

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS**  
**PUC Minas Virtual**  
**Pós-graduação *Lato Sensu* em Engenharia de *Software***

Projeto Integrado

Relatório Técnico

BuyBye Store

Lucas Gabriel Scaramelo

Belo Horizonte  
Agosto 2022.



# Projeto Integrado

## *Sumário*

Projeto Integrado	3
1. Cronograma de Trabalho	4
2. Introdução	6
3. Definição Conceitual da Solução	7
3.1 Diagrama de Casos de Uso	7
3.2 Requisitos Funcionais	8
3.3 Requisitos Não-funcionais	8
4. Protótipo Navegável do Sistema	9
5. Diagrama de Classes de Domínio	10
6. Arquitetura da Solução	11
6.1 Padrão Arquitetural	11
6.2 C4 model - Diagrama de Contexto	12
7. Frameworks de Trabalho	12
8. Estrutura Base do Front End	13
9. Modelo Relacional ou Projeto de Banco de Dados NoSQL	14
10. Plano de Testes	15
11. Relatório de Execução de Testes de Software	15
12. Apropriação de horas no projeto	16
13. Código da Aplicação	18
14. Avaliação Retrospectiva	18
14.1 Objetivos Estimados	18
14.2 Objetivos Alcançados	18
14.3 Lições aprendidas	18
15. Referências	19

## 1. Cronograma de Trabalho

Datas		Atividade / Tarefa	Produto / Resultado
De	Até		
01/05/2022	15/05/2022	1. Definir escopo do projeto	Requisitos funcionais e não funcionais
16/05/2022	31/05/2022	2. Diagrama de caso de uso	Desenho do diagrama de caso de uso
01/06/2022	10/06/2022	3. Diagrama de classes de domínio	Desenho do digrama de classes de domínio
11/06/2022	15/06/2022	4. Escrita da introdução do projeto	Introdução e objetivos do projeto desenvolvido
16/06/2022	31/07/2022	5. Protótipo navegável do sistema	Protótipo do sistema e vídeo de apresentação
01/08/2022	13/08/2022	6. Revisão dos artefatos da primeira entrega	Artefatos revistos e corrigidos se necessário
14/08/2022	15/08/2022	7. Finalização e entrega da etapa 1	Escopo da etapa 1 entregue
01/09/2022	05/09/2022	8. Definição da arquitetura utilizada junto a tecnologia e frameworks	Definir arquitetura que será usada para desenvolvimento
10/09/2022	20/09/2022	9. Diagrama de contexto arquitetura	Diagrama arquitetural da solução
21/09/2022	25/09/2022	10. Desenvolvimento das classes de domínio	Backend dos models desenvolvidos
26/09/2022	30/09/2022	11. Desenvolvimento da estrutura base(mestre) do frontend de administradores	Layout base do frontend de administradores desenvolvido
01/10/2022	05/10/2022	12. Desenvolvimento da estrutura base(mestre) do frontend de clientes	Layout base do frontend de clientes desenvolvido
06/10/2022	09/10/2022	13. Desenvolver controllers de acesso as telas de administrador	Controllers desenvolvidos para poder acessar cada tela
10/10/2022	11/10/2022	14. Desenvolver controllers de acesso as telas de cliente	Controllers desenvolvidos para poder acessar cada tela
12/10/2022	13/10/2022	15. Desenvolver função para perfil administrador cadastrar produtos	Funcionalidade que permite o adm cadastrar produtos novos
14/10/2022	15/10/2022	16. Finalização e entrega da etapa 2	Escopo da etapa 2 entregue
20/10/2022	25/10/2022	17. Modelo relacional MER	Diagrama MER
26/10/2022	31/10/2022	18. Plano e relatório de testes	Plano de execução de testes
01/11/2022	06/11/2022	19. Desenvolver função para perfil cliente de pesquisar produtos cadastrados	Backend e frontend desenvolvidos para pesquisa de produtos
07/11/2022	10/11/2022	20. Desenvolver função para perfil cliente de adicionar produtos no carrinho	Backend e frontend desenvolvidos para adicionar novos produtos
11/11/2022	16/11/2022	21. Desenvolver função para perfil cliente de finalizar pedido com pagamento	Backend e frontend desenvolvidos para concluir pedido

Projeto Integrado – Engenharia de *Software* - PMV

17/11/2022	20/11/2022	22. Desenvolver função para perfil cliente para alterar seus dados	Backend e frontend desenvolvidos para alteração de dados
21/11/2022	22/11/2022	23. Desenvolver função para perfil administrador cadastrar nova marca	Backend e frontend desenvolvidos para cadastro de nova marca
23/11/2022	25/11/2022	24. Desenvolver função para perfil administrador cadastrar nova categoria	Backend e frontend desenvolvidos para cadastro de nova categoria
26/11/2022	29/11/2022	25. Desenvolver função para perfil administrador cadastrar novo modelo	Backend e frontend desenvolvidos para cadastro de novo modelo
30/11/2022	02/12/2022	26. Desenvolver função para perfil cliente verificar histórico de seus pedidos	Backend e frontend desenvolvidos para visualizar histórico de pedidos
03/12/2022	06/12/2022	27. Desenvolver função para perfil cliente alterar ou excluir o pedido	Backend e frontend desenvolvidos para exclusão ou alteração de pedidos
07/12/2022	11/12/2022	28. Testes unitários do sistema	Todos os módulos com camadas aceitáveis de cobertura de testes
12/12/2022	13/12/2022	29. Criar vídeo final com apresentação do projeto	Vídeo com visão geral e final do projeto desenvolvido
14/12/2022	14/12/2022	30. Avaliação retrospectiva	Descrição dos objetivos alcançados no desenvolvimento geral do projeto
14/12/2022	15/12/2022	31. Finalização e entrega da etapa 3	Escopo da etapa 3 entregue

*Observação: acrescente ou retire linhas, caso seja necessário.*

## **2. Introdução**

Sabemos que devido a pandemia as pessoas do mundo inteiro entraram em uma situação extremamente delicada, fazendo com que muitas medidas invasivas fossem tomadas e aderidas de forma inegociável. Uma delas foi sua forma de comprar produtos. Pessoas que nunca antes tinham utilizado a internet para realizar compras, passaram a utilizar e aquele preconceito com compras online foi de certa forma sendo mitigado.

Atualmente o comércio eletrônico teve uma alta de 48,41% em comparação ao ano de 2020, isso nos mostra que o mercado digital tende a crescer cada vez mais, principalmente com os altos investimentos de empresas na experiência de seus clientes, atraindo a cada ano mais e mais pessoas para este universo.

Podemos destacar uma pesquisa recente do veículo de comunicação “Consumidor Moderno” que levantou um dado onde metade dos brasileiros preferem realizar compras online e abrem mão do modelo físico pelo conforto e praticidade, além de valores diferenciados pela isenção de taxas de vendedores/funcionários. É nítido que a cada dia que passa as pessoas estão querendo cada vez mais resolver seus problemas sem saírem de suas casas, fugindo de filas, procedimentos burocráticos, dores de cabeça no geral e sem falar nos quesitos de segurança.

Dados os fatores apresentados acima, decidimos criar um e-commerce de vendas com objetivo de trazer a outra metade de nossa população para esse universo de forma totalmente segura. Além de levar o conforto para dentro de casas, levamos qualidade, segurança e uma variedade enorme de produtos ofertados. Com suporte 24h por dia, 100% de nossos produtos são oferecidos por nossa plataforma, assim não gerando dependência de fornecedores terceiros, ou seja, a relação do cliente é diretamente conosco. Com objetivo de fidelizar nossos usuários, para nós todo dia é dia do consumidor, com ofertas diárias exclusivas.

O objetivo desse trabalho é atrair a outra metade da população para o mundo dos comércios eletrônicos, potencializando cada vez mais o mercado e atraindo novos investidores, além de levar segurança e qualidade para dentro de todas as casas.

Os objetivos específicos são:

- Desenvolver um ecommerce que possa atender a todos os nichos.

- Entregar uma aplicação web que implemente casos de uso de um e-commerce proposto na definição da solução.
- Liberdade do cliente em realizar qualquer tipo de compra de maneira segura dentro da plataforma e manter uma relação direta com o fornecedor

Referências introdutórias:

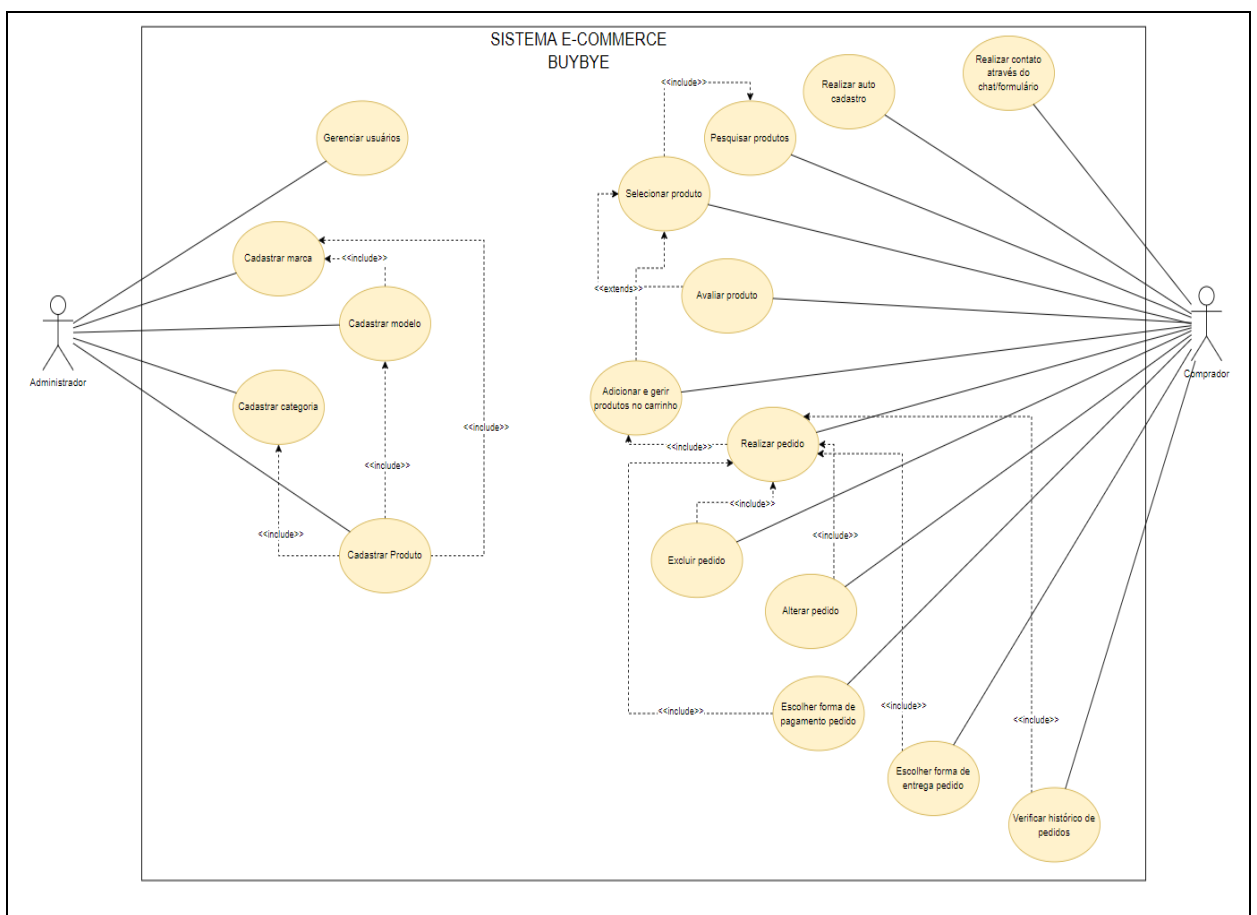
<https://www.conversion.com.br/blog/relatorio-ecommerce-mensal/>

<https://olhardigital.com.br/2022/02/25/tira-duvidas/aumento-das-compras-online-impacta-na-evolucao-da-logistica-brasileira/>

<https://www.consumidormoderno.com.br/2021/09/24/metade-brasileiros-prefere-lojas-online/>

### 3. Definição Conceitual da Solução

#### 3.1 Diagrama de Casos de Uso



### 3.2 Requisitos Funcionais

ID	Descrição Resumida	Dificuldade (B/M/A)*	Prioridade (B/M/A)*
RF01	O comprador deve conseguir se auto cadastrar no sistema	B	A
RF02	O comprador deve ser capaz de se autenticar no sistema	B	A
RF03	O comprador deve conseguir alterar sua senha em caso de esquecimento	B	M
RF04	O comprador deve ser capaz de pesquisar produtos por nome	M	A
RF05	O comprador deve ser capaz de pesquisar produtos por marca	M	M
RF06	O comprador deve ser capaz de pesquisar produtos por categoria	M	M
RF07	O comprador deve ser capaz de pesquisar produtos por modelo	M	M
RF08	O comprador deve ser capaz de selecionar qualquer produto para verificar seus dados	M	A
RF09	O comprador poderá inserir o produto em seu carrinho	A	A
RF10	O comprador poderá esvaziar seu carrinho	M	A
RF11	O comprador poderá efetuar pedidos	A	A
RF12	O comprador poderá alterar pedidos	A	A
RF13	O comprador poderá excluir pedidos	A	A
RF14	O comprador deve ser capaz de consultar seu histórico de pedidos	B	M
RF15	O comprador poderá ver detalhes de seus pedidos	M	M
RF16	O Administrador deve ser capaz de se autenticar no sistema	B	A
RF17	O Administrador deve ser capaz de gerir marcas	M	A
RF18	O Administrador deve ser capaz de gerir categoria	M	A
RF19	O Administrador deve ser capaz de gerir modelos	M	A
RF20	O Administrador deve ser capaz de gerir produtos	M	A
RF21	O Administrador deve ser capaz de gerir usuários	M	A

\* B = Baixa, M = Média, A = Alta.

*Observação: acrescente quantas linhas forem necessárias.*

### 3.3 Requisitos Não-funcionais

ID	Descrição	Prioridade B/M/A
RNF01	Uso de Design responsivo nas interfaces gráficas	B
RNF02	Compatibilidade com todos os navegadores	M
RNF03	Tempo de resposta deve estar entre 200ms para requisições	A
RNF04	Utilizar spring security para autenticação segura	A



RNF05	Sistema deverá utilizar base em memória H2	A
RNF06	Sistema deverá suportar pelo menos 100 compradores simultâneos	M

