenche todos os campos</p> <p>3.1.1. O sistema redireciona a página para o passo 2</p> <p>4.1. O servidor verifica que a matrícula não foi cadastrada</p> <p>4.1.1. O sistema exibe uma mensagem “Usuário não cadastrado!”</p> <p>4.1.2. O sistema redireciona a página para o passo 2</p> <p>4.2. O servidor verifica que a senha informada não é a mesma cadastrada</p>

	<p>4.2.1. O sistema exibe uma mensagem “Senha não confere!”</p> <p>4.2.2. O sistema redireciona a página para o passo 2</p> <p>4.3. O servidor verifica que a senha e confirmação de senha não são iguais</p> <p>4.3.1. O sistema exibe uma mensagem “Senha e confirmação de senha não são iguais!”</p> <p>4.3.2. O sistema redireciona a página para o passo 2</p>
--	---