

**CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA**  
**FACULDADE DE TECNOLOGIA DE LINS PROF. ANTONIO SEABRA**  
**CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS PARA INTERNET**

**LUCCAS GODOY MIAZZO**

**PORTFÓLIO ACADÊMICO**

**LINS/SP**  
**1º SEMESTRE/2025**

**CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA**  
**FACULDADE DE TECNOLOGIA DE LINS PROF. ANTONIO SEABRA**  
**CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS PARA INTERNET**

**LUCCAS GODOY MIAZZO**

**PORTFÓLIO ACADÊMICO**

Trabalho de Conclusão de Curso apresentado à  
Faculdade de Tecnologia de Lins Prof. Antônio  
Seabra, para obtenção do Título de Tecnólogos em  
Sistemas para Internet

Orientador: Prof. Dr. Everaldo Silva de Freitas

**LINS/SP**  
**1º SEMESTRE/2025**

Godoy Miazso, Luccas

G588      Portfólio acadêmico / Luccas Godoy Miazso. — Lins, 2025.

59f.

Trabalho de Conclusão de Curso (Tecnologia em Sistemas para Internet) — Faculdade de Tecnologia de Lins Professor Antonio Seabra: Lins, 2025.

Orientador(a): Dr. Everaldo Silva de Freitas

1. Portfólio Acadêmico. 2. Portfólio Digital. 3. Currículo Digital. 4. Formação Acadêmica. 5. Sistemas para Internet. I. Silva de Freitas, Everaldo. II. Faculdade de Tecnologia de Lins Professor Antonio Seabra. III. Título.

CDD 004.61

Gerada automaticamente pelo módulo web de ficha catalográfica da FATEC Lins  
mediante dados fornecidos pelo(a) autor(a).

**LUCCAS GODOY MIAZZO**

## **PORTFÓLIO ACADÊMICO**

Trabalho de Conclusão de Curso apresentado à Faculdade de Tecnologia de Lins Prof. Antônio Seabra, como parte dos requisitos para obtenção do título de Tecnólogo em Sistemas para Internet sob orientação do Prof. Dr. Everaldo Silva de Freitas

Data de aprovação: \_\_\_\_/\_\_\_\_/\_\_\_\_

---

Prof. Dr. Everaldo Silva de Freitas (Orientador)

---

Nome do Examinador 1

---

Nome do Examinador 2

## DEDICATÓRIA

Dedico este trabalho àqueles que, com paciência, apoio e fé, caminharam ao meu lado mesmo quando o caminho parecia incerto. Cada passo só foi possível graças a vocês.

**Luccas Godoy Miazso**

## **AGRADECIMENTOS**

Agradeço, primeiramente, à minha família, pelo apoio, incentivo e compreensão durante toda a trajetória acadêmica. O suporte oferecido foi essencial para que este trabalho pudesse ser realizado com dedicação e perseverança.

Aos colegas e amigos do curso, pela troca de experiências, pelo companheirismo nas atividades em grupo e pelas contribuições que, direta ou indiretamente, auxiliaram no desenvolvimento deste projeto.

Aos professores, pelo compartilhamento de conhecimento, orientação e constante incentivo ao aprendizado. As aulas e orientações foram fundamentais para a formação acadêmica e realização deste trabalho.

Por fim, agradeço a todos que, de alguma forma, contribuíram para a conclusão desta etapa, deixando minha sincera gratidão pelo apoio durante esse percurso.

**Luccas Godoy Miazso**

## RESUMO

Observando o cenário atual, onde mostrar o que se sabe fazer é quase tão importante quanto saber de fato, surgiu a ideia de criar um portfólio acadêmico digital. A proposta deste trabalho foi desenvolver um portfólio que reunisse, de forma organizada, visual e funcional, os projetos, códigos, estudos e demais entregas feitas ao longo do curso de Sistemas para Internet. O foco principal foi destacar as competências adquiridas durante a graduação de um jeito prático e estruturado, refletindo tanto as habilidades técnicas quanto a identidade visual de quem o produziu, por meio da personalização presente nas atividades. Cada capítulo deste trabalho apresenta uma disciplina específica e um projeto correspondente, como os desenvolvidos nas matérias de Design Digital, Prática de Design, Programação para Internet III, Banco de Dados e Internet II, Prototipagem e Usabilidade, e Tópicos Especiais. A metodologia adotada consistiu na seleção e organização das atividades realizadas durante o curso, integradas em um ambiente digital intuitivo e acessível. Ao final, o resultado é um currículo digital que mostra, de forma clara e objetiva, tudo o que foi construído ao longo dos semestres da formação acadêmica, servindo como uma vitrine concreta do percurso e das conquistas do aluno.

Palavras-chave: Portfólio Acadêmico. Portfólio Digital. Currículo Digital. Formação Acadêmica. Sistemas para Internet.

## **ABSTRACT**

Observing the current scenario, where showing what one can do is almost as important as actually knowing how to do it, the idea of creating a digital academic portfolio emerged. The aim of this work was to develop a portfolio that gathers, in an organized, visual, and functional way, the projects, code, studies, and other assignments completed throughout the Internet Systems course. The main focus was to highlight the skills acquired during the degree in a practical and structured manner, reflecting both the technical abilities and the visual identity of the student, through the personalization present in the activities. Each chapter of this work presents a specific subject and a corresponding project, such as those developed in courses like Digital Design, Design Practice, Web Programming III, Databases and Internet II, Prototyping and Usability, and Special Topics. The methodology adopted consisted of selecting and organizing the activities carried out during the course, integrated into an intuitive and accessible digital environment. In the end, the result is a digital résumé that clearly and objectively showcases everything that was built throughout the academic journey, serving as a concrete showcase of the student's path and achievements.

Keywords: Academic Portfolio. Digital Portfolio. Digital Curriculum. Academic Training. Internet Systems.



## LISTA DE ILUSTRAÇÕES

Figura 2.1 – Bloco de Código HTML.....	13
Figura 2.2 – Bloco de Código CSS .....	14
Figura 2.3 – Pagina Home do projeto Metaverso.....	14
Figura 2.4 – Pagina Curiosidades do projeto Metaverso .....	15
Figura 2.5 – Conteúdo Visual de Imagens projeto Metaverso.....	16
Figura 3.1 – Trabalho de Rede Conexão de Diferentes Rotas.....	17
Figura 3.2 – Trabalho de Rede Conexão de Diferentes Redes .....	18
Figura 4.1 – Página Inicial do projeto Pizzaria Mobile .....	20
Figura 4.2 – Menu do projeto Pizzaria Mobile.....	21
Figura 4.3 – Menu do projeto Pizzaria Mobile.....	22
Figura 4.4 – Página Pizza Menu Session.....	23
Figura 4.5 – Página Pizza conteúdo da session .....	23
Figura 5.1 – Chave estrangeira.....	25
Figura 5.2 – Código SQL .....	26
Figura 5.3 – MER Vendas.....	27
Figura 5.4 – Código SQL Banco de Dados .....	28
Figura 5.5 – Tabelas com chaves estrangeira .....	29
Figura 5.6 – Manipulação de dados em SQL.....	30
Figura 6.1 – Tela Inicial de Login.....	32
Figura 6.2 – Tela Inicial do Aplicativo .....	33
Figura 6.3 – Notificação.....	35
Figura 6.4 – Menu Catálogo.....	36
Figura 6.5 – Carrinho de Compras .....	37
Figura 6.6 – Intanciando o Banco de Dados.....	38
Figura 6.7 – Tela de Pagamento .....	39
Figura 6.8 – Tela de Pedidos .....	39
Figura 6.9 – Tela de Perfil.....	40
Figura 7.1 – Configuração Inicial do Projeto.....	41
Figura 7.2 – Configuração da Classe Java.....	42
Figura 7.3 – Código Java do Primeiro Método.....	42
Figura 7.4 – Código Java do Segundo Método.....	43
Figura 7.5 – Teste funcional da disponibilidade do serviço pelo Tomcat.....	43

Figura 7.6 – Código Java do Cliente.....	44
Figura 7.7 – Criando o Web Services .....	45
Figura 7.8 – Execução do Arquivo JAR Cliente .....	46
Figura 8.1 – Paleta de Cores do Portifólio.....	47
Figura 8.2 – Portifólio Mobile.....	48
Figura 8.3 – Prototipo do Figma.....	49
Figura 8.4 – Fonte Roboto .....	50
Figura 8.5 – Logo do Portifólio .....	50
Figura 8.6 – Trecho de Código CSS .....	51
Figura 8.7 – Portifólio Desktop .....	52
Figura 8.8 – Div Modal no Site .....	52
Figura 8.9 – Modal Expandido .....	53
Figura 8.10 – Mapa do Site.....	54

## LISTA DE ABREVIATURAS E SIGLAS

AJAX – *Asynchronous JavaScript and XML*

API – *Application Programming Interface*

BR Modelo – Banco de Relacionamento Modelo (ferramenta de modelagem)

CSS – *Cascading Style Sheets*

DHCP – *Dynamic Host Configuration Protocol*

HTML – *HyperText Markup Language*

HTTP – *HyperText Transfer Protocol*

IP – *Internet Protocol*

Ionic – *Framework híbrido para desenvolvimento mobile/web*

JS – *JavaScript*

LAN – *Local Area Network*

MAN – *Metropolitan Area Network*

MER – *Modelo Entidade Relacionamento*

SQL – *Structured Query Language*

TCP/IP – *Transmission Control Protocol / Internet Protocol*

TS – *TypeScript*

UI – *User Interface*

URL – *Uniform Resource Locator*

UX – *User Experience*

VPN – *Virtual Private Network*

VSCoDe – *Visual Studio Code (IDE)*

WAN – *Wide Area Network*

XML – *Extensible Markup Language*

IDE - *Integrated Development Environment*

SOAP - *Simple Object Access Protocol*

JAR - *Java Archive*

vh - *Viewport Height*

vw - *Viewport Width*

## SUMÁRIO

1	INTRODUÇÃO .....	11
2	PADRÕES DE PROJETO I .....	12
3	REDES E INTERNET .....	17
4	PROGRAMAÇÃO DE SITIOS INTERNET .....	20
5	BANCO DE DADOS E INTERNET II .....	25
6	DESENVOLVIMENTO PARA DISPOSITIVOS MOVEIS I.....	31
7	ARQUITETURA ORIENTADA A OBJETO .....	41
8	MANUAL DO PORTIFOLIO DIGITAL .....	47
9	CONSIDERAÇÕES FINAIS .....	55
	REFERÊNCIAS .....	56

## 1 INTRODUÇÃO

Atualmente, sabe-se que mostrar as habilidades (o que sabe fazer) vale quase o mesmo que realmente saber fazer. Foi pensando nisso que foi desenvolvido esse portfólio acadêmico digital que traz um resumo dos trabalhos, códigos, estudos e projetos que foram desenvolvidos durante a caminhada acadêmica no curso de Sistemas para Internet. A ideia não é apenas guardar esses arquivos, mas sim dar uma cara prática, organizada e com a identidade visual de seu criador.

A proposta é simples: apresentar uma plataforma clara que una as habilidades técnicas a um estilo pessoal. Cada capítulo do portfólio foca em uma disciplina, dentre elas o Design Digital, Programação para Internet III, Banco de Dados e Internet II, Prototipagem e Usabilidade trazendo uma síntese do projeto, o que foi desenvolvido, como foi planejada cada etapa.

Para isso, foram selecionados os melhores projetos, foi criado/desenvolvido um site funcional com ferramentas utilizadas no dia a dia como Figma, VSCode e Ionic. Houve um cuidado especial para apresentar a identidade visual do autor (tudo tenha a minha “cara”). Foi pensada a usabilidade, o visual com o objetivo de fazer algo que seja funcional tanto em desktop quanto em mobile.

O portfólio foi pensado para acompanhar toda uma trajetória dentro do curso, destacando o que foi aprendido e como houve uma evolução quando é observado o que tinha de conhecimento no início e o quanto houve de evolução ao término do curso. Mais do que uma entrega acadêmica, o portfólio serve como uma vitrine real da formação acadêmica e profissional, preparado para mostrar “quem eu sou” no mercado. Como diria Donald Norman, “Design centrado no ser humano começa por entender as necessidades” e esse é exatamente o propósito deste trabalho: de forma clara, objetiva e com personalidade.

## 2 PADRÕES DE PROJETO I

Durante o primeiro semestre do curso foram ministradas disciplinas para o aprendizado das tecnologias de programação web. Fazendo uso do HTML (HyperText Markup Language ou em português Linguagem de Marcação de HyperTexto) como linguagem de marcação e o CSS (Cascading Style Sheets) como linguagem de marcação de estilos para personalizar os projetos, além de uma introdução da linguagem Javascript. Foi ensinado as principais técnicas e abordagens dessas linguagens, dando foco para o design e estruturação. A aula foi desenvolvida com parte teórica e pratica, sendo que a avaliação final foi feita por meio de um projeto multidisciplinar entre as disciplinas: **Padrões de Projeto I, Criação de Conteúdo na Web e Design Digital**, introduzindo de maneira mais sucinta as ferramentas descritas.

O HTML é uma linguagem de marcação com o foco no desenvolvimento de aplicações web, funcionando como uma estrutura em blocos utilizando tags, a grosso modo ela define a estrutura de toda a web. Seu criador é Tim Berners-Lee, o mesmo que criou outros protocolos como o HTTP (Hypertext Transfer Protocol em português Protocolo de Transferência de HyperTexto), sendo a ideia original da criação um conjunto de ferramentas operacionais para a Tim empresa de telefonia, contudo devido as possibilidades que o HTML fornece e suas melhoras durante o tempo acabou se tornando a principal linguagem de marcação da web.

Operando em blocos de código com tags pré-estabelecidas (palavras reservadas), criando uma estrutura onde cada elemento tem o seu lugar definido, evitando complicações de leitura caso haja diferentes idiomas ou diferenças culturais (formato de teclado, por exemplo), sendo isso possível pela escolha da definição do inglês para a declaração das tags, observando isso no trecho de código HTML na Figura 2.1.

Figura 2.1 – Bloco de Código HTML

```
<header>
  

  <nav class="menu-opcoes">
    <ul>
      <li>
        <a href="home.html">Home</a>
      </li>
      <li>
        <a href="historia.html">Historia</a>
      </li>
      <li>
        <a href="curiosidades.html">Curiosidades</a>
      </li>
      <li>
        <a href="exemplos.html">Exemplos</a>
      </li>
      <li>
        <a href="integrantes.html">Integrantes</a>
      </li>
    </ul>
  </nav>
</header>
```

Fonte: Elaborada pelo autor, 2022

O CSS foi criado tendo como foco a edição e melhoria visual das páginas na internet sendo um marcador de texto de estilos, ele estiliza e personaliza as páginas da web, através dele é possível criar identidade visual e desenvolvimento gráfico. Sua funcionalidade principal é bastante direta, trabalhando com um bloco de declaração com seletor, onde se atribui uma propriedade e um valor. Assim como no HTML, utiliza-se a língua inglesa para a declaração das tags, como pode ser observado na Figura 2.2, que mostra um bloco de código CSS.



Figura 2.2 – Bloco de Código CSS

```
h1 {
  color: white;
  line-height: 25px;
  font-size: 25px;
  padding-top: 10px;
  padding-bottom: 15px;
  border-bottom: 3px solid #424242;
}
```

Fonte: Elaborada pelo autor, 2022

Voltando ao tema principal deste capítulo, que é o desenvolvimento do trabalho, ele foi realizado em equipes de 3 a 4 alunos. O resultado final foi um site estático, com o conteúdo apresentado relacionado à tecnologia. Observa-se na Figura 2.1, que mostra a página principal, o home, como é conhecido esse termo da web para definir a página inicial das aplicações web.

Figura 2.3 – Pagina Home do projeto Metaverso



Fonte: Elaborada pelo autor, 2022

O material desenvolvido pelo grupo foi chamado de “metaverso”, que de uma maneira resumida seria a interação ampliada com outras pessoas ou aplicação através do uso da tecnologia de aspecto visual e auditivo, proporcionando uma experiência de realidade aumentada em um mundo “virtual”. A ideia era que fosse possível a realização de treinamentos e interações em ambientes virtuais de forma a promover o desenvolvimento pessoal e profissional dos usuários. A figura 2.2 mostra a página de curiosidades do site.

Figura 2.4 – Pagina Curiosidades do projeto Metaverso



Fonte: Elaborada pelo autor, 2022

O conceito de Design e estruturação fica implícito com as técnicas sendo aplicadas durante o desenvolvimento do projeto. Além disso, foram seguidas as orientações e métodos de escrita aprendidos durante a matéria de Criação de Conteúdo na Web, e também é importante citar que foi utilizado um material bibliográfico com orientações da escrita Web, no caso o livro: Como escrever para a Web do escritor Guilherme Franco. Uma frase importante do autor é: *"Para que a Web alcance esse status completo, os desenvolvedores de conteúdo e os usuários devem tirar vantagem de seus atributos, ambiente e funcionalidades. A Web deve passar por um processo de maturação (pagina 21)"*.

Voltando para a parte da estruturação, os conceitos de apresentação, as técnicas de adaptação do conteúdo visual de imagens e sua empregabilidade podem ser observados na Figura 2.3.

O espaçamento, a disposição de espaço, além da legenda abaixo das imagens, são códigos de CSS combinados com a estruturação do HTML, permitindo manipular e demonstrar as imagens, aproveitando o espaço e criando um padrão. Isso resulta

em uma apresentação mais agradável e padronizada, seguindo as recomendações técnicas da web.

Figura 2.5 – Conteúdo Visual de Imagens projeto Metaverso



Fonte: Elaborada pelo autor, 2022

Vale ressaltar as escolhas das cores do projeto um fundo sendo um roxo quase preto com letras e material escrito em branco permitindo uma fácil visualização do conteúdo sem prejudicar a visualização.

Essa técnica foi aprendida num dos tópicos da disciplina de Design Digital. Ao final da elaboração do projeto foi possível perceber a interação entre as disciplinas. Mesmo que a disciplina escolhida tenha sido Padrões de Projeto I e seus conceitos tenham sido bastante utilizados, a combinação de outros conceitos e técnicas das outras disciplinas é perceptível. Isso leva a conclusão de que o projeto resultou em uma experiência técnica aprimorada, mostrando que as disciplinas aprendidas durante o curso se complementam entre si e contribuem para um melhor aprendizado profissional.

### 3 REDES E INTERNET

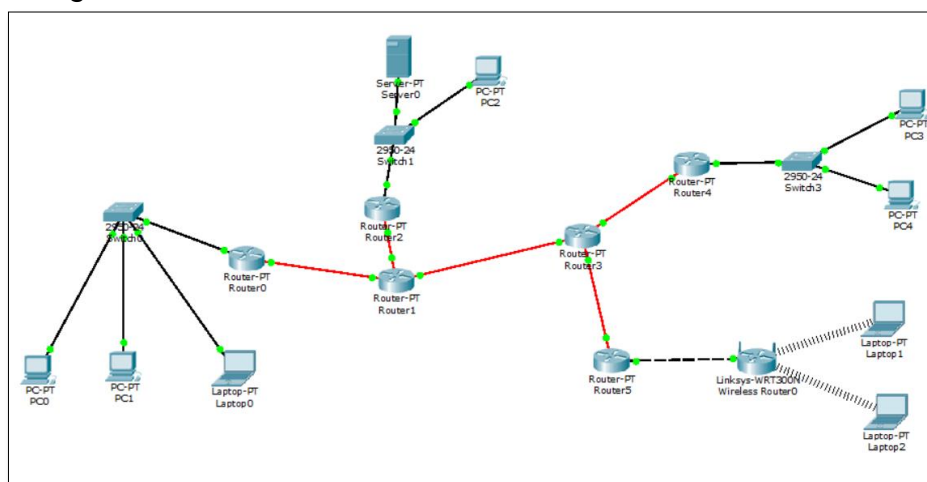
Na disciplina de Redes e Internet, o conteúdo desenvolvido durante o semestre foi o funcionamento dos diferentes tipos de rede que existem. "A internet é essencialmente uma interconexão de várias redes, permitindo que computadores em diferentes partes do mundo se comuniquem." - Andrew S. Tanenbaum, em Redes de Computadores.

Utilizando o software Packet Tracer, uma aplicação desenvolvida pela Cisco para testes de configuração de diferentes redes, foi possível simular as conexões e a comunicação de rede, estudando os métodos e configurações. Entre as diferentes redes que foram apresentadas, e suas organizações físicas e lógicas, temos: WAN (Wide Area Network), LAN (Local Area Network) e MAN (Metropolitan Area Network).

A rede WAN rede de longas distancias que conecta varias LANs, MANs, conectando cidades estados até mesmo países os cabos subaquáticos que passam por debaixo dos oceanos estão utilizando a organização WAN. A rede LAN uma rede local para conectar dispositivos, seja em uma casa, escola ou empresa, é a rede mais comum de se deparar, estando mais diretamente ligada ao nosso cotidiano através dos roteadores. E rede MAN é uma rede que geralmente é metropolitana, ligando as LANs entre si e entre cidades, dependendo do tamanho, mas ainda menor que uma WAN.

A organização das redes se torna algo necessário devido a vários fatores, entre os principais: segurança, organização lógica e física, limitando ou expandindo a capacidade da rede de computadores conforme a necessidade.

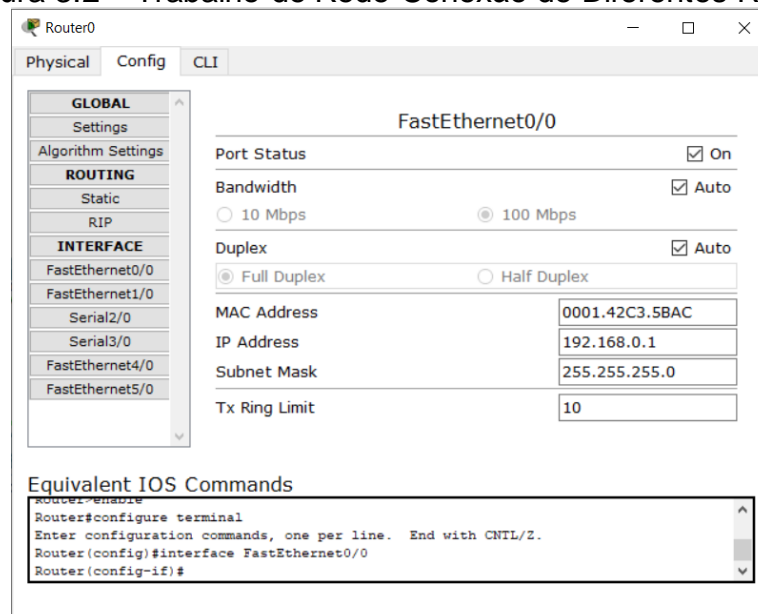
Figura 3.1 – Trabalho de Rede Conexão de Diferentes Redes



Fonte: Elaborada pelo autor, 2023

Observa-se na figura 3.1 é o trabalho desenvolvido como atividade avaliativa. Pode-se notar que mesmo que haja diferentes dispositivos e conexões a comunicação é mantida de uma ponta para a outra. Detalhando o funcionamento a o trabalho de maneira sucinta, são 3 redes LAN conectadas com uma rede WAN com diferentes dispositivos e tecnologias, configurando o protocolo de rede de cada roteador para que os computadores ligados não conflitem e possam se comunicar com outras máquinas de outra rede LAN.

Figura 3.2 – Trabalho de Rede Conexão de Diferentes Redes



Fonte: Elaborada pelo autor, 2023

O IP (Internet Protocol) é um identificador numérico único que é atribuído para dispositivo na rede, permitindo a comunicação envio e recebimento de pacotes, funciona como se fosse um “endereço” da máquina. Existem dois padrões de IP utilizados, IPv4, constituído por 32bits exemplo 192.168.0.1, e o IPv6 com 128bits exemplo 2001:0db8:85a3::8a2e:0370:7334.

Em redes locais os computadores usam o protocolo DHCP(Dynamic Host Configuration Protocol), que atribui os endereços IP de forma automática sendo necessário como na imagem definir um padrão de IP , a interligação entre as redes é garantida através do protocolo TCP/IP, quando um pacote é enviado na rede ele contém os dados do destinatário e de quem enviou como forma de segurança contendo uma chave que somente quem recebeu o pacote vai conseguir “ler” o seu conteúdo, vale ressaltar que todo dispositivo possui um “CPF” conhecido como Mac

Address(Media Access Control), sendo um meio de identificação único através dele que os dispositivos são reconhecidos na rede.

Esse projeto avaliativo e essa disciplina contribuíram de maneira significativa para o desenvolvimento profissional, com foco em abordar, em vez de apenas o que vemos na tela, algo como entender como aquilo que enxergamos chegou até nós, e toda a estrutura física da internet por trás disso.



## 4 PROGRAMAÇÃO DE SÍTIOS INTERNET

A disciplina Programação de sítios Internet foi ministrada durante o segundo semestre. Esta disciplina abordou a aprendizagem de uma nova linguagem e estruturação web, além de aprimorar a base já adquirida na disciplina anterior Padrões de Projeto 2, em específico foi ministrado as linguagens de programação web CSS (Cascading Style Sheets), JavaScript e HTML (HyperText Markup Language). Foram utilizadas como ferramentas para desenvolvimento dos projetos o Visual Studio Code e Navegadores Web.

Sendo uma continuação da disciplina do semestre anterior os projetos desenvolvidos em sala foram um aprimoramento do conhecimento técnico e habilidade técnica, mostrando uma visível evolução. O projeto abordado neste capítulo demonstra a refinação do ensino e desenvolvimento acadêmico.

O projeto utilizado como avaliação teve como foco testar e avaliar os conhecimentos técnicos além da capacidade criativa do aluno, foi proposto um site de pizzeria estático, porém farto da utilização de recursos web visuais conforme observa-se na figura 4.1, página inicial do site.

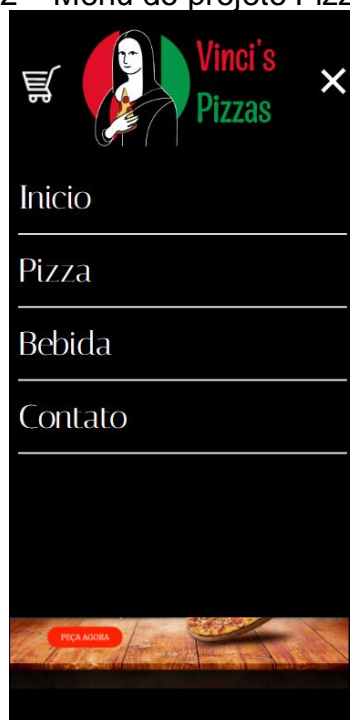
Figura 4.1 – Página Inicial do projeto Pizzaria Mobile



Fonte: Elaborada pelo autor, 2023

Pode-se perceber na figura 4.1 um layout de tela mobile (termo técnico para dispositivos moveis, em geral os mesmo tem a orientação de tela na vertical diferente dos dispositivos desktop que tem a orientação de tela na horizontal). Essa orientação de tela e adaptação de visualização só foi possível graças a um recurso da linguagem CSS sendo ela media query que seria a adaptação de elementos gráficos para diferentes tipos de tela, além de usar um método de programação responsivo para o design utilizando unidade de medidas como Vh e Vw (medidas de escala de visualização, é comum utilizar elas pois as mesma se adaptam ao tamanho da tela não a uma medida fixa) em vez de pixel( Pixel é a menor unidade de uma imagem , sendo um ponto de cor que combinado com outros, formam um elemento gráfico). Comumente utiliza se de uma técnica de programação denominada Mobile First ela foi utilizada neste projeto, olhando para a figura é nítido ver a preocupação com o design de se criar um menu em um único botão ao ser clicado pelo usuário ele expande mostrando opções de navegação para outras áreas além de contrair caso clique no elemento “X” todo esse comportamento manipulação de elementos gráficos só foi possível utilizando a linguagem JavaScript para ajustar e comandar o comportamento dos itens do menu.

Figura 4.2 – Menu do projeto Pizzaria Mobile



Fonte: Elaborada pelo autor, 2023



A navegação tradicional não foi esquecida apenas não foi o real foco do projeto levando em conta que se trata de um site de vendas de pizza referenciando muito aos moldes de aplicativos de venda mesmo sendo um e-commerce (termo técnico usado para descrever sites que tem o foco em vendas). Conforme a figura 4.3 é possível observar o CSS através do media query mudando o tamanho dos elementos gráficos do site de forma automática apenas alterando a resolução de exibição da tela, o menu por exemplo ficou livre em uma linha horizontal sem a necessidade de estar em um botão pois há espaço para isso, aproveitando da resolução maior que possui.

Figura 4.3 – Página Inicial da Pizzaria em Resolução para Desktop



Fonte: Elaborada pelo autor, 2023

O desenvolvimento do projeto teve um foco maior em desenvolver habilidades para manipular os elementos gráficos conforme descrito, como os recursos da linguagem CSS e JavaScript, assim como no menu da página inicial a pagina Pizza possui um conjunto de elementos gráficos que demonstra o uso das ferramentas, e olhando a figura 4.4 nota-se que não há nenhuma pizza devido ao recurso de menu session(sessão em uma tradução literal, seria como se partes do site ficassem oculta até determinada ação ser executada) fazendo a manipulação do comportamento pelo Javascript.

Figura 4.4 – Página Pizza Menu Session



Fonte: Elaborada pelo autor, 2023

Quando o usuário clicar em algum item do menu aparecerá um novo conteúdo na página conforme o escolhido, mostrando as pizzas disponíveis na aplicação, conforme a figura 4.5 demonstra.

Figura 4.5 – Página Pizza conteúdo da session



Fonte: Elaborada pelo autor, 2023

O conteúdo e sua organização estão sendo estruturado e mostrado na tela através do uso do CSS Grid, para alinhar corretamente os quadrados das pizzas e seus elementos obtendo um resultado padronizado, definindo também quantas pizzas estarão uma do lado da outra caso mude a resolução do dispositivo. Além disso, utilizou-se o CSS Display Flex para alinhar de maneira automática o conteúdo no centro dentro do Grid, mesclando as técnicas e aprendizado da disciplina em um único conjunto de elementos gráficos.

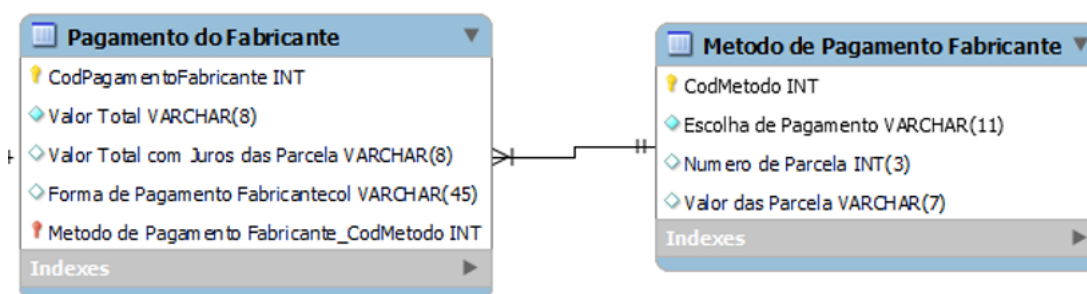
A disciplina foi concluída mesclando diversas ferramentas e práticas e ensinamentos com o foco em criar um padrão de desenvolvimento consistente e intuitivo conforme demonstrado neste projeto, sendo possível criar elementos gráficos e lógicos avançados para web.

## 5 BANCO DE DADOS E INTERNET II

A disciplina Banco de Dados e Internet II teve como foco aprimorar o conhecimento e técnicas de sua matéria antecessora dando maior destaque para a parte prática de desenvolvimento e correção de banco de dados.

Falando um pouco da matéria antecessora, foi aprendido os conceitos de interpretação de banco de dados e modelagem de tabela por parte visual. Além disso, foi possível compreender o “fluxo” que o banco segue através de projetos com o software BR Modelo, além de desenvolver o Diagrama Entidade Relacionamento ou Modelo Entidade Relacionamento (MER) no MySQL Workbench, por suas ferramentas de criação visual. Isso seria a introdução para aprender a programar a linguagem Structured Query Language (SQL), desenvolver lógica para conseguir interpretar os conceitos como relacionamento de tabelas, herança e chave estrangeira.

Figura 5.1 – Chave estrangeira



Fonte: Elaborada pelo autor, 2024

No problema descrito nessa atividade, representado na tabela da figura 5.1, a tabela Pagamento do Fabricante possui uma chave estrangeira que seria a chave primária do Metodo de Pagamento Fabricante, vale ressaltar que em banco de dados seja no MER ou programando se tem a boa prática de não usar pontuação, pois a mesma pode atrapalhar uma futura modificação ou inserção de dados. Ao invés de se criar os atributos de pagamento na tabela, pela sua complexidade e modelo de negócio, se tornou necessário criar mais uma tabela.

Isso é apenas uma das formas de representatividade do banco de dados, sendo formado por um conjunto de códigos e instruções de linguagem de banco, nesta disciplina o SQL.

O SQL é uma linguagem de programação padrão criada para gerenciar e manipular bancos de dados relacionais. Foi projetada inicialmente na década de 70

pela *International Business Machines Corporation* (IBM), utilizando de uma estrutura de dados tabulares, fazendo uso de comandos como SELECT, INSERT e UPDATE e DELETE entre outros para manipular dados armazenados, além claro da capacidade de criar e modificar estruturas de dados já existentes possibilitando uma grande flexibilidade operacional de sistema.

Essa linguagem voltada para banco de dados se tornou um padrão internacional, pela International Organization for Standardization (ISO) e pela American National Standards Institute (ANSI), sendo utilizada em gerenciamento de banco de dados, como MySQL, Oracle Database entre outros, possibilitando uma ferramenta eficaz para integrar dados de diferentes sistemas independente de sua complexidade. *“SQL permite a integração e o gerenciamento de dados provenientes de diferentes sistemas, promovendo a interoperabilidade em ambientes heterogêneos.”* - Elmasri e Navathe em Sistemas de Banco de Dados.

O foco da disciplina Banco de Dados e Internet II, esta na programação da linguagem de banco SQL, que possui uma estrutura de blocos intercalando com comandos diretos. A linguagem adota o idioma inglês como sua base de comandos.

Figura 5.2 – Código SQL

```
create table cliente(  
  cpf varchar(14) primary key,  
  nome varchar(35) not null,  
  rg varchar(8) not null,  
  bairro varchar(20) not null,  
  cidade varchar(32) not null,  
  rua varchar(28) not null,  
  numero_da_casa varchar(4) not null  
);  
  
insert into cliente(cpf, nome, rg, bairro, cidade, rua, numero_da_casa)  
values ('10366617314','Edvaldo da Cunha Santos do Nascimento', '98765432',  
'suburbio', Limeira, 'Borges Lima', '6663');
```

Fonte: Elaborada pelo autor, 2024

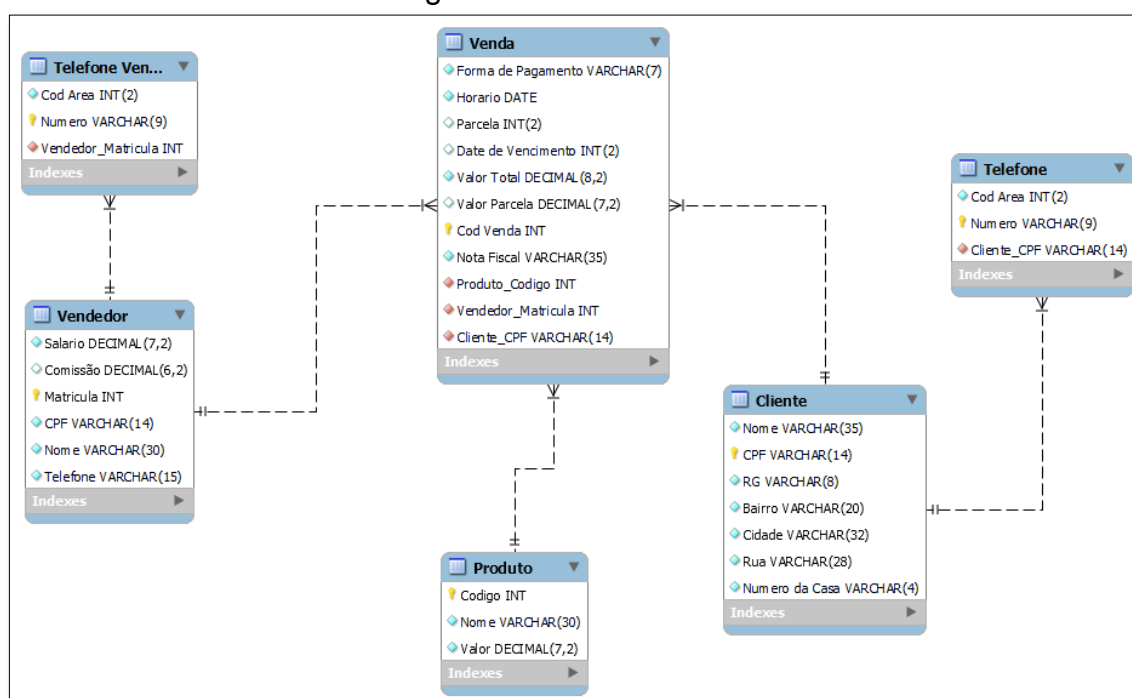
Em SQL a linha de código finaliza somente com “;” ponto e vírgula quando se tem um bloco de código costuma se utilizar a “,” virgula conforme mostrado na figura 5.2, percebe-se que nesse código está sendo criado uma tabela cliente e a inserção direta de um dado nessa mesma tabela. O SQL interpretado de cima para baixo além

de que, uma vez dado o comando não há necessidade de repedir para continuar funcionando.

Conforme foi descrito nesse capítulo usado como exemplo, a programação de banco de dados SQL segue suas próprias regras e condições, seguindo instruções do professor foi desenvolvido em sala de aula um projeto avaliativo no qual consiste em desenvolver um banco de dados relacional. A escolha do banco ficou a cargo do aluno, porém com a condição de haver pelo menos 3 tabela e chave estrangeira, para que o mesmo você testado com inserção de dados e alterações.

O Banco escolhido foi um banco de dados de vendas observando nesse MER na figura 5.3

Figura 5.3 – MER Vendas



Fonte: Elaborada pelo autor, 2024

Possuindo 6 tabelas e chaves estrangeiras, esse banco de dados teve sua lógica corrigida em sala pelo professor da disciplina. O MER foi indispensável para a criação e aplicação dos conceitos e técnicas de gerenciamento e criação de banco de dados. Conforme observado na figura 5.3, a tabela venda possui chave estrangeira por questões lógicas e de boa prática em SQL. Quando uma tabela possui chave estrangeira de outra(s), se criam primeiro as tabelas que não possuem, como a tabela produto, cliente e vendedor, mostrado na figura 5.4.

Figura 5.4 – Código SQL Banco de Dados

```
create database venda;

create table vendedor(
vendedor_matricula int primary key,
salario decimal(7,2) not null,
comissao decimal(6,2),
cpf varchar(14) not null,
nome varchar (30) not null,
telefone varchar(15) not null
);

create table cliente(
cpf varchar(14) primary key,
nome varchar(35) not null,
rg varchar(8) not null,
bairro varchar(20) not null,
cidade varchar(32) not null,
rua varchar(28) not null,
numero_da_casa varchar(4) not null
);

create table produto(
codigo_produto int primary key,
nome varchar(38) not null,
valor decimal(7,2) not null
);
```

Fonte: Elaborada pelo autor, 2024

Conforme descrito na figura 5.4 para criar as tabelas, foi necessário criar primeiramente um banco de dados, e neste caso, foi utilizado o comando CREATE DATABASE VENDA; como foi utilizado o mesmo script (termo para definir o que seria uma sequência de código) e as tabelas foram escritas na sequência, segundo a ferramenta MySQL Workbeanch visto onde foi feito e executado entende-se que pertencem ao mesmo banco não tendo a necessidade de usar o comando USE DATABASE <NOME DO BANCO>. Depois de criadas as tabelas, foram definidas as tabelas que possuem chave estrangeira conforme o estabelecido no MER, é através delas que foi definido o relacionamento no banco de dados, qual tabela vai ligar com qual, além de ser feito o controle do fluxo de dados, observando isso na figura 5.5.

Figura 5.5 – Tabelas com chave estrangeira

```
create table telefone(  
numero varchar(9) primary key,  
codigo_area int(2) not null,  
cpf varchar(14),  
foreign key (cpf) references cliente(cpf)  
);  
  
create table telefone_vendedor(  
numero varchar(9) primary key,  
codigo_area int(2) not null,  
vendedor_matricula int,  
foreign key (vendedor_matricula) references vendedor(vendedor_matricula)  
);  
  
create table venda(  
codigo_venda int primary key,  
forma_de_pagamento varchar(7) not null,  
horario date,  
parcela int(2),  
date_de_vencimento int(2),  
valor_total decimal(8,2) not null,  
valor_parcela decimal(7,2),  
nota_fiscal varchar(35) not null,  
codigo_produto int,  
foreign key (codigo_produto) references produto(codigo_produto),  
vendedor_matricula int,  
foreign key (vendedor_matricula) references vendedor(vendedor_matricula),  
cliente_cpf varchar(14),  
foreign key (cliente_cpf) references cliente(cpf)  
);
```

Fonte: Elaborada pelo autor, 2024

As foreign key (chave estrangeira) geralmente são as chaves primarias de outras tabelas isso ocorre quando se tem mais de um atributo para um atributo ou em relacionamento n para n ou quando se tem a necessidade de se cruzar dados entre entidades. Observando o script do código é notável que um cliente pode ter mais de um telefone assim como uma venda quase sempre vai ter mais de um produto vendido e que esse produto possua diferentes códigos, então se utiliza essa estratégia de relacionamento para gerir melhor os dados perante ao problema para o qual ele foi criado. Para testar e aprender com os resultados, foi realizado nessa matéria inserções e comandos de manipulação no banco, a figura 5.6 demonstra as principais linhas de código que se utiliza em SQL.



Figura 5.6 – Manipulação de dados em SQL

```
insert into telefone(numero, codigo_area, cpf) values ('125415515' , '27' , '9876543211');

DELETE FROM cliente WHERE cpf = '10366617314';

DELETE FROM telefone WHERE cpf = '10366617314' LIMIT 1;

alter table cliente
add column sexo char(1);

alter table cliente
drop column sexo;

alter table cliente
modify column rua varchar(34);

select * from cliente;

select * from cliente where cpf = '9876543211';

select * from cliente order by cpf asc;

select * from cliente order by cpf desc;
```

Fonte: Elaborada pelo autor, 2024

A manipulação do banco com os comandos abre um leque de opção bem vasto com ferramentas uteis, inserindo dados com insert ou buscando algum dado com select fazendo uso de especificação com o Where ou order by, deletando ou adicionando. Fica nítido a evolução no aprendizado ao desenvolver essa atividade avaliativa na forma em que se trabalha especificando os dados criando excluindo ou desenvolvendo a lógica e a solução para o problema conforme as necessidades para qual ela se apresenta.

## 6 DESENVOLVIMENTO PARA DISPOSITIVOS MOVEIS I

Durante o quinto semestre do curso foi desenvolvido um projeto interdisciplinar entre Desenvolvimento para Dispositivos Moveis, Tópicos especiais em sistema para Internet II e Projeto de prototipagem e usabilidade que consistia em desenvolver um aplicativo para um supermercado local de pequeno porte buscando atender suas necessidades como o público alvo além de atender funcionabilidades estabelecidas pelo professor gestor da matéria pois a atividade foi avaliativa.

Unindo conceitos de design, programação assíncrona e banco de dados o aplicativo foi prototipado seguindo normas e regras do design UX *“O design centrado no ser humano começa com uma boa compreensão das pessoas e das necessidades que o design deve atender.”* – Donald A. Norman em O Design do Dia a Dia, separando o projeto em etapas de concepção de design antes de iniciar etapa da programação, pois a usabilidade da aplicação é o principal fator em softwares mobile.

Foram utilizados os conceitos de prototipação por etapas, sendo elas baixa fidelidade aonde o protótipo são rabiscos em papel para definir primeiramente o fluxo de telas além de das possibilidades de corrigir incoerência, media fidelidade aonde nesse projeto, utilizamos um software de criação, que no caso foi o Figma, para definir a posição e a lógica das telas. Foi feito um protótipo de alta fidelidade, ficando quase fidedigno ao projeto final

Passando para o desenvolvimento da aplicação, ao inicia-la é apresentado uma tela de login já demonstrando os conceitos de UX e sua usabilidade além do quesito funcional de segurança do software, demonstrado na figura 6.1.

Figura 6.1 – Tela Inicial do Login



Fonte: Elaborada pelo autor, 2024

Essa tela como descrita anteriormente tem o objetivo de garantir a segurança dos dados de quem usar, o aplicativo possuindo 3 opções na tela: clicar no mercado X e entrar, caso ela apareça se o usuário já realizou o login, ou através do serviço de login do Google. Se confirmado, seguiria o fluxo para a próxima tela. Além das opções “esqueci a senha” e “criar conta”, que também dariam a opção do Google de cadastro automático, só pedindo ao usuário para confirmar o compartilhamento dos dados com o aplicativo. Não há um botão de “fazer login”, pois o mesmo seria iniciado ao entrar na página. Caso não ocorresse, iria pra essa tela, similar ao comportamento de aplicativo de entrega e encomenda de supermercado.

A próxima tela do projeto é o Home figura 6.2, como o nome sugere é o centro do fluxo de telas separando os produtos por categoria além de anteder as principais sugestões e comportamento de Design como margem e escolha de cores condizentes para tornar o uso mais confortável, graças ao uso de ferramentas do framework Ionic

e as ferramentas do plugin capacitor permitindo um auxílio muito grande na parte de criação por trazer componentes quase “prontos” a maior vantagem em se desenvolver com framework. *"O uso de frameworks permite que os desenvolvedores se concentrem mais nas funcionalidades específicas da aplicação, uma vez que muitos dos componentes fundamentais já estão prontos para uso, otimizando o tempo de desenvolvimento e garantindo maior eficiência. Essa reutilização de componentes, que são bem testados e configurados, é uma das maiores vantagens de utilizar frameworks no processo de desenvolvimento de software."* - Roger S Pressman em Engenharia de Software Uma abordagem profissional.

Figura 6.2 – Tela Inicial do Aplicativo



Fonte: Elaborada pelo autor, 2024

O framework utilizado como descrito no parágrafo anterior é o Ionic, framework em sua definição resumida seria um conjunto de ferramentas bibliotecas e componentes criados com a intenção de otimizar o processo de desenvolvimento.

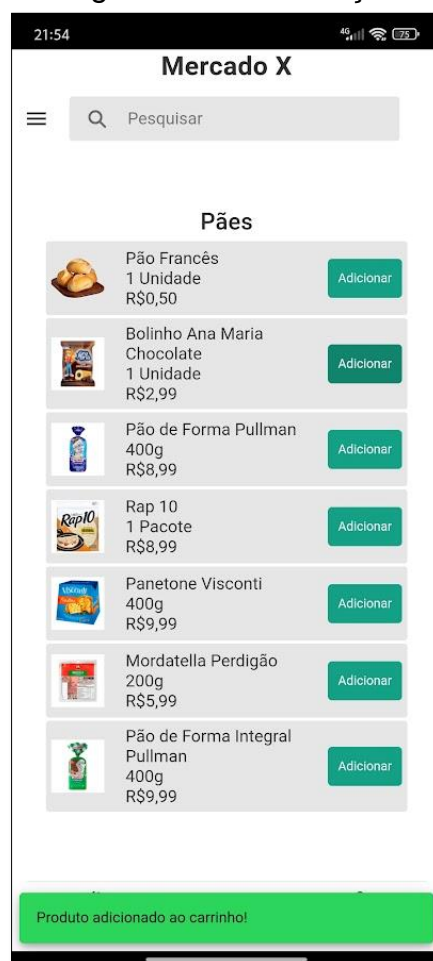
O Ionic tem seu maior destaque devido ao Capacitor, permite a criação de aplicações híbridas permitindo o acesso nativo a funcionalidades de dispositivos moveis, fornecendo uma camada de abstração consistente entre Application Programming Interface (API) nativas e TypeScript (TS).

Para programação da aplicação é utilizado o TypeScript que é uma linguagem de programação de código aberto desenvolvida pela Microsoft , sendo um superconjunto do JavaScript , ou seja todo código JavaScript roda no TypeScript permitindo um maior controle e flexibilidade ao software os recursos que valem ser mencionados são a tipagem estática e verificação de tipo, por isso que framework como o Angular e o Ionic preferem usar o TS em vez do JavaScript padrão, só por quesito de curiosidade quando um script TS executa o Navegador traduz ele para JavaScript e executa.

Retomando o assunto do Ionic a flexibilidade de APIs como o geolocation, notificação de push sem a necessidade de rescrever o código para cada plataforma android, ios ou web, já que foi mencionado esse assunto esse aplicativo de mercado foi desenvolvido com o foco no Android inclusive as figuras desses capítulos são capturas da tela de um smarthphone android e os testes e desenvolvimento foram executados em um servidor web local em um computador Windows, mostrando a versatilidade. Contudo, houve algumas modificações no projeto, e devido a isso, serão explicadas mais a frente nesse capítulo.

Depois de o usuário escolher a sessão pelos ícones no home ou no menu superior ele selecionara o produto em uma lista da categoria escolhida, ou clicando em algum item em promoção na tela inicial. Aparecerá então uma notificação informando que o item foi adicionado ao carrinho de comprar. Caso tenha algum erro aparecerá uma mensagem de notificação. Essa mensagem é gerada pelo API push.

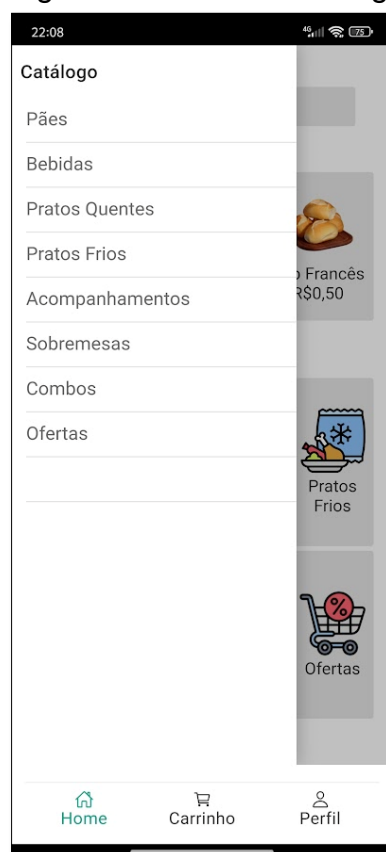
Figura 6.3 – Notificação



Fonte: Elaborada pelo autor, 2024

A notificação aparece logo abaixo após o usuário clicar no botão verde adicionar, ele não será direcionado para a tela do carrinho depois de adicionar o produto, pois pode ocorrer de o usuário querer adicionar mais produtos dessa mesma categoria. Todas as categorias são acessíveis de qualquer parte do catálogo ou no carrinho de compras para facilitar a navegação do aplicativo para que se tenha melhor interatividade com o usuário e o fluxo de telas.

Figura 6.4 – Menu Catálogo



Fonte: Elaborada pelo autor, 2024

Quando clicado no menu superior esquerdo do lado da barra de pesquisa aparece o menu catálogo com as categorias de produtos do aplicativo para fechar basta usar o gesto voltar ou o botão caso o celular utiliza navegação por botão, clicando na parte cinza ou em algum botão do menu inferior.

Depois de o usuário escolher seus produtos no catalogo, ele poderá verificar o que foi adicionado e continuar a compra ou excluir os itens indesejados no carrinho de compras que ao clicar no botão carrinho sera redirecionado para o mesmo.

Figura 6.5 – Carrinho de Compras



Fonte: Elaborada pelo autor, 2024

Estando no carrinho de compras como citado anteriormente o usuário visualiza seus itens e escolhe qual ação tomar baseado nos botões apresentados em tela, vale ressaltar que o lugar que se encontra “pagina”, estará tanto o símbolo quando o local em verde destacado na barra inferior.

Para realizar todo esse processo de manipulação dos dados do produto selecionado para o carrinho excluir o mesmo ou para passar para as próximas telas de pagamento e acompanhar o pedido foi feito um banco de dados não orientado utilizando programação assíncrona para que se tenha os métodos do banco funcionando sem depender um do outro dando flexibilidade para o projeto.

A API de banco de dados aprendida na disciplina e a outra opção sugerida no caso noSQL e Cordova apresentaram um ponto que os tornaram trabalhosos, pois ambos só funcionavam em dispositivos mobile, não sendo possível utilizar do servidor do localhost para testes e aperfeiçoamento do projeto. Isso porque que era necessário



dar build (processo em que se transforma o código do software em um programa funcional), em toda modificação.

A solução para isso foi mudar para uma API de banco de dados que fosse possível seus serviços funcionarem tanto no dispositivo móvel quando no ambiente de teste. Pesquisando sobre, chegou-se à conclusão de que seria viável utilizar o IndexedDB, um banco de dados que contemplaria as necessidades desse projeto, permitindo o seu funcionamento independente do ambiente em que esteja, e isso agilizou todo o processo de criação.

Quanto a API IndexedDB por ser assíncrona não interrompe a execução do restante do código, sendo baseado em JavaScript não se tem incompatibilidade com nenhum dispositivo sendo possível rodar no localhost, todos os dados armazenados do banco ficam salvo no ObjectStore do navegador ou no caso dos dispositivos móveis nos dados do usuário, mais para isso foi necessário instancia-lo no ObjectStore e definir qual versão do banco seria salvo como está descrito no trecho de código da figura 6.6.

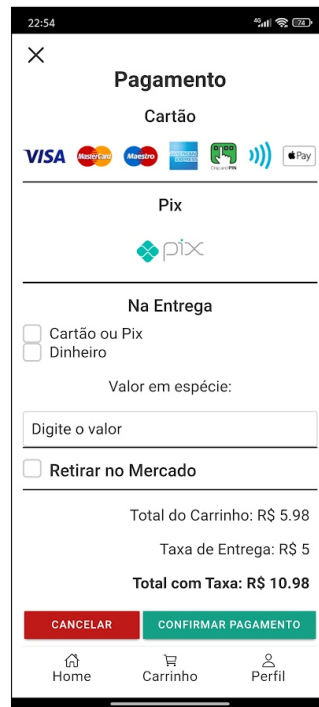
Figura 6.6 – Instanciando o Banco de Dados

```
private async openDB(): Promise<IDBDatabase> {  
  const request = indexedDB.open(this.dbName, 1);  
  return new Promise<IDBDatabase>((resolve, reject) => {  
    request.onsuccess = (event) => resolve((event.target as IDBRequest).result);  
    request.onerror = () => reject('Erro ao abrir o banco de dados');  
  
    request.onupgradeneeded = (event) => {  
      const db = (event.target as IDBRequest).result;  
  
      if (db.objectStoreNames.contains(this.pedidosStoreName)) {  
        db.deleteObjectStore(this.pedidosStoreName);  
      }  
  
      if (!db.objectStoreNames.contains(this.pedidosStoreName)) {  
        db.createObjectStore(this.pedidosStoreName, { keyPath: 'id', autoIncrement: true });  
      }  
  
      if (!db.objectStoreNames.contains(this.storeName)) {  
        db.createObjectStore(this.storeName, { keyPath: 'id', autoIncrement: true });  
      }  
    };  
  });  
}
```

Fonte: Elaborada pelo autor, 2024

Voltando ao processo do aplicativo de compra, ao clicar em comprar no carrinho é direcionado a pagina pedidos onde o usuário irá selecionar o método de pagamento descrito na figura 6.7. Também aparece o valor e a taxa de entrega, com os botões de confirmar ou cancelar ao final da página.

Figura 6.7 – Tela de Pagamento



Fonte: Elaborada pelo autor, 2024

Ao clicar em confirmar pagamento será direcionado para a tela de pedidos figura 6.8 onde é possível acompanhar o status do pedido e confirmar a entrega além do preço do pedido.

Figura 6.8 – Tela de Pedidos



Fonte: Elaborada pelo autor, 2024

A tela de pedidos é acessível pela pagina perfil, onde seu acesso pode ser feito pelo botão inferior na barra de navegação clicando em Perfil, depois clicando na opção Pedidos que aparece na tela.

Figura 6.9 – Tela de Perfil



Fonte: Elaborada pelo autor, 2024

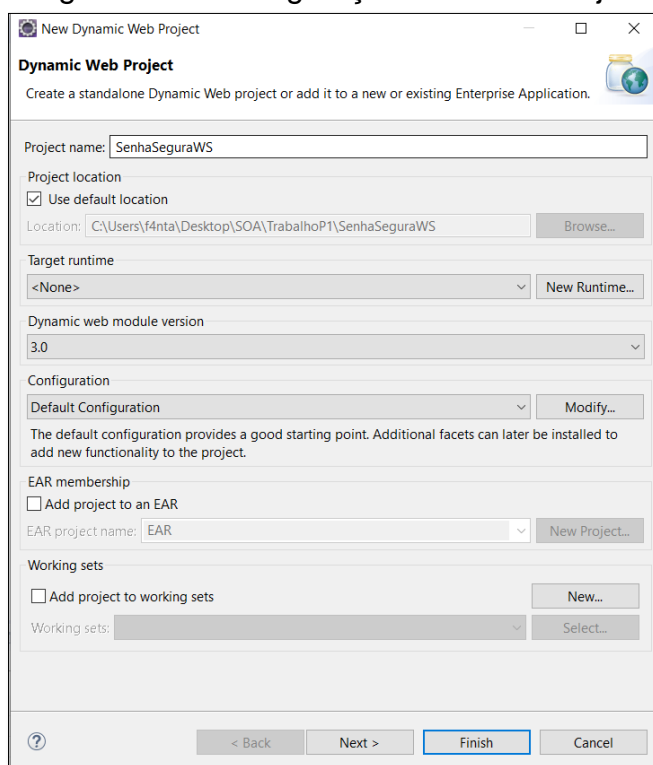
O aplicativo no final da disciplina se mostrou um grande desafio principalmente pela questão de design e adaptabilidade de API, porém foi desenvolvido com êxito cumprindo a atividade proposta e a emenda das matérias para as qual foi avaliada proporcionando uma experiência de desenvolvimento com design UX para dispositivos móveis bem próximo do que é praticado no mercado de trabalho.

## 7 ARQUITETURA ORIENTADA A OBJETO

A atividade desenvolvida nesse semestre foi um projeto avaliativo, no qual foi criada uma aplicação que consome outra aplicação utilizando o conceito de programação orientada a serviço. A linguagem de programação utilizada foi o Java e a IDE de desenvolvimento foi o Eclipse em sua versão 2019-12. Foi escolhida essa versão em virtude da compatibilidade com o servidor apache tomcat 9.0 que só funciona em sua plenitude nessa versão.

Criando o serviço utilizando a ferramenta Eclipse para facilitar e configurar as pastas e arquivos adicionais, como observado na figura 7.1.

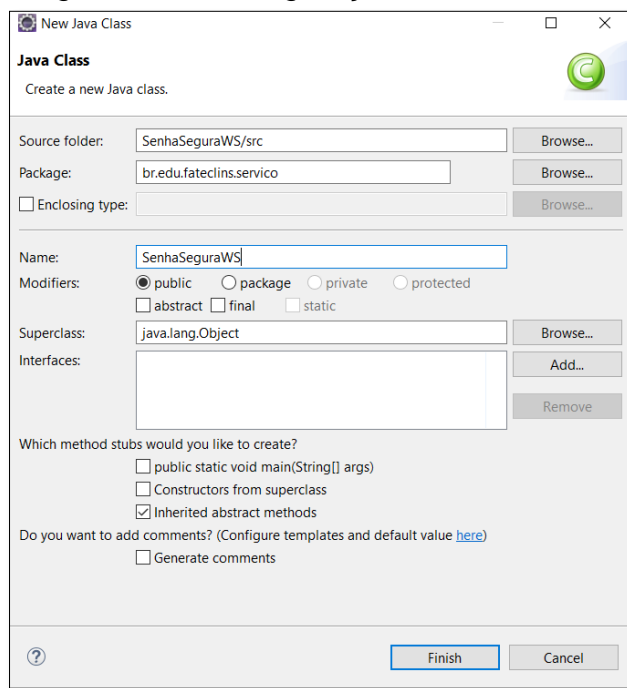
Figura 7.1 – Configuração Inicial do Projeto



Fonte: Elaborada pelo autor, 2025

Após definir a configuração do projeto e a criação de pastas de arquivos além do pacote para organizar o projeto, continuou-se criando uma classe java na qual foi inserido o código do serviço e foi realizado ajustes em sua configuração para atender aos requisitos, como observado na figura 7.2

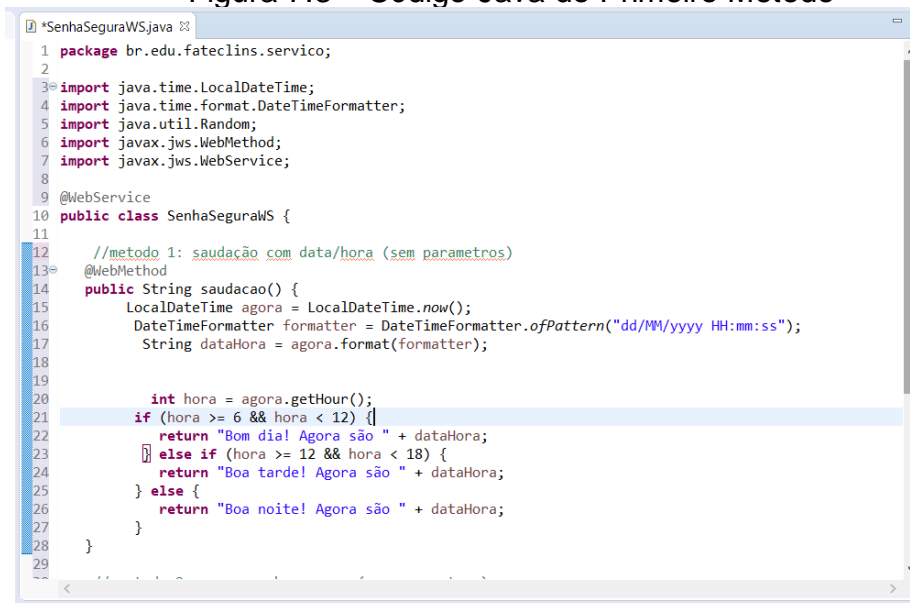
Figura 7.2 – Configuração da Classe Java



Fonte: Elaborada pelo autor, 2025

Após definidas as configurações da classe Java, foi criada uma lógica de programação e inseridos os dois métodos desse serviço: um de saudação com data e hora, sem necessidade de envio de parâmetros, e o segundo, um método para gerar senha segura, onde são recebidos os parâmetros informados pelo usuário.

Figura 7.3 – Código Java do Primeiro Metodo



Fonte: Elaborada pelo autor, 2025

Figura 7.4 – Código Java do Segundo Método

```

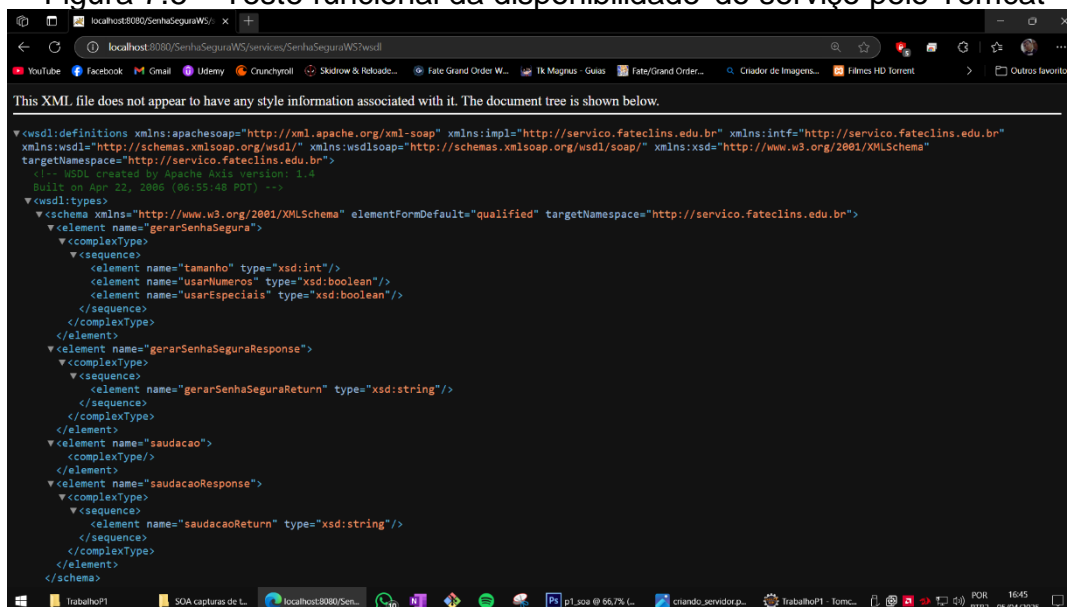
30 // metodo 2: gerar senha segura (com parametros)
31 @WebMethod
32 public String gerarSenhaSegura(int tamanho, boolean usarNumeros, boolean usarEspeciais) {
33     String caracteres = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz";
34     if (usarNumeros) caracteres += "0123456789";
35     if (usarEspeciais) caracteres += "!@#$%^&*()_+~[]{}|;:,.<.>?";
36
37     Random random = new Random();
38     StringBuilder senha = new StringBuilder();
39     for (int i = 0; i < tamanho; i++) {
40         senha.append(caracteres.charAt(random.nextInt(caracteres.length())));
41     }
42     return senha.toString();
43 }
44 }

```

Fonte: Elaborada pelo autor, 2025

Foi testado a conexão do serviço com seus métodos utilizando o Tomcat e o o programa SOAP(Simple Object Access Protocol em português Protocolo Simples de Acesso a Objetos) para verificar se o serviço estava sendo consumido com seus métodos. Através dele foi possível testar e simular um cliente consumindo a aplicação acessando o localhost no navegador e analisando se o mesmo está ativado conforme observado na figura 7.5.

Figura 7.5 – Teste funcional da disponibilidade do serviço pelo Tomcat

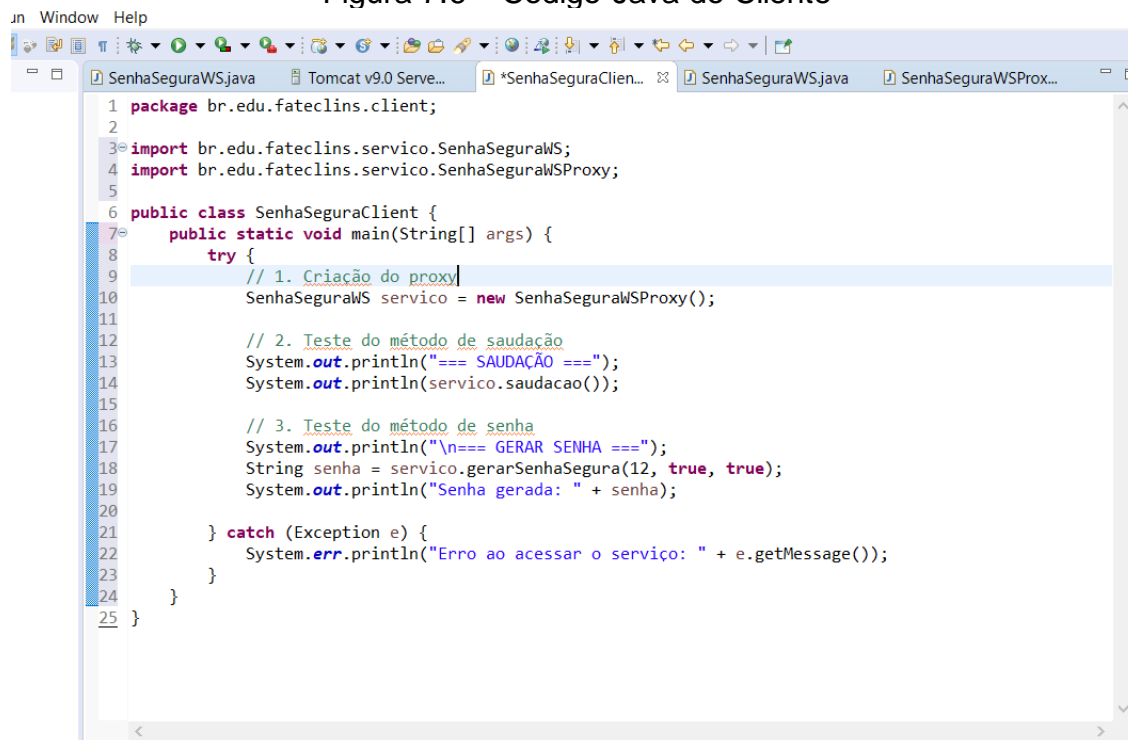


Fonte: Elaborada pelo autor, 2025

Após verificar o funcionamento e a disponibilidade do serviço, foi criado um novo projeto o cliente, para consumir o serviço, ajustando as configurações para atender à demanda do serviço e possibilitar sua execução. Dentro do projeto cliente,

foi criada a classe Java para inserir o código que irá consumir o serviço, conforme observado na figura 7.6.

Figura 7.6 – Código Java do Cliente

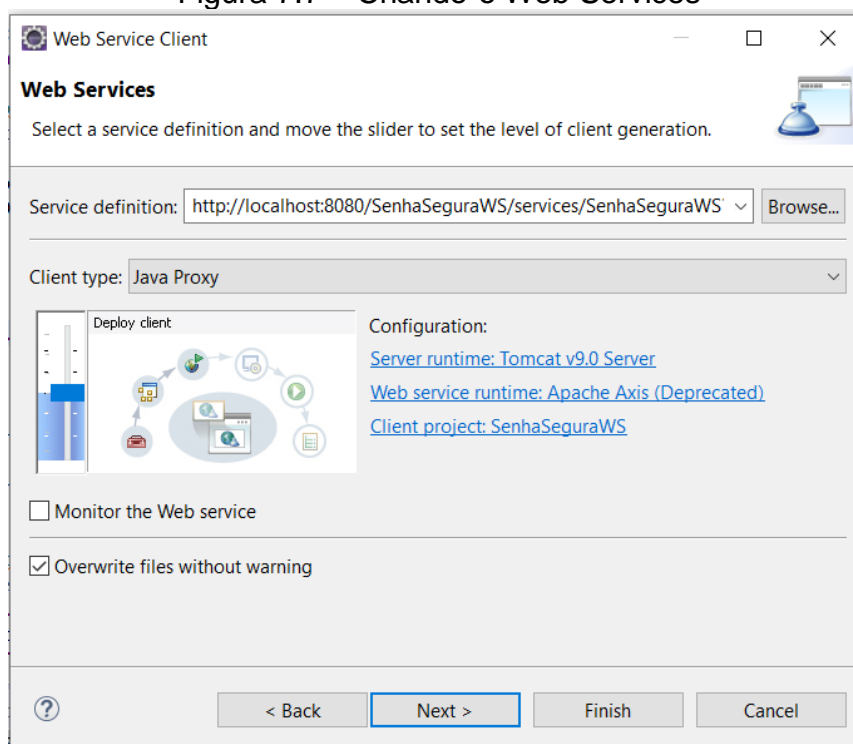


Fonte: Elaborada pelo autor, 2025

O código Java da figura é uma estrutura de execução na qual ele vai tentar executar o serviço, consumindo a aplicação. Caso não consiga, o mesmo vai exibir uma mensagem de erro. No começo da execução, é feita a ligação do mesmo com o serviço. “A arquitetura orientada a serviços é um estilo arquitetural que promove o reuso de componentes de software por meio de serviços bem definidos que podem ser publicados, descobertos e utilizados em uma rede.” - Roger S. Pressman, em Engenharia de Software: Uma abordagem profissional.

Foi criado um web service no projeto cliente, copiando o link do serviço obtido no navegador, e feita a ligação utilizando o tipo Java Proxy, fornecido na ferramenta Eclipse, conforme descrito na figura 7.7. Graças ao Eclipse, boa parte da configuração, arquivos complementares e pacotes adicionais são criados automaticamente e organizados de forma funcional.

Figura 7.7 – Criando o Web Services



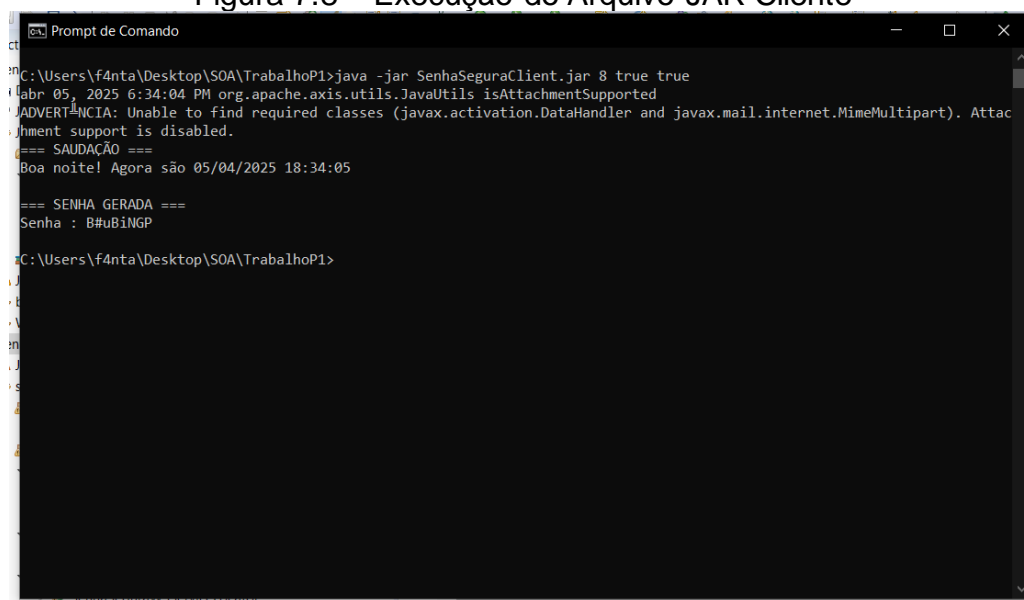
Fonte: Elaborada pelo autor, 2025

Dando continuidade ao projeto é gerado um arquivo JAR, que pode ser executado fora da IDE transformando o projeto do cliente em um programa executável. É feito o acesso a pasta onde está executado o arquivo JAR gerado e executa-se o arquivo conforme demonstrado na figura 7.8, usando-se o comando: `java -jar SenhaSeguraClient.jar 8 true true`.

Esse comando começa com a declaração do tipo e o parâmetro do arquivo executado seguido de seu nome e os parâmetros passado para ele, mesmo dando um erro aparante no terminal devido a maquina onde estava sendo executada estando com uma versão mais recente, a aplicação foi executada e o serviço criado com os dois métodos consumidos.



Figura 7.8 – Execução do Arquivo JAR Cliente



```

C:\Users\f4nta\Desktop\SOA\TrabalhoP1>java -jar SenhaSeguraClient.jar 8 true true
05/05/2025 6:34:04 PM org.apache.axis.utils.JavaUtils isAttachmentSupported
WARNING: Unable to find required classes (javax.activation.DataHandler and javax.mail.internet.MimeMultipart). Attachment support is disabled.
=== SAUDAÇÃO ===
Boa noite! Agora são 05/04/2025 18:34:05

=== SENHA GERADA ===
Senha : B#uBINGP

C:\Users\f4nta\Desktop\SOA\TrabalhoP1>
```

Fonte: Elaborada pelo autor, 2025

Ao final do projeto, pode-se analisar que o desenvolvimento da atividade representou um percurso importante para a construção de uma base sobre o funcionamento dos conceitos e a execução da programação orientada a serviços, na qual se programa com o intuito de que quem vá consumir aquela aplicação seja outra aplicação, um conceito muito utilizado na web em geral criando assim uma base lógica e funcional.

## 8 MANUAL DO PORTIFOLIO DIGITAL

A inspiração das cores de design foi baseada em referências de álbuns musicais com essa temática, além da aplicação da psicologia das cores. O uso do branco e preto foi escolhido por trazer um tom de elegância e refinamento, enquanto o vermelho se destaca por ser uma cor que representa atenção. "*O vermelho é uma cor que chama atenção imediatamente, sendo associada à excitação, energia e urgência.*" - Eva Heller, *Psicologia das Cores: Como as Cores Afetam a Emoção e a Razão*,

Figura 8.1 – Paleta de cores do portfólio



Fonte: Elaborada pelo autor, 2025

Conforme observado na imagem 8.1, as cores são tons de cinza e branco. A ideia da organização de cores do portfólio é criar um aspecto de degradê, trazendo a atenção do leitor para o meio do site, com rodapé e menus pretos com elementos brancos, usando do vermelho para chamar a atenção do leitor para alguns itens fora desse degradê, conforme mostrado na figura 8.2.

Figura 8.2 – Portfólio mobile



Fonte: Elaborada pelo autor, 2025

A figura 8.2 mostra os elementos em vermelho. Apesar de parecer uma quebra do design, é justamente por meio dessa que alguns elementos são mais destacados que outros, fazendo a visão do leitor se sobrepôr a eles para chamar a atenção.

Durante o começo do desenvolvimento foi utilizada a ferramenta de criação de material visual Figma para a criação do protótipo inicial. O protótipo, em si, tinha o aspecto de baixa fidelidade, onde o principal ponto abordado era o formato e a posição do conteúdo disposto, conforme demonstrado na figura 8.3, as cores e fontes dispostas.

Figura 8.3 – Prototipo do Figma



Fonte: Elaborada pelo autor, 2025

Observando o protótipo, é notável que todos os aspectos visuais foram alterados, a cor sendo a principal mudança. O início do projeto seria um portfólio usando o estilo e fonte similar a um jornal impresso, com caligrafias e disposição do conteúdo e visualização usando de colunas e tons de cinza e preto em seus componentes. Um aspecto mantido na versão final foi o escopo de landing page com o botão home, além do método de visualização do menu suspenso fixo.

Mudando o estilo de tons de cinza de material impresso para um estilo mais elegante e colorido também se deve ao fato da disposição de outros elementos, que teriam que ser nessa escala de cor, causando uma apresentação irreal do conteúdo dos projetos e outros elementos coloridos.

Conforme notado, as cores utilizadas são #F20505, #FFFFFF, #ECEBEB e #323232 escalas de preto e branco e o elemento vermelho para itens de destaque.

Além das cores, outro ponto a ser observado é a escolha e uso de fonte. No projeto foi utilizada a fonte Roboto.

Figura 8.4 – Fonte Roboto

## Roboto(Variable)

Fonte: Elaborada pelo autor, 2025

A escolha da fonte Roboto se deu pela sua tipografia moderna, limpa e de alta legibilidade, além de ter uma estética quadrada, lembrando uma tipografia meio robótica e elegante. Sua estrutura sans-serif combina com a proposta estética do portfólio, com moldes mais geométricos, transmitindo objetividade e profissionalismo, sem comprometer a leitura em diferentes tamanhos de tela, graças ao seu bom espaçamento entre os caracteres.

Optou-se por ela também devido à sua compatibilidade universal, sendo facilmente importada para o projeto pela integração com o Google Fonts, além de possuir uma licença livre, que permite seu uso seguro.

Com a fonte definida do portfólio sendo a Roboto, meu logo teve inspiração nela devido à sua tipografia meio robótica, misturando as cores definidas da paleta de cor com um elemento que remetesse a algum aspecto da programação.

Figura 8.5 – Logo do Portfólio



Fonte: Elaborada pelo autor, 2025

A logo, conforme observada, faz uso dela para formar LGM, as iniciais do nome Luccas Godoy Miazza. Além disso, os elementos na frente remetem a um terminal de sistema, algo simples, porém com significado, na qual as letras são a unidade de armazenamento e os símbolos, os prompt de comando, como são chamados, indicam em que se deve inserir o código, dando destaque para o vermelho do cursor, à espera ou mostrando no qual está digitando.

A cor vermelha foi usada para dar foco nisso, mostrando o indicativo de código, além de seu fundo preto destacando todos os elementos das letras, tendo um cuidado para não deixar algo muito exagerado, mantendo a simplicidade, podendo posicioná-la no site, deixando visível no menu, sem atrapalhar a leitura.

O desenvolvimento do portfólio, no quesito de desenvolver em uma resolução específica, foi uma mescla, fazendo uso de parâmetros de medida como vh (view height) e vw (view width) no código CSS, que são baseados no view da página, se ajustando independente da resolução, e para ajustes finos dependendo da resolução, conforme mostrado no exemplo da figura 8.6 no trecho de código do CSS.

Figura 8.6 – Trecho de Código CSS

```
.sobre{  
    align-self: center;  
    max-width: 64vw;  
    max-height: 600px;  
    padding-top: 20px;  
}
```

Fonte: Elaborada pelo autor, 2025

Para facilitar a organização dos arquivos e pastas, foi criado arquivo de CSS separado: um para o modal, media query e index, esse sendo o arquivo principal, os demais são para suas respectivas funções, permitindo uma melhor organização do documento, além de incluir comentários e sua estruturação, seguindo o padrão de arquivo HTML de cima para baixo.

Além da aparência web, com menu hambúrguer, conforme já mostrado na figura 8.2, para aproveitamento de espaço de layout, o menu fica na horizontal uma vez que a área de visualização se torna maior.

Figura 8.7 – Portfólio Desktop



### Sobre Mim

Sou técnico em Informática pela ETEC de Lins e atualmente estudo Sistemas para Internet na FATEC Lins, onde mergulho nas tecnologias mais relevantes do mercado. Apaixonado por desenvolvimento web e design, estou sempre aprendendo e aplicando novas tendências para criar soluções que unam funcionalidade e criatividade. Se você busca um profissional dedicado para dar vida ao seu projeto digital, adoraria conversar e contribuir com essa jornada!

Fonte: Elaborada pelo autor, 2025

Assim como descrito anteriormente, foi utilizada a aplicação de cores mais claras para chamar a atenção do leitor, fazendo com que se note logo de cara a cor branca e o texto da seção "Sobre mim", além de transmitir um tom estético mais limpo e profissional, com figuras compostas por elementos mais quadrados, ressaltando os detalhes em vermelho, o que dá um ar de classe.

Outro elemento presente neste portfólio foi uma abordagem com JavaScript, chamada de modal, um conteúdo que se expande ao se clicar em um elemento na tela. Esse recurso foi projetado para acomodar os projetos desenvolvidos durante os semestres e manter o site com um design one page, em que todo o conteúdo se encontra em uma única página, com o auxílio de recursos combinados de HTML e CSS. Basta observar as figuras 8.7 e 8.8 para compreender seu funcionamento.

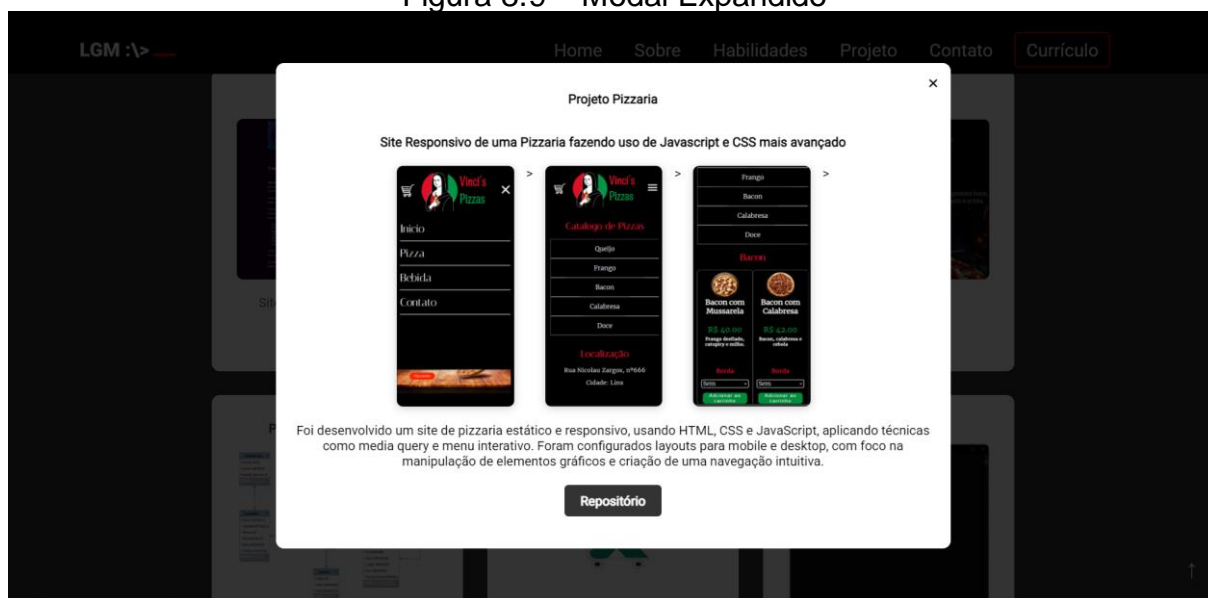
Figura 8.8 – Div Modal no Site



Fonte: Elaborada pelo autor, 2025

Para a visualização de conteúdo, o leitor irá precisar interagir com o botão "Ver Mais". Quando clicado, o mesmo vai expandir-se para mostrar o conteúdo nele contido referente ao projeto em que interagiu.

Figura 8.9 – Modal Expandido



Fonte: Elaborada pelo autor, 2025

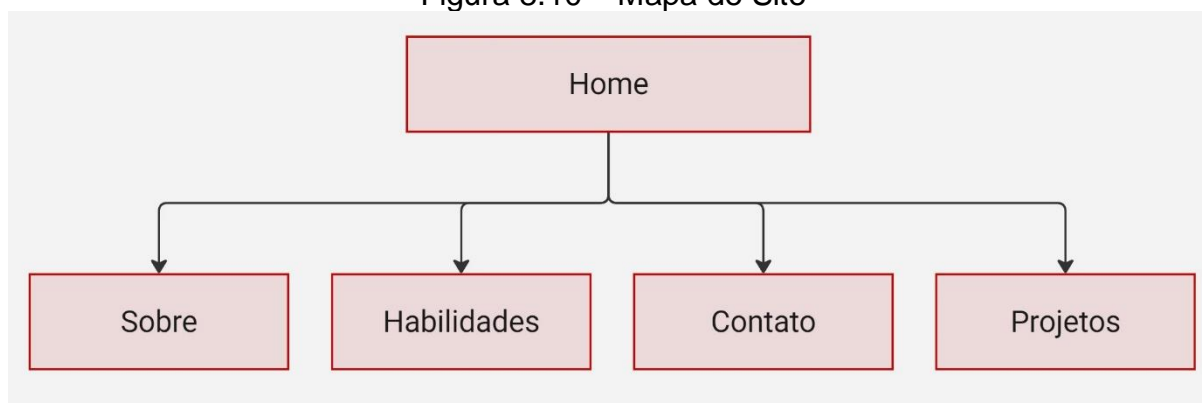
O fundo escurece para melhor foco no conteúdo da página, além de imagens do projeto, uma breve descrição. Também há o botão "Repositório" da plataforma GitHub, onde contém o projeto para visualização de código e seu funcionamento no mesmo, graças ao recurso do GitHub Pages presente no GitHub. Contudo, alguns projetos, como banco de dados, por serem projetos específicos, não há necessidade de demonstrar ele executando na web. Nesse caso, em vez do repositório, é disponibilizado um botão de imagem para ver o projeto e um botão para visualizar os arquivos de projeto.

Além disso, foi codificado o comportamento do elemento para fechar clicando no "x" no canto superior ou clicando na área escurecida. Ele se comporta e se ajusta automaticamente graças ao media query que possui.

A navegação do portfólio é feita no formato de organização páginas One Page, na qual todo o site fica disposto em uma única página, separado em seções, dispondo seu conteúdo. É possível observar na Figura 8.10 como foram organizadas as seções no mapa do site, estando tudo na mesma página.



Figura 8.10 – Mapa de Site



Fonte: Elaborada pelo autor, 2025

Desenvolver o portfólio foi uma tarefa desafiadora, não pela complexidade da codificação, mas principalmente pelo processo criativo envolvido. Criar algo que refletisse preferências pessoais de design, ao mesmo tempo em que seguisse normas e padrões estabelecidos, exigiu uma análise sobre como organizar e estruturar um projeto sem um objetivo funcional específico. Com isso, foi possível planejar, testar e replanejar diversas vezes, ajustando o resultado conforme novas ideias surgiam ao longo do desenvolvimento.

Dessa forma, o portfólio foi finalizado com sucesso, atendendo às diretrizes propostas e, acima de tudo, demonstrando as principais habilidades técnicas, além da capacidade de personalização e adaptação na construção de projetos web.

## 9 CONSIDERAÇÕES FINAIS

A construção do portfólio acadêmico representa o encerramento do ciclo na formação no curso de Sistemas para Internet, sendo um reflexo direto do conhecimento e da evolução profissional e pessoal desenvolvido ao longo dos semestres de graduação.

Cada projeto aqui apresentado foi essencial para a consolidação de uma base teórica e prática desenvolvida durante das disciplinas estudadas, reforçando a importância da prática e na fixação do aprendizado. O desenvolvimento das atividades e dos projetos propostos durante o curso proporcionou uma evolução gradual, tendo início com estruturas básicas com HTML e CSS até o uso de frameworks modernos além da manipulação de banco de dados, passando por conceitos fundamentais de base como programação orientada a serviços, aplicação web, desenvolvimento mobile e virtualização de sistemas e aplicação, com cada código diagrama ou erro sendo um pequeno passo para o fortalecimento e aprimoramento de habilidade técnica.

Além do conteúdo técnico, a vivência acadêmica proporcionou conhecimento e experiências para o desenvolvimento como profissional, os projetos em grupo, as pesquisas e aplicações desenvolvidas em sala ajudaram a desenvolver uma postura crítica e mais alinhada a realidade do mercado de trabalho.

Com o portfólio digital eficaz e revelante sendo uma ferramenta de apresentação dos conhecimentos e recursos obtidos e desenvolvidos durante o curso, podendo ser utilizado futuramente como uma ferramenta de apoio na inserção no mercado de trabalho demonstrando minhas habilidades e capacitação.

Finalizar, este portfólio acadêmico não representa somente a conclusão desse curso acadêmico, mas também um novo início como profissional da área de tecnologia, com uma boa base em desenvolvimento, raciocínio lógico criatividade e senso crítico. Toda essa trajetória está documentada nesse trabalho simbolizando o compromisso com a aprendizagem e com a constante evolução como profissional.

## REFERÊNCIAS

ESTRELLA, C. **CSS inline, interno ou externo**: entenda a diferença entre eles. [S. l.]: Hostinger Tutoriais, 2022. Disponível em: <https://www.hostinger.com.br/tutoriais/diferenca-entre-estilos-css>. Acesso em: 15 maio 2025.

ELMASRI, R.; NAVATHE, S. B. **Sistemas de Banco de Dados**. 7ª ed. São Paulo: Pearson, 2019.

FLANAGAN, D. **O guia definitivo do JavaScript: o melhor da linguagem**. 6. ed. São Paulo: O'Reilly, 2013.

FRANCO, G. **Como escrever para a Web**. 1ª ed. Austin: Knight Center for Journalism in the Americas, 2009. Disponível em: <https://journalismcourses.org/pt-br/ebook/como-escrever-para-a-web/>. Acesso em: 1 abr. 2025.

HELLER, EVA. **A psicologia das cores: como as cores afetam a emoção e a razão**. Tradução de Maria Lúcia Lopes da Silva. 1. ed. São Paulo: Editora Gustavo Gili, 2012.

NORMAN, D. A. **O Design do Dia a Dia**. 1ª ed. Rio de Janeiro: Rocco, 2006.

PRESSMAN, R. S.; MAXIM, B. R. **Engenharia de Software: Uma Abordagem Profissional**. 8ª ed. Porto Alegre: AMGH, 2016.

TANENBAUM, A. S.; FEAMSTER, N. **Redes de Computadores**. 6ª ed. Porto Alegre: Bookman, 2021.