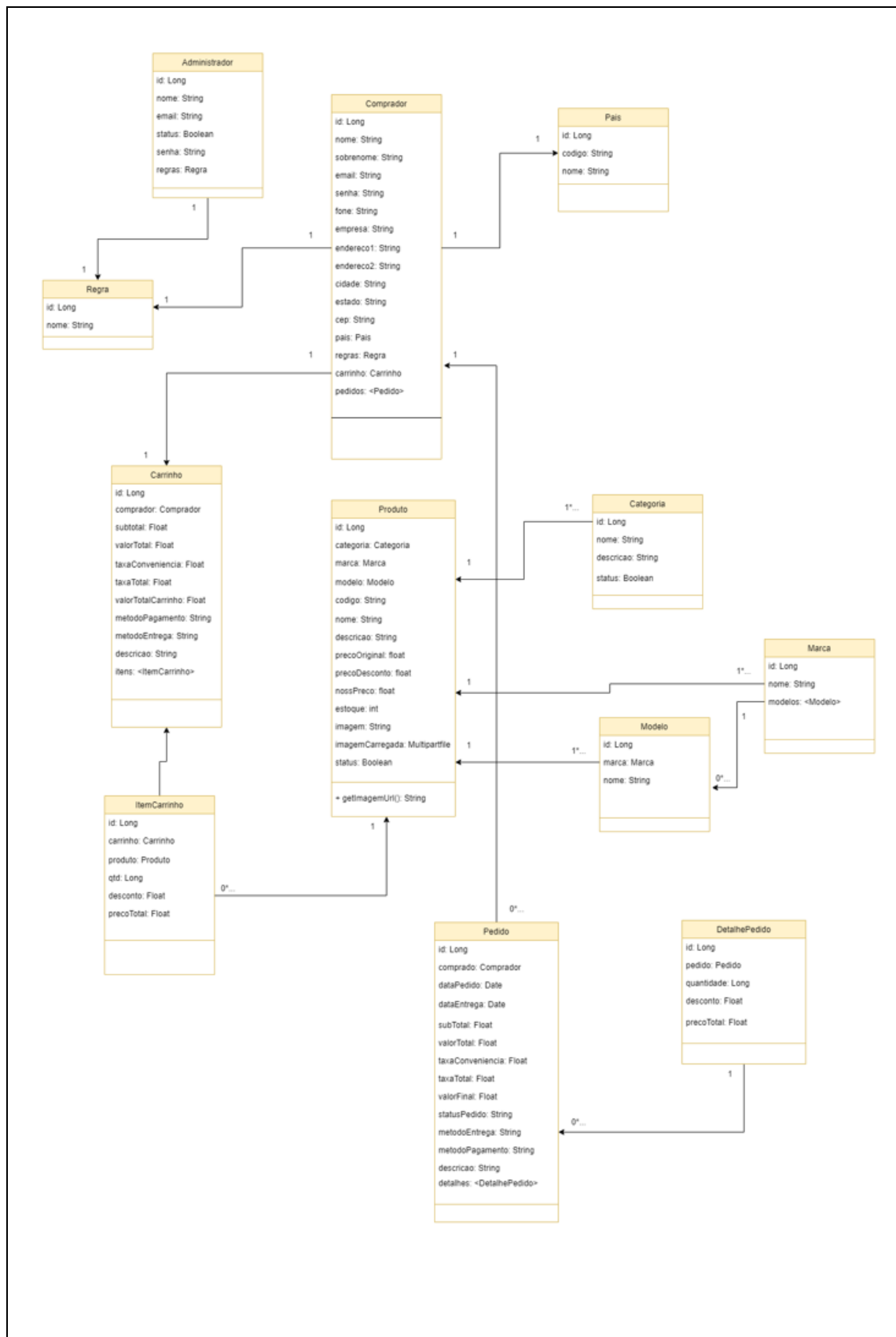
*Observação: acrescente quantas linhas forem necessárias.*

#### **4. Protótipo Navegável do Sistema**

Link do github do projeto com protótipo e vídeo explicativo:

<https://github.com/lucasscaramelo/puc-projeto-integrado-buybye>

## 5. Diagrama de Classes de Domínio



## **6. Arquitetura da Solução**

### **6.1 Padrão Arquitetural**

O padrão arquitetural definido para solução foi o Hexagonal, esse tipo de arquitetura trabalha em camadas, assim como Clean Architecture ou Onion, sendo possível privar as regras de negócio do mundo externo. Seu principal conceito se baseia em portas e adaptadores, sendo as portas representadas como entrada e saída do dado e os adaptadores representados como pontos de interação através das portas, ou seja, uma porta sempre conversará com um adaptador para acessar determinada camada da aplicação.

Por ser um padrão simples de aplicar e com uma estrutura organizacional bem transparente, ele foi escolhido. Além de todos os pontos já ditos, ele cobre toda a camada de segurança e encapsulamento que precisamos para garantir uma interação precisa dos usuários. Fazendo com que a aplicação fique limpa, transparente, segura e objetiva.

