

Escola Politécnica da Universidade de São Paulo



PMR 3304 -Sistemas da Informação:

WebSite com Gerador de Sites Estáticos

Autor:

Mateus Matzkin Gurza; Nusp: 6010281

Professores: Luís de Sá & Marcos Tsuzuki

São Paulo

2024

SUMÁRIO

| | |
|--|----|
| 1. INTRODUÇÃO..... | 2 |
| 2. OBJETIVOS..... | 2 |
| 3. METODOLOGIA..... | 3 |
| 3.1. Front Matter e Metadados..... | 3 |
| 3.2. Organização do Sistema de Arquivos..... | 4 |
| 3.3. Partials e Templates Base..... | 4 |
| 3.4. Utilização da Sintaxe {{ }}..... | 4 |
| 4. RESULTADOS..... | 5 |
| 4.1 Páginas..... | 5 |
| 4.2 Layouts..... | 5 |
| default/baseof.html..... | 5 |
| default/single.html..... | 6 |
| partials/head.html..... | 6 |
| partials/header.html..... | 6 |
| partials/footer.html..... | 6 |
| experience/list.html..... | 7 |
| experience/single_IC.html..... | 7 |
| experience/single_polijunior.html..... | 7 |
| schooling/list.html..... | 8 |
| schooling/single_band.html..... | 9 |
| schooling/single_btc.html..... | 9 |
| schooling/single_poli.html..... | 9 |
| index.html..... | 10 |
| 4.3 CSS..... | 11 |
| css/styles2.css..... | 11 |
| 4.4 O Site..... | 16 |
| Figura 1: Home Page..... | 16 |
| Figura 2: Lista de Conteúdos em “Experiência”..... | 16 |
| Figura 3: Lista de Conteúdos em “Formação”..... | 17 |
| Figura 4: Página “Poli Júnior” na Lista da figura 2..... | 17 |
| Figura 5: Página com media query aplicada..... | 18 |

CONTEÚDO

Site:

<https://hugowebste-gurza-mateus-20c71c39c14def7a4a0e4fe33807464828fdca.e.usp.br/>

GitLab:

<https://gitlab.uspdigital.usp.br/gurza.mateus/hugowebste>

1. INTRODUÇÃO

Este relatório tem como objetivo documentar a construção do website pessoal, elaborado durante a disciplina de Sistemas da Informação (PMR 3304), utilizando o framework Hugo. O projeto teve como propósito desenvolver um site estático que funcione como um currículo online, exibindo informações sobre minha experiência profissional, formação acadêmica, interesses e outras questões relevantes. Ao longo do processo, buscou-se não apenas a criação do site em si, mas também o aprendizado das ferramentas e técnicas envolvidas na sua construção e publicação.

A escolha por um site estático se deu pelas suas características de simplicidade e eficiência em termos de carregamento e manutenção. Diferente de um site dinâmico, que depende de um servidor e banco de dados para gerar páginas de forma dinâmica em tempo real, o site estático gera as páginas previamente, tornando-as prontas para exibição imediata ao usuário. Esse tipo de estrutura é ideal para sites com conteúdo relativamente fixo, como um portfólio ou currículo, onde as atualizações são pontuais e não há a necessidade de interação constante do usuário.

Além de compreender as funcionalidades do framework Hugo, o projeto também foca em se familiarizar com o processo de deployment, ou seja, a publicação do site na web, utilizando as ferramentas disponíveis, como o GitLab Pages. O estudo das lógicas por trás da estrutura de arquivos, personalização de leiautes e inclusão de CSS faz parte dos objetivos técnicos do trabalho.

O documento está organizado em três seções principais:

- **Objetivo:** onde será discutida a importância do projeto e suas metas;
- **Metodologia:** com uma explicação detalhada sobre o Hugo e os passos seguidos para a criação e personalização do site;
- **Resultados:** que incluirá a listagem dos códigos de leiautes, arquivos CSS, figuras da página inicial e de uma página com media query para diferentes tamanhos de tela, além do link para o repositório no GitLab e o endereço do site no GitLab Pages.

2. OBJETIVOS

O principal objetivo deste projeto é proporcionar um aprendizado aprofundado sobre o uso do framework Hugo para a construção de sites estáticos. O Hugo é uma ferramenta amplamente utilizada pela sua eficiência e rapidez na geração de páginas, sendo uma escolha robusta para criar websites que não exigem interações dinâmicas constantes, como um site pessoal ou portfólio. A familiarização com este framework é relevante, pois permite o desenvolvimento de páginas que são rápidas, fáceis de atualizar, manter e publicar, sem a necessidade de servidores complexos ou banco de dados, o que reduz significativamente a necessidade de infraestrutura técnica e simplifica o processo de manutenção.

Além do aprendizado técnico relacionado ao Hugo, outro objetivo central do projeto é a criação de uma página pessoal bem estruturada e funcional, que centralize informações importantes como currículo, formação acadêmica, interesses e outros aspectos relevantes para a apresentação pessoal. Um site de fácil atualização e manutenção é essencial em um contexto onde o dinamismo e a rapidez na disponibilização de informações são valiosos. A capacidade de atualizar rapidamente os conteúdos e realizar mudanças no layout, sem depender de processos complexos, agrega valor prático ao uso de um site estático como ferramenta de comunicação pessoal.

Um dos desafios propostos no projeto foi construir o site a partir do zero, sem a utilização de temas prontos. A meta era não apenas desenvolver um site funcional, mas também entender as nuances da estruturação dele, abordando aspectos mais amplos da construção da estrutura que o compõe. Assim, foi trabalhado o aprendizado completo acerca das diversas técnicas de design, codificação, e interligação das diferentes referências, arquivos e páginas do site. Um exemplo desse aprendizado mais aprofundado foi a elaboração de “front matter” em páginas de “markdown” para utilizar metadados como forma de correlacionar páginas, imagens e layouts.

Por fim, o projeto buscou organizar as informações de maneira clara e objetiva, tornando o conteúdo atraente para quem visita o site. O desafio de criar um layout que seja ao mesmo tempo funcional e esteticamente agradável é essencial para garantir que o site pessoal cumpra seu papel: despertar o interