



Universidade  
Veiga de Almeida

Sistemas de Informação – Noite

Prof. ADRIANA APARICIO SICSÚ AYRES DO NASCIMENTO

**GRUPO:**

Bianca Machado Mariano  
Daniel Tibério da Silva Dourado  
Igor de Freitas carneiro da silva  
João Pedro Callipo Vilhena

PROJETO MAKER

# **LOJA GEEKPOWER**

A4 – algoritmo e laboratório de programação

Rio de Janeiro, Junho de 2020

# Índice

<b>INTRODUÇÃO.....</b>	<b>1</b>
1. O que é E-commerce?.....	1
1.2. E-Commerce X Loja Virtual.....	2
1.3. Tipos de E-commerce.....	2
 <b>2.O Projeto Maker.....</b>	<b>3</b>
2.1 O Manual.....	3
2.2. Protótipo do projeto.....	4
2.2.1. Ferramentas utilizadas para o desenvolvimento do protótipo.....	4
2.2.2. Funcionalidades referentes ao Carrinho de Compra.....	4
2.2.3. Breve descrição de como funcionam as funcionalidades do protótipo.....	7
2.2.4. Trecho do código em C das funções desenvolvidas das funcionalidades.....	8
2.2.5. Layout do Protótipo.....	11
2.3. O Código.....	13
2.3.1. Desenvolvimento em C.....	13
2.3.2. Print das telas de testes das principais funcionalidades do carrinho de compra.....	19
 <b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>20</b>

# INTRODUÇÃO

O mundo está mudando, e com ele, a maneira como as pessoas compram produtos e serviços. Não é mais necessário uma loja física para se obter um bem ou um serviço. Ter uma loja virtual corta custos enormes e diminui a quantidade de pessoas necessárias para gerenciar um projeto. Com o advento da internet e consequentemente do E-commerce nos anos 90, a forma como as pessoas lidam com o mercado foi transformada completamente e para sempre. As fronteiras entre o virtual e o físico foram redefinidas e por consequência as relações humanas.

O presente projeto tem como objetivo explorar o universo do E-commerce e focar na construção de uma loja virtual na linguagem de programação C.

Na primeira etapa, conceitos serão definidos e comparados, a origem nacional e internacional desta modalidade de comércio será explicada, os processos logísticos explorados, além de descrever formas de pagamento que podem ser utilizados, as funções dos colaboradores em uma empresa de E-commerce, o que são canais do E-commerce, a diferença entre E-commerce e loja virtual, tipos de E-commerce, e finalmente problemas e limitações do E-commerce.

Na segunda etapa, mostraremos o desenvolvimento de nossa loja de E-commerce em C com todas as suas informações: tipo de E-commerce, nome da loja, tema, conteúdo, logomarca, público alvo, etc.

## 1. O que é E-commerce?

Existem algumas definições diferentes sobre o que um E-commerce é exatamente. A definição do termo é em sentido amplo “o uso de redes de computadores para melhorar o desempenho organizacional” (Zattar & Stefano, 2016, p10). Para os autores, um E-commerce é um “fenômeno de negócios” baseado em aplicações web potencializado pela chamada *nova economia*, ou seja, envolve TI para a melhora de comunicações e transações com todas as partes de uma empresa: instituições financeiras, o público em geral, clientes, fornecedores, colaboradores, etc.

Pode-se dizer também que um E-commerce é uma combinação de plataformas, tecnologias, marketing, redes sociais e pessoas, e que na prática, E-commerce é vender um produto ou serviço pela internet (Junior, 2019), ou seja, são basicamente transações comerciais realizadas totalmente online.

Para Turban & King (2003), E-commerce é também conhecido como E-business, e é definido como “a maneira como transações ocorrem através de redes”, principalmente a internet(...).<sup>\*</sup> É o processo de comprar e vender eletronicamente bens, serviços e informação (p8). Já Chaffey (2003) define E-commerce como todos os tipos de transações eletrônicas entre as organizações e os stakeholders, sejam eles transações financeiras ou trocas de informação ou outros serviços, ou seja, também inclui atividades de pré-venda e também de pós-venda ao longo da cadeia de suprimentos.

Podemos observar que as definições são bem parecidas e costumam concordar em alguns pontos: E-commerce é uma nova forma de fazer negócios por meio da internet, e vai além de um simples processo de compra e venda de bens e serviços.

## 1.2. E-Commerce X Loja Virtual

Enquanto o E-commerce engloba todo o comércio de mercadorias e serviços realizados no ambiente virtual, a Loja Virtual é o site de um vendedor (LOMBARDI, 2019). Ou seja, o E-commerce pode ser uma loja virtual, um Marketplace (Mercado Livre, Amazon), e hoje é até possível fazer comércio por redes sociais (Facebook, Instagram, Whatsapp, LinkedIn). (poderia escrever isso: sendo hoje possível fazer comércio por redes sociais (Facebook, Instagram, Whatsapp, LinkedIn))

Já numa Loja virtual você pode expor os produtos para que o cliente escolha o que comprar, como uma vitrine, para o consumidor selecionar o que deseja, colocar no carrinho de compras e concluir a compra. (como numa vitrine: O consumidor seleciona o que deseja, coloca no carrinho de compras e conclui a compra)

Podemos então enxergar claramente que a diferença entre os dois é que a loja virtual é apenas uma das formas de vender online. Já o E-commerce engloba várias formas de vender online.

## 1.3. Tipos de E-commerce

Existe uma classificação de tipos de E-commerce diferenciados pela natureza da transação e pelo relacionamento entre os participantes (TURBAN & KING, 2003). Os mais comumente encontrados são:

- ❖ Business-to-business (B2B): Todos os participantes são empresas ou organizações. A Maior parte do comércio eletrônico hoje é do tipo B2B.
- ❖ Business-to-consumer (B2C): São transações de varejo entre empresas e pessoas físicas. Outro nome pra esse tipo de E-commerce é varejo eletrônico.
- ❖ Business-to-Business-to-consumer (B2B2C): É quando uma empresa oferece produtos a uma empresa que seja sua cliente, para esta oferecer estes produtos para o consumidor final. Hoje a maioria das grandes empresas empregam o B2B2C, como Amazon, Submarino, Americanas.
- ❖ Mobile commerce (M-commerce): Transações de e-commerce realizadas em ambientes sem fio.
- ❖ Consumer-to-consumer (C2C): Consumidores vendem diretamente uns aos outros. Exemplos são o mercadolivre.com e enjoei.com.

## 2.O Projeto Maker

### 2.1 O Manual

**NOME DA EMPRESA:** GEEK POWER

**LOGOTIPO:**



**PÚBLICO ALVO:** O público-alvo da loja são as pessoas interessadas em jogos eletrônicos; gamers.

**TEMA:** Loja para venda de jogos eletrônicos, com uma abrangência de produtos que abarca diferentes plataformas e consoles como PC, Xbox, Playstation, Nintendo. A GeekPower não trabalha com a venda de jogos usados, apenas com a venda de jogos novos.

**TIPO:** **B2C:** As transações deste varejo eletrônico serão feitas diretamente com os clientes. A empresa terá um armazém e um controle de estoque próprio, com todo o processamento sendo feito na empresa, não terceirizando nada.

**ESTRUTURA DA PLATAFORMA:** A plataforma de compra será um *website* responsivo, ou seja, ele se ajustará à tela do cliente, seja num PC, tablet ou celular. O site também terá um aplicativo próprio, assim como a Submarino, o Mercado Livre e a Aliexpress tem. Dessa forma, o cliente terá mais facilidade para salvar itens na lista de desejos, consultar produtos, visualizar formas de pagamento, e acessar diariamente os descontos exclusivos que haverão na plataforma.

**FORMAS DE PAGAMENTO:** A plataforma oferecerá como forma de pagamento cartão de crédito, débito, boleto bancário, PayPal e Ame. Será oferecido um desconto de 10% se o cliente escolher crédito à vista ou boleto bancário.

**PROCESSOS:** Após o pedido de compra ser efetuado, o número do pedido com as informações será enviada para a área de *picking* para que os itens possam ser coletados. Após a coleta, será emitida a nota fiscal, que será encaminhada para a área de conferência

para conferir se não há erros com o pedido. Caso tudo esteja correto, os itens e a nota fiscal são enviadas para o *packing*, onde os produtos serão embalados e enviados para a transportadora. Assim que for enviado, o cliente receberá um código de rastreio por e-mail, e que também estará disponível na plataforma.

**POLÍTICA DE TROCA E DEVOLUÇÃO:** Caso haja insatisfação com o produto, existirá uma política de troca dentro de 7 dias de acordo com a lei. Em caso de avaria, o período de troca ou devolução será de 30 dias.

## 2.2. Protótipo do projeto

### 2.2.1. Ferramentas utilizadas para o desenvolvimento do protótipo.

Em nosso projeto utilizamos as seguintes ferramentas:

[fabricadeaplicativos.com.br](http://fabricadeaplicativos.com.br)

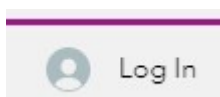
[www.wix.com](http://www.wix.com)

### 2.2.2. Funcionalidades referentes ao Carrinho de Compra.

Em nosso site <https://biancamariano.wixsite.com/website> você pode simular um processo de compras. Abaixo citamos as principais:

1. Fazer Login
2. Visualização Rápida
3. Adicionar no Carrinho
4. Visualizar Carrinho
5. Retirar do Carrinho
6. Cálculo do Frete
7. Fechar o Pedido
8. Sair

**Observação:** Nosso aplicativo é somente uma ilustração do que é apresentado em nosso site.



# Registre-se

Já é um membro? [Login](#)



Registre-se com Facebook



Registre-se com Google+

ou

Registre-se com seu email



Visualização Rápida

Kit Acessórios Esportivos 32 In 1 - Nintendo

R\$ 85,00



Kit Acessórios Esportivos 32 In 1 - Nintendo

R\$ 85,00 [Ver todos os detalhes](#)

Quantidade

1

ADICIONAR AO CARRINHO

Visualização Rápida

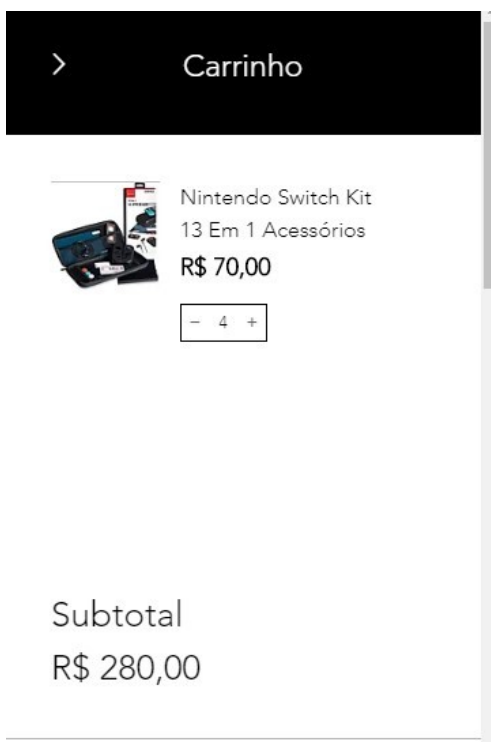
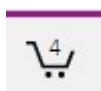
R\$ 70,00

Quantidade

4

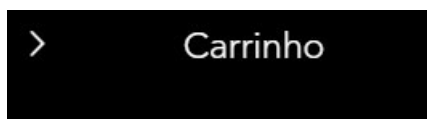
Adicionar ao Carrinho

Adicionar no Carrinho



Visualizar Carrinho





Nintendo Switch Kit  
13 Em 1 Acessórios  
R\$ 70,00



- 2 +

Subtotal  
R\$ 140,00

Retirar do Carrinho

#### Meu carrinho



Nintendo Switch Kit 13  
Em 1 Acessórios  
R\$ 70,00

4

R\$ 280,00



🔑 Insira o código promocional

📝 Adicione uma nota

#### Resumo do Pedido

Subtotal R\$ 280,00

[Rio de Janeiro, Brasil](#)

Total R\$ 280,00

🔒 Checkout

Fechar Pedido

### 2.2.3. Breve descrição de como funcionam as funcionalidades do protótipo.

1. Fazer Login - Pode ser através de suas redes sociais (Google ou Facebook) ou registrando-se com seu e-mail.
2. Visualização Rápida – Você tem a opção de olhar o produto antes mesmo de efetuar a compra, passando o cursor do mouse por cima da imagem é possível ver com mais detalhes pelo zoom.
3. Adicionar no Carrinho - Você define quantos itens deseja comprar
4. Visualizar Carrinho - Conferir se comprou o que precisava, se falta adicionar mais alguma coisa ou deseja retirar algo.
5. Retirar do Carrinho - Caso não tenha interesse no produto.

6. Cálculo do Frete - Saber quanto você pagará para o envio do produto para sua residência, trabalho, etc.
7. Fechar o Pedido - Finalizar seu pedido no site.
8. Sair - Sair da sua conta.

## 2.2.4. Trecho do código em C das funções desenvolvidas das funcionalidades.

### 1. Display da página de compras

```
void display()
{
    printf("GEEKPOWER - LOJA\n\n");
    printf("1. Minecraft - R$ 120.00\n");
    printf("2. WARFRAME - R$ 10.00\n");
    printf("3. PES2020 - R$ 199.99\n");
    printf("4. XBOX ONE S - R$ 1999.99\n");
    printf("5. Playstation 4 - R$ 2199.99\n");
    printf("6. Nintendo Switch - R$ 2699.99\n");
}
```

### 2. Cadastro de Produtos

```
void cadastrar() //Função cadastrar.
{
    printf("\nCADASTRO DE PRODUTOS GEEKSTORE\nPara cadastrar um produto novo, é necessário ter as seguintes informações:\nSKU\nNome do produto\nNúmero de unidades em estoque\nPreço base unitário\nDeseja cadastrar um novo produto? Digite S para sim, N para não: ");
    scanf("%s", &opera);
    while (strcmp(&opera, "S") == 0 || (strcmp(&opera, "s") == 0)) //Enquanto for S ou s a resposta de "quer continuar?", o código continuará repetindo a partir daqui
    {
        while (strcmp(&opera, "S") != 0 && (strcmp(&opera, "s") != 0) && (strcmp(&opera, "n") != 0) && (strcmp(&opera, "N") != 0))
        { //Enquanto o usuário não digitar S, s, n ou N,
            printf("\n Comando inválido. Deseja inserir outro cadastro? S/N: ");
            scanf("%s", &opera); // ele repete "Comando inválido" e pede pra reinserir.
        }
        if (strcmp(&opera, "S") == 0 || (strcmp(&opera, "s") == 0)) //Caso seja S ou s, ele pede as informações e armazena na posição atual de i (que é iniciada em -1, mas já é pré-incrementada para 0 na primeira execução)
        {
            i++;
            printf("\nDESCRIÇÃO: ");
            scanf("%s", &name[i]);
            printf("\nSKU: ");
            scanf("%d", &SKU[i]);
            printf("\nUNIDADES EM ESTOQUE: ");
            scanf("%d", &estoque[i]);
            printf("\nVALOR BASE DO PRODUTO: ");
```

```

        scanf("%f", &price[i]);
        printf("\n-----\n");
        printf("Deseja fazer mais um cadastro? Digite S para sim, ou N para voltar: ");
        scanf("%s", &opera);
    }
    else if (strcmp(&opera, "N") //Caso seja N ou n, ele sai.
        || (strcmp(&opera, "n")))
    {
        return;
    }
}
}

```

### 3. Consulta de Produtos

void consultar() //Função consultar.

```

{
    printf("\nCONSULTA DE PRODUTOS GEEKSTORE\nPara consultar informações
sobre um produto específico, é necessário possuir o SKU atrelado a este. Deseja
consultar algum produto? Digite S para sim, N para não: ");
    scanf("%s", &opera);
    while (strcmp(&opera, "S") == 0 || (strcmp(&opera, "s") == 0)) //Enquanto for S ou s a
resposta de "quer continuar?", o código continuará repetindo a partir daqui
    {
        while (strcmp(&opera, "S") != 0 && (strcmp(&opera, "s") != 0) && (strcmp(&opera,
"n") != 0) && (strcmp(&opera, "N") != 0))
        { //Enquanto o usuário não digitar S, s, n ou N,
            printf("\n Comando inválido. Deseja fazer uma nova consulta? S/N: ");
            scanf("%s", &opera); // ele repete "Comando inválido" e pede pra reinserir.
        }
        if (strcmp(&opera, "S") == 0 || (strcmp(&opera, "s") == 0)) //Caso seja S ou s, ele
pede o código SKU a ser pesquisado.
        {
            printf("\nDigite o código SKU a ser consultado: ");
            scanf("%d", &busca);
            for (j = 0; j <= 100; j++) //Ele busca de 1 a 100 nas posições por um SKU igual ao
fornecido,
            {
                if (SKU[j] == busca)
                {
                    posicao = j; //e caso encontre, ele armazena a posição deste
                }
            }
            //Resultado da busca
            if (posicao == -1) //Faz um check pra ver se ela foi preenchida/encontrada, caso
não tenha sido modificado o valor de "posição", ou seja, não tenha sido encontrado SKU
equivalente, ela será -1 e retornará "produto está indisponível".
            {
                printf("\nEste produto não se encontra disponível. ");
                printf("Deseja fazer mais uma consulta? Digite S para sim, N para não: ");
                scanf("%s", &opera);
            }
        }
    }
}

```

```

    else
    { //Com essa posição, ele acha a descrição, número de unidades e preço.
        printf("\nDescrição do produto: %s", &name[posicao]);
        printf("\nUnidades em estoque: %d", estoque[posicao]);
        printf("\nPreço base: %.2f\n", price[posicao]);
        printf("Deseja fazer mais uma consulta? Digite S para sim, N para não: ");
        scanf("%s", &opera);
    }
}
else if (strcmp(&opera, "N") //Caso seja N ou n, ele sai.
        || (strcmp(&opera, "n")))
{
    return;
};
}
}

```

#### 4. Carrinho da Loja

```

void carrinho()
{
    printf("\nCARRINHO GEEKSTORE\nPara adicionar ao carrinho, você deve ter em
    mente o código SKU do produto e a quantidade a ser adicionada.\n Deseja adicionar um
    produto novo ao carrinho? S para sim, N para não: ");
    scanf("%s", &opera);
    while (strcmp(&opera, "S") == 0 || (strcmp(&opera, "s") == 0)) //Enquanto for S ou s a
    resposta de "quer continuar?", o código continuará repetindo a partir daqui
    {
        while (strcmp(&opera, "S") != 0 && (strcmp(&opera, "s") != 0) && (strcmp(&opera,
        "n") != 0) && (strcmp(&opera, "N") != 0))
        { //Enquanto o usuário não digitar S, s, n ou N,
            printf("\n Comando inválido. Deseja inserir outro item no carrinho? S/N: ");
            scanf("%s", &opera); // ele repete "Comando inválido" e pede pra reinserir.
        }
        if (strcmp(&opera, "S") == 0 || (strcmp(&opera, "s") == 0)) //Caso seja S ou s, ele
        pede as informações e armazena na posição atual de i (que é iniciada em -1, mas já é
        pré-incrementada para 0 na primeira execução)
        {
            printf("\nCÓDIGO SKU: ");
            scanf("%d", &carrinhoPDT[item]);
            printf("\nQUANTIDADE DO PRODUTO: ");
            scanf("%d", &carrinhoQTD[item]);
            for (j = 0; j < 100; j++)
            {
                if (SKU[j] == carrinhoPDT[item])
                {
                    posicao = j;
                }
            }
        }
        printf("\n-----\n");
        printf("VALOR TOTAL DO CARRINHO: %.2f", (price[posicao] *
        carrinhoQTD[item]));
    }
}

```

```

        printf("\nDeseja adicionar outro produto ao carrinho? S para sim, N para finalizar
a compra: ");
        scanf("%s", &opera);
    }
    else if (strcmp(&opera, "N") //Caso seja N ou n, ele sai.
        || (strcmp(&opera, "n")))
    {
        return;
    }
}
}

```

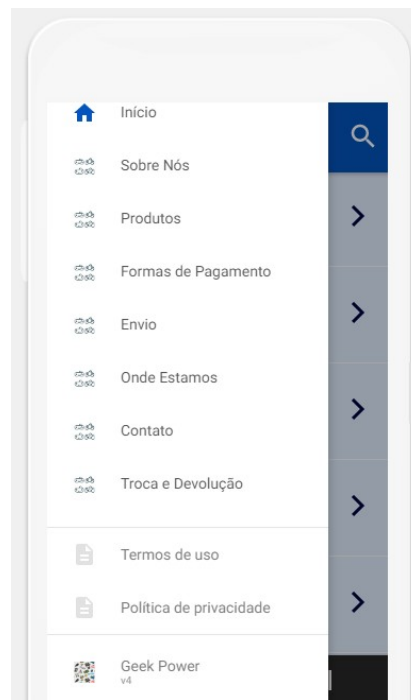
### 2.2.5. Layout do Protótipo

Link para visualização completa com todos os detalhes

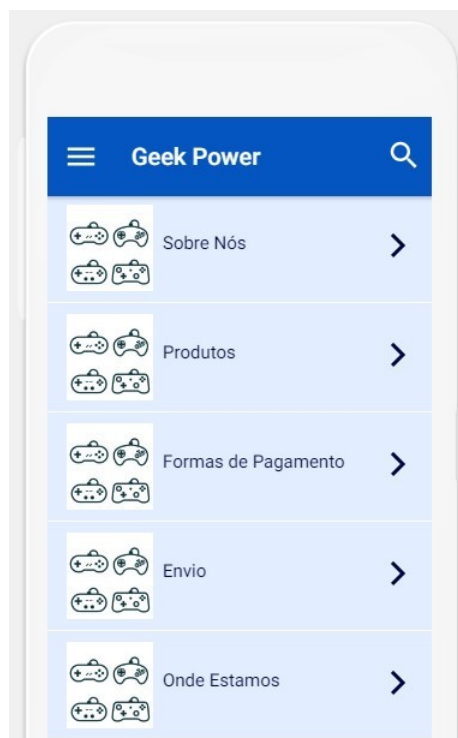
[https://galeria.fabricadeaplicativos.com.br/geek\\_power](https://galeria.fabricadeaplicativos.com.br/geek_power), abaixo só destacamos telas principais.



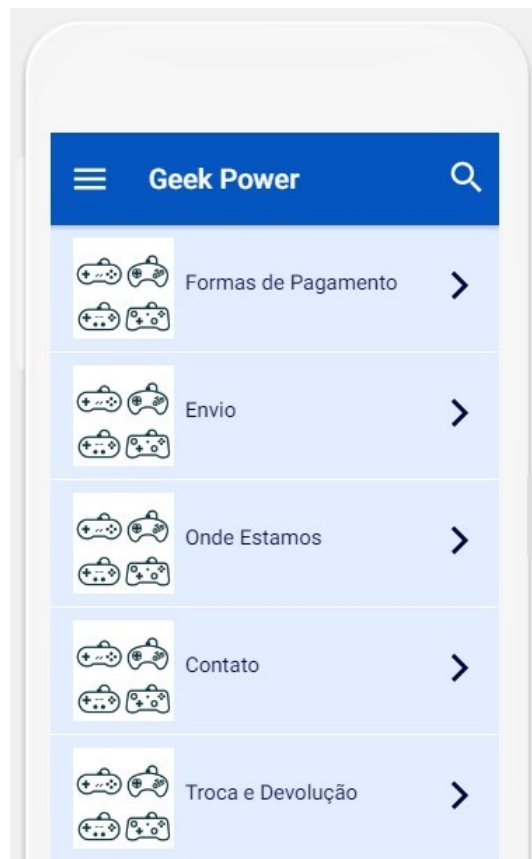
Tela Inicial



Menu 1



Menu 2 (continuação)



Menu 3 (continuação)

## 2.3. O Código

### 2.3.1. Desenvolvimento em C

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

// Variáveis a serem utilizadas
int SKU[100], estoque[100], i = -1, j, busca, posicao = -1, running, carrinhoPDT,
carrinhoQTD, item[100];
char opera, name[100];
float price[100], precototal;

//Display da página de compras
void display()
{
    printf("GEEKPOWER - LOJA\n\n");
    printf("1. Minecraft - R$ 120.00\n");
```

```

printf("2. WARFRAME - R$ 10.00\n");
printf("3. PES2020 - R$ 199.99\n");
printf("4. XBOX ONE S - R$ 1999.99\n");
printf("5. Playstation 4 - R$ 2199.99\n");
printf("6. Nintendo Switch - R$ 2699.99\n");
}

//Cadastro de Produtos
void cadastrar() //Função cadastrar.
{
    printf("\nCADASTRO DE PRODUTOS GEEKSTORE\nPara cadastrar um produto
    novo, é necessário ter as seguintes informações:\nSKU\nNome do produto\nNúmero de
    unidades em estoque\nPreço base unitário\nDeseja cadastrar um novo produto? Digite S
    para sim, N para não: ");
    scanf("%s", &opera);
    while (strcmp(&opera, "S") == 0 || (strcmp(&opera, "s") == 0)) //Enquanto for S ou s a
    resposta de "quer continuar?", o código continuará repetindo a partir daqui
    {
        while (strcmp(&opera, "S") != 0 && (strcmp(&opera, "s") != 0) && (strcmp(&opera,
        "n") != 0) && (strcmp(&opera, "N") != 0))
        { //Enquanto o usuário não digitar S, s, n ou N,
            printf("\n Comando inválido. Deseja inserir outro cadastro? S/N: ");
            scanf("%s", &opera); // ele repete "Comando inválido" e pede pra reinserir.
        }
        if (strcmp(&opera, "S") == 0 || (strcmp(&opera, "s") == 0)) //Caso seja S ou s, ele
        pede as informações e armazena na posição atual de i (que é iniciada em -1, mas já é
        pré-incrementada para 0 na primeira execução)
        {
            i++;
            printf("\nDESCRIÇÃO: ");
            scanf("%s", &name[i]);
            printf("\nSKU: ");
            scanf("%d", &SKU[i]);
            printf("\nUNIDADES EM ESTOQUE: ");
            scanf("%d", &estoque[i]);
            printf("\nVALOR BASE DO PRODUTO: ");
            scanf("%f", &price[i]);
            printf("\n-----\n");
            printf("Deseja fazer mais um cadastro? Digite S para sim, ou N para voltar: ");
            scanf("%s", &opera);
        }
    }
}

```



```

    }
    else if (strcmp(&opera, "N") //Caso seja N ou n, ele sai.
        || (strcmp(&opera, "n")))
    {
        return;
    }
}
}

// Consulta de Produtos
void consultar() //Função consultar.
{
    printf("\nCONSULTA DE PRODUTOS GEEKSTORE\nPara consultar informações
sobre um produto específico, é necessário possuir o SKU atrelado a este. Deseja
consultar algum produto? Digite S para sim, N para não: ");
    scanf("%s", &opera);
    while (strcmp(&opera, "S") == 0 || (strcmp(&opera, "s") == 0)) //Enquanto for S ou s a
resposta de "quer continuar?", o código continuará repetindo a partir daqui
    {
        while (strcmp(&opera, "S") != 0 && (strcmp(&opera, "s") != 0) && (strcmp(&opera,
"n") != 0) && (strcmp(&opera, "N") != 0))
        { //Enquanto o usuário não digitar S, s, n ou N,
            printf("\n Comando inválido. Deseja fazer uma nova consulta? S/N: ");
            scanf("%s", &opera); // ele repete "Comando inválido" e pede pra reinserir.
        }
        if (strcmp(&opera, "S") == 0 || (strcmp(&opera, "s") == 0)) //Caso seja S ou s, ele
pede o código SKU a ser pesquisado.
        {
            printf("\nDigite o código SKU a ser consultado: ");
            scanf("%d", &busca);
            for (j = 0; j <= 100; j++) //Ele busca de 1 a 100 nas posições por um SKU igual ao
fornecido,
            {
                if (SKU[j] == busca)
                {
                    posicao = j; //e caso encontre, ele armazena a posição deste
                }
            }
            //Resultado da busca
            if (posicao == -1) //Faz um check pra ver se ela foi preenchida/encontrada, caso

```

não tenha sido modificado o valor de "posição", ou seja, não tenha sido encontrado SKU equivalente, ela será -1 e retornará "produto está indisponível".

```

    {
        printf("\nEste produto não se encontra disponível. ");
        printf("Deseja fazer mais uma consulta? Digite S para sim, N para não: ");
        scanf("%s", &opera);
    }
    else
    { //Com essa posição, ele acha a descrição, número de unidades e preço.
        printf("\nDescrição do produto: %s", &name[posicao]);
        printf("\nUnidades em estoque: %d", estoque[posicao]);
        printf("\nPreço base: %.2f\n", price[posicao]);
        printf("Deseja fazer mais uma consulta? Digite S para sim, N para não: ");
        scanf("%s", &opera);
    }
}
else if (strcmp(&opera, "N") //Caso seja N ou n, ele sai.
        || (strcmp(&opera, "n")))
{
    return;
};
}
}

// Carrinho da Loja
void carrinho()
{
    printf("\nCARRINHO GEEKSTORE\nPara adicionar ao carrinho, você deve ter em mente o código SKU do produto e a quantidade a ser adicionada.\n Deseja adicionar um produto novo ao carrinho? S para sim, N para não: ");
    scanf("%s", &opera);
    while (strcmp(&opera, "S") == 0 || (strcmp(&opera, "s") == 0)) //Enquanto for S ou s a resposta de "quer continuar?", o código continuará repetindo a partir daqui
    {
        while (strcmp(&opera, "S") != 0 && (strcmp(&opera, "s") != 0) && (strcmp(&opera, "n") != 0) && (strcmp(&opera, "N") != 0))
        { //Enquanto o usuário não digitar S, s, n ou N,
            printf("\n Comando inválido. Deseja inserir outro item no carrinho? S/N: ");
            scanf("%s", &opera); // ele repete "Comando inválido" e pede pra reinserir.
        }
    }
}

```

if (strcmp(&opera, "S") == 0 || (strcmp(&opera, "s") == 0)) //Caso seja S ou s, ele pede as informações e armazena na posição atual de i (que é iniciada em -1, mas já é pré-incrementada para 0 na primeira execução)

```
{
    printf("\nCÓDIGO SKU: ");
    scanf("%d", &carrinhoPDT[item]);
    printf("\nQUANTIDADE DO PRODUTO: ");
    scanf("%d", &carrinhoQTD[item]);
    for (j = 0; j < 100; j++)
    {
        if (SKU[j] == carrinhoPDT[item])
        {
            posicao = j;
        }
    }
    printf("\n-----\n");
    printf("VALOR TOTAL DO CARRINHO: %.2f", (price[posicao] *
carrinhoQTD[item]));
    printf("\nDeseja adicionar outro produto ao carrinho? S para sim, N para finalizar
a compra: ");
    scanf("%s", &opera);
}
else if (strcmp(&opera, "N") //Caso seja N ou n, ele sai.
|| (strcmp(&opera, "n")))
{
    return;
}
}
```

int main(void)

```
{
    printf("Este programa consiste numa frente de e-commerce para uma loja digital de
foco em jogos eletrônicos.\nAqui será demonstrado o controle de estoque: adicionando
produtos novos na loja, checando o estoque existente, e o protótipo de um carrinho
virtual para executar compras no ambiente digital.\nDigite 1 para abrir a loja e ver a lista
de produtos já cadastrados.\nDigite 2 para cadastrar um produto novo.\nDigite 3 para
consultar um produto no estoque.\nDigite 4 para abrir o carrinho atual.\nDigite 5 para
sair.\n ");
    scanf("%d", &running);
```

```

while (running != 5)
{
    if (running == 1)
    {
        display();
        printf("\nDeseja:\n(1) abrir o menu de produtos, \n(2) cadastrar um novo produto, \
n(3) consultar o estoque, \n(4) abrir o carrinho, \n(5) ou sair?\n");
        scanf("%d", &running);
    }
    else if (running == 2)
    {
        cadastrar();
        printf("\nDeseja:\n(1) abrir o menu de produtos, \n(2) cadastrar um novo produto, \
n(3) consultar o estoque, \n(4) abrir o carrinho, \n(5) ou sair?\n");
        scanf("%d", &running);
    }
    else if (running == 3)
    {
        consultar();
        printf("\nDeseja:\n(1) abrir o menu de produtos, \n(2) cadastrar um novo produto, \
n(3) consultar o estoque, \n(4) abrir o carrinho, \n(5) ou sair?\n");
        scanf("%d", &running);
    }
    else if (running == 4)
    {
        carrinho();
        printf("\nDeseja:\n(1) abrir o menu de produtos, \n(2) cadastrar um novo produto, \
n(3) consultar o estoque, \n(4) abrir o carrinho, \n(5) ou sair?\n");
        scanf("%d", &running);
    }
    else
    {
        running = 0;
        printf("\nDeseja:\n(1) abrir o menu de produtos, \n(2) cadastrar um novo produto, \
n(3) consultar o estoque, \n(4) abrir o carrinho, \n(5) ou sair?\n");
        scanf("%d", &running);
    }
}
return 0;
}

```

## 2.3.2. Print das telas de testes das principais funcionalidades do carrinho de compra

### CADASTRO DE PRODUTOS

```
CADASTRO DE PRODUTOS GEEKSTORE
Para cadastrar um produto novo, é necessário ter as seguintes i
nformações:
SKU
Nome do produto
Número de unidades em estoque
Preço base unitário
Deseja cadastrar um novo produto? Digite S para sim, N para não
: s

DESCRIÇÃO: playstation4

SKU: 1

UNIDADES EM ESTOQUE: 10

VALOR BASE DO PRODUTO: 2000

-----
Deseja fazer mais um cadastro? Digite S para sim, ou N para vol
tar: █
```

### CONSULTA DE PRODUTOS

```
CONSULTA DE PRODUTOS GEEKSTORE
Para consultar informações sobre um produto específico, é neces
sário possuir o SKU atrelado a este. Deseja consultar algum pro
duto? Digite S para sim, N para não: s

Digite o código SKU a ser consultado: 1

Descrição do produto: playstation4
Unidades em estoque: 10
Preço base: 2000.00
Deseja fazer mais uma consulta? Digite █ para sim, N para não:
```

### CARRINHO DE COMPRAS

```
CARRINHO GEEKSTORE
Para adicionar ao carrinho, você deve ter em mente o código SKU
do produto e a quantidade a ser adicionada.
Deseja adicionar um produto novo ao carrinho? S para sim, N pa
ra não: s

CÓDIGO SKU: 1

QUANTIDADE DO PRODUTO: 5

-----
VALOR TOTAL DO CARRINHO: 10000.00
Deseja adicionar outro produto ao carrinho? S para sim, N para
finalizar a compra: █
```

## REFERÊNCIAS BIBLIOGRÁFICAS

CHAFFEY, Dave. **Gestão de E-Business e E-Commerce**: estratégia, implementação e prática. 5. ed. Brasil: Editora Elsevier, 2003. 552 p.

LOMBARDI, Lucas. **Qual é a diferença entre Ecommerce e loja virtual?** 2019. Disponível em: <https://ecommercedesucesso.com.br/diferenca-entre-ecommerce-e-loja-virtual/>. Acesso em: 07 jun. 2020.

MAURICI JUNIOR,. **O QUE é E-commerce? Como Vender pela Internet?**. 2019. Elaborado por Comschool. Disponível em: <https://www.youtube.com/watch?v=4juR0bWq1Bk>. Acesso em: 06 jun. 2020.

TURBAN, Efraim; KING, David. **Comércio Eletrônico**: estratégia e gestão. Brasil: Editora Pearson, 2003. 480 p.

ZATTAR, Izabel Cristina; STEFANO, Nara. **E-commerce**: conceitos, implementação e gestão. Brasil: Editora Intersaberes, 2016. 310 p.