As tecnologias utilizadas na solução foram bem pensadas para que a junção delas com o padrão arquitetural escolhido fosse de grande sucesso. Abaixo lista das tecnologias utilizadas:

1. Maven
2. Java 11
3. Spring
4. H2 Database
5. Lombok
6. JUnit
7. Thymleaf
8. JavaScript
9. JQuery
10. Bootstrap
11. HTML 5
12. CSS

## 6.2 C4 model - Diagrama de Contexto

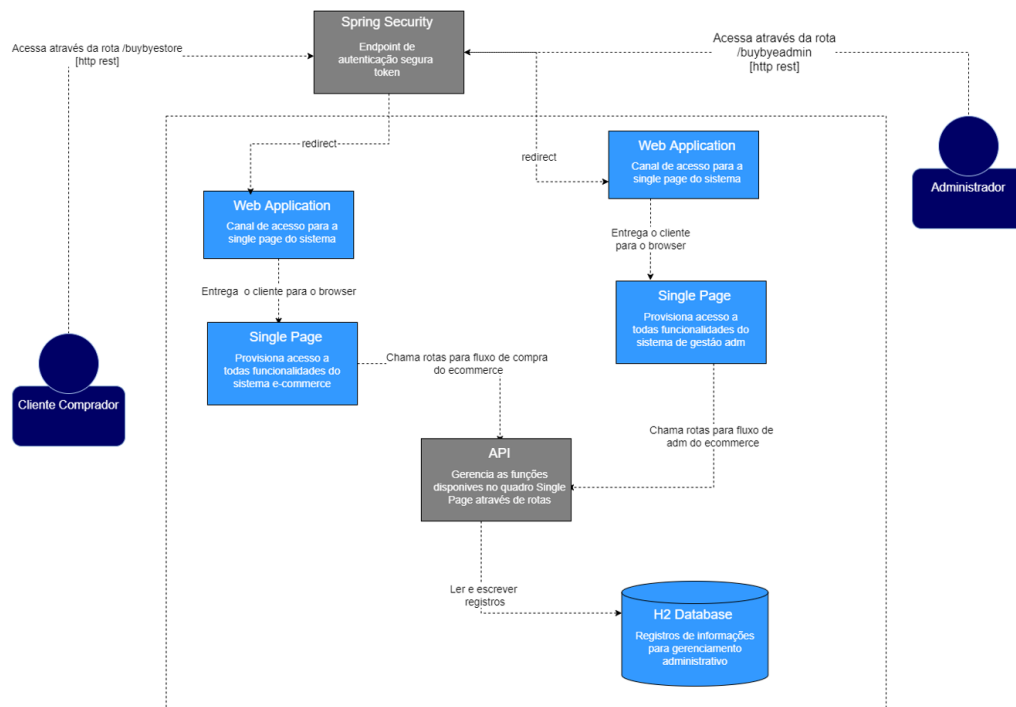


Figura 1 - Visão Geral da Solução

A figura 1 exibe o diagrama de contexto da solução proposta, com seus principais módulos e interfaces, sendo representados a especificação de entrada e os parâmetros de tráfego do usuário até o momento de comunicação com o banco de dados. Essa figura nos traz uma visão simplificada do que acontece nos bastidores da integração dos serviços, porém de maneira completa em ressaltar todas as fronteiras de integração, entrada e saída.

## 7. Frameworks de Trabalho

Os frameworks utilizados para a composição do projeto, são:

### 1. Frontend:

- Bootstrap
- JQuery
- Thymleaf
- Fontawesome

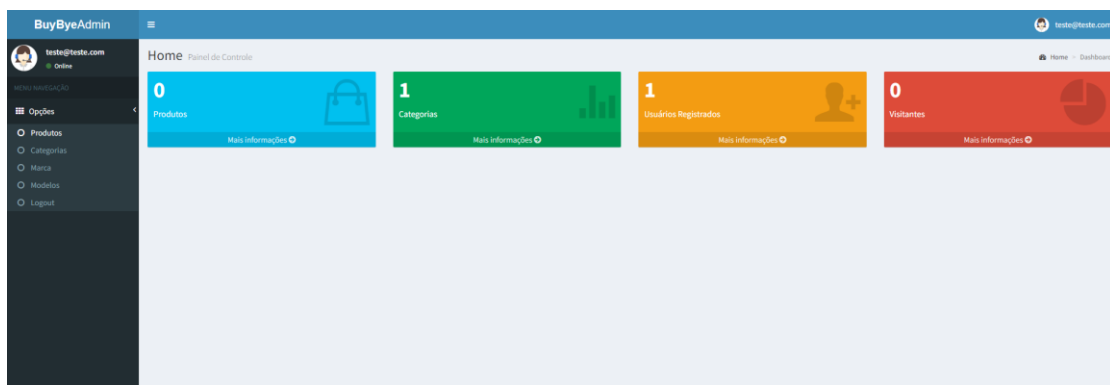
## 2. Backend e Persistência:

- Spring Framework
- JPA
- Hibernate
- Lombok
- H2 Database

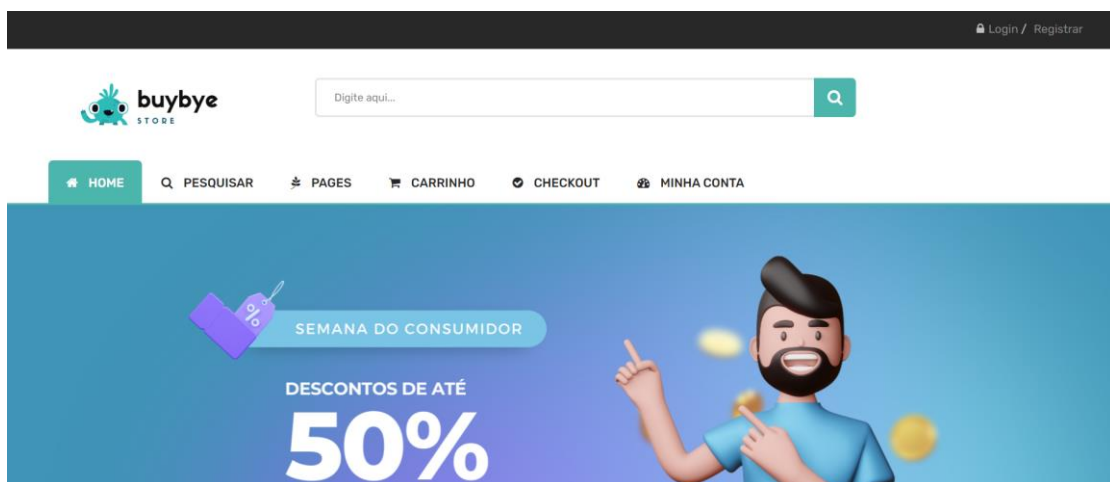
## 8. Estrutura Base do Front End

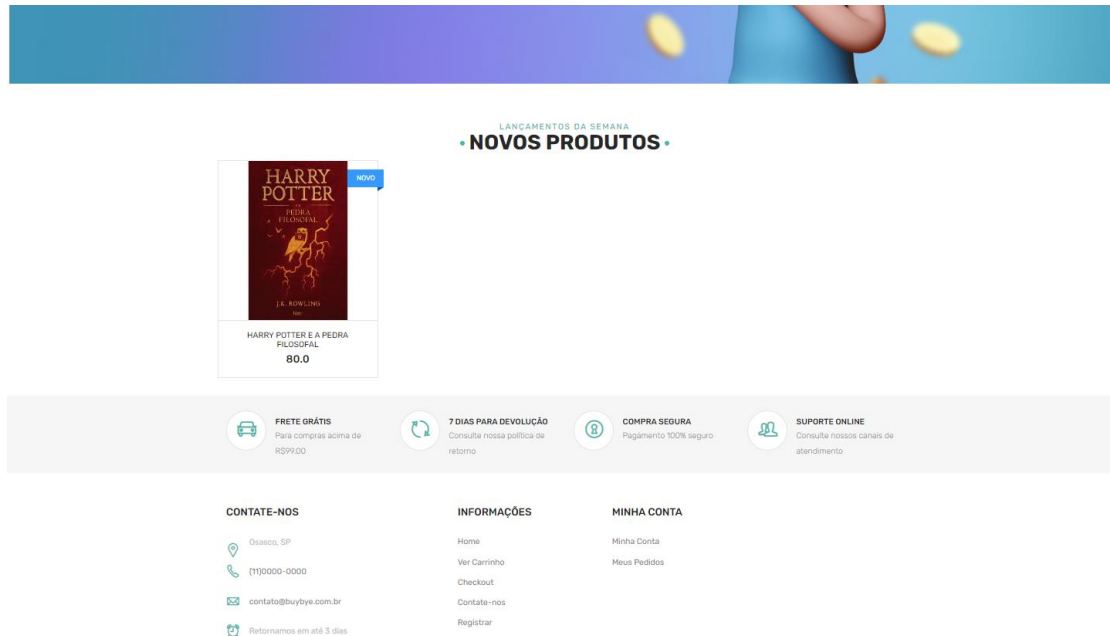
Serão apresentados abaixo os layouts mestres das 2 plataformas, administrador e comprador, ambas com suas particularidades devido a distinção de atividades de um para com outro.

- **Layout Mestre Administrador:**

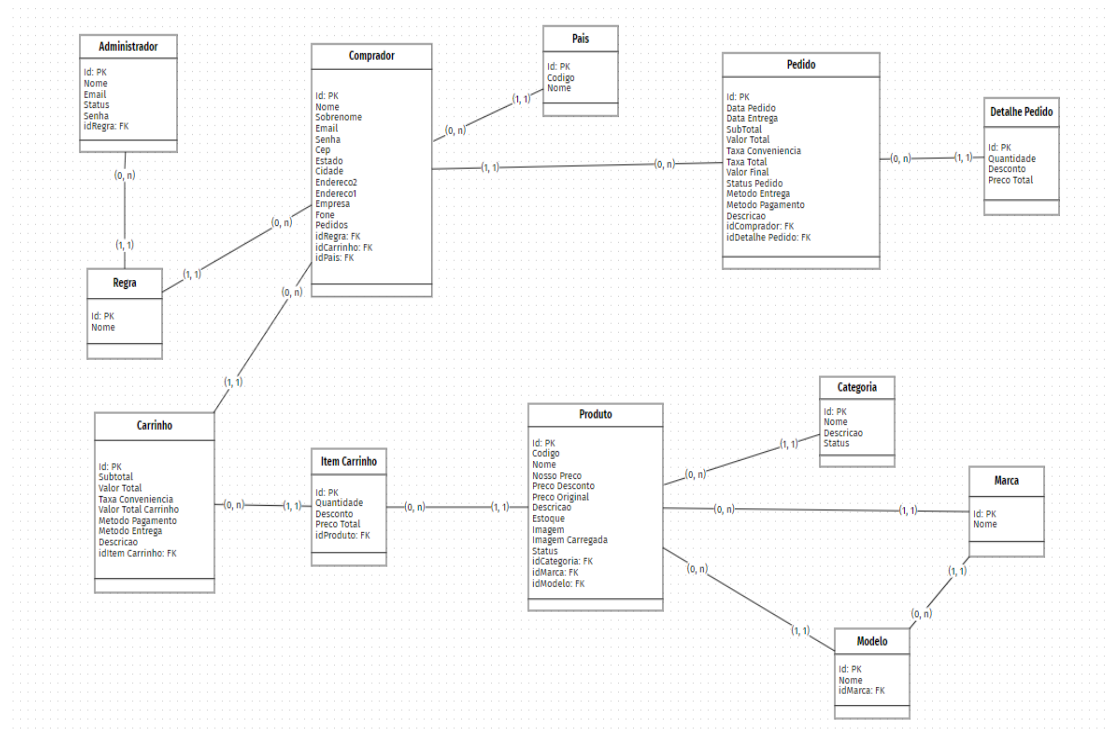


- **Layout Mestre Comprador:**





## 9. Modelo Relacional ou Projeto de Banco NoSQL



## 10. Plano de Testes

Número	Caso de uso	Objetivo do caso de teste	Entradas	Resultados esperados
1	Gerenciar Usuários	Alterar senha usuário existente	Senha nova	Senha alterada com sucesso
2	Gerenciar Usuários	Alterar senha usuário inexistente	Senha nova	Impossibilidade de alterar cadastro inexistente
3	Cadastrar Marca	Cadastrar marca	Informar nome válido para marca	Marca cadastrada com sucesso
4	Cadastrar Marca	Cadastrar marca	Informar nome inválido para marca	Impossibilidade de cadastrar marca
5	Cadastrar Modelo	Cadastrar modelo selecionando marca válida	Selecionar marca existente e informar nome modelo	Modelo cadastrada com sucesso
6	Cadastrar Modelo	Cadastrar modelo selecionando marca inválida	Selecionar marca inexistente e informar nome modelo	Impossibilidade de cadastrar modelo
7	Cadastrar Categoria	Cadastrar categoria válida	Informar nome válido para categoria	Categoria cadastrada com sucesso
8	Cadastrar Categoria	Cadastrar categoria inválida	Informar nome inválido para categoria	Impossibilidade de cadastrar categoria
9	Cadastrar Produto	Cadastrar produto válido	Informar categoria, marca, modelo válidos	Produto cadastrado com sucesso
10	Cadastrar Produto	Cadastrar produto inválido	Informar categoria, marca, modelo inválidos	Impossibilidade de cadastrar produto

## 11. Relatório de Execução de Testes

Número	Caso de teste	Saída esperada	Resultados encontrados	Aprovado?
1	Gerenciar Usuários	Cadastrar, editar e gerenciar usuários com sucesso	Quando há uma entrada inesperada o sistema retorna uma	Sim

			mensagem de erro e solicita que a ação seja revista	
2	Cadastrar Marca	Cadastrar, editar e excluir marca com sucesso	Quando há uma entrada inesperada o sistema retorna uma mensagem de erro e solicita que a ação seja revista	Sim
3	Cadastrar Modelo	Cadastrar, editar e excluir modelo com sucesso	Quando há uma entrada inesperada o sistema retorna uma mensagem de erro e solicita que a ação seja revista	Sim
4	Cadastrar Categoria	Cadastrar, editar e excluir categoria com sucesso	Quando há uma entrada inesperada o sistema retorna uma mensagem de erro e solicita que a ação seja revista	Sim
5	Cadastrar Produto	Cadastrar, editar e excluir produto com sucesso	Quando há uma entrada inesperada o sistema retorna uma mensagem de erro e solicita que a ação seja revista	Sim

## 12. Apropriação de Horas no Projeto

Histórico de apropriação de horas		
Data do registro	Atividade	Quantidade de horas
01/05/2022	Definir escopo do projeto	5h
03/05/2022	Definir escopo do projeto	1h
07/05/2022	Definir escopo do projeto	3h
12/05/2022	Definir escopo do projeto	3h 30min
14/05/2022	Definir escopo do projeto	4h
17/05/2022	Diagrama de caso de uso	3h
20/05/2022	Diagrama de caso de uso	1h
27/05/2022	Diagrama de caso de uso	5h
02/06/2022	Diagrama de classes de domínio	3h
06/06/2022	Diagrama de classes de domínio	4h
09/06/2022	Diagrama de classes de domínio	3h
13/06/2022	Introdução	4h
18/06/2022	Protótipo navegável	3h
22/06/2022	Protótipo navegável	4h



27/06/2022	Protótipo navegável	5h
01/07/2022	Protótipo navegável	4h
05/07/2022	Protótipo navegável	3h
05/07/2022	Protótipo navegável	2h
10/07/2022	Protótipo navegável	5h
15/07/2022	Protótipo navegável	6h
02/08/2022	Revisão artefatos entrega 1	3h
09/08/2022	Revisão artefatos entrega 1	4h
14/08/2022	Revisão artefatos entrega 1	6h
15/08/2022	Entrega etapa 1	2h
01/09/2022	Definição padrão arquitetural	8h
03/09/2022	Desenho diagrama contexto – C4	4h
05/09/2022	Definição de todos frameworks e tecnologias	20h
10/09/2022	Revisão do layout mestre “administrador”	4h
12/09/2022	Revisão do layout mestre “comprador”	4h
15/09/2022	Desenvolver frontend administrador	30h
21/09/2022	Desenvolver funcionalidade de cadastro de usuário administrador	3h
22/09/2022	Desenvolver funcionalidade de cadastro de produtos	5h
23/09/2022	Desenvolver funcionalidade de cadastro de categorias	4h
24/09/2022	Desenvolver funcionalidade de cadastro de marcas	4h
01/10/2022	Desenvolver funcionalidade de cadastro de modelos	4h
05/10/2022	Revisão artefatos entrega 2	8h
10/10/2022	Gravar vídeo explicativo	1h
15/10/2022	Entrega etapa 2	2h
01/11/2022	Desenvolver modelo relacional	2h
01/11/2022	Desenvolver plano de testes	1h
02/11/2022	Incluir funcionalidade de autenticação	8h
03/11/2022	Incluir menu sistêmico	7h
04/11/2022	Incluir funcionalidades de CRUD	8h
05/11/2022	Incluir funcionalidades de CRUD	8h
06/11/2022	Incluir funcionalidade adicionais	4h
07/11/2022	Incluir funcionalidade adicionais	5h
08/11/2022	Incluir funcionalidade adicionais	4h
09/11/2022	Execução e relatório de testes	5h
15/11/2022	Avaliação retrospectiva	1h
20/11/2022	Deploy aplicação	5h
30/11/2022	Revisão todo conteúdo	7h
15/12/2022	Entrega etapa 3	1h

### 13. *Código da Aplicação*

O código fonte se encontra em:

<https://github.com/lucasscaramelo/puc-projeto-integrado-buybye>

etapa3 > codigo\_fonte.

Link da aplicação:

<https://buybyestore.herokuapp.com/login>

### 14. *Avaliação Retrospectiva*

Com base na proposta inicial, foi percorrido diversos caminhos para compreensão e desenvolvimento do projeto. Cada etapa foi um desafio novo, onde tivemos que nos adaptar e estudar possibilidades de solução a todo tempo, não deixando de lado o objetivo do projeto, a entrega de um software que contemplasse os requisitos apresentados. Após a definição do que de fato seria desenvolvido, foi dado início aos desenhos técnicos e modelos que fariam parte do projeto, na sequência conseguimos colocar em prática aquilo que foi pensado para criação do protótipo e projeto final. Sinceramente, foi um misto de desafio, aprendizados, erros e acertos. A experiência foi incrível.

#### 14.1 *Objetivos Estimados*

O objetivo principal era construir um e-commerce que atendesse todos os nichos de forma que o administrador pudesse ter o controle e rastreamento das ações no decorrer dos processos de vendas, trazendo a segurança, flexibilidade e autonomia do comprador como pontos de relevância.

#### 14.2 *Objetivos Alcançados*

Conseguimos alcançar o objetivo principal, a criação de uma plataforma que dispusesse de um fluxo totalmente automático e gerenciável por seu administrador, com flexibilidade para atender todos os nichos e autonomia aos clientes compradores na hora de escolher o melhor fornecedor para seu produto.

#### 14.3 *Lições aprendidas*

Retrospectiva (Lições Aprendidas)		
	Descrição da Lição	Classificação
1	Segurança de aplicação	9

2	Performance volumétrica	8
3	Arquitetura e desenho de solução	9
4	Gerenciamento e controle de dados, utilizando normas LGPD	7
5	Modelagem de dados	8
6	Facilidade, não complexidade	10
7	Expansão, atender todos os públicos	8
8	Testes são super importantes no fluxo de desenvolvimento	10
9	Esteira automática faz toda a diferença no processo de CI/CD	10
10	Controle de versão, sempre se atentar a versão atual do ambiente	10
11	Se adaptar as tecnologias do mercado, fazendo combinações	8

## 15. **Referências**

[Vieira, José. Produtividade com Spring Boot. Medium, 2016.](#)

[Almeida, Felipe. Guia da modelagem de dados. Medium, 2018.](#)

[Zaguini, Luis. Sobre testes de software. Medium, 2019.](#)

[Martin, Robert. Clean Architecture. Alta Books, 2017.](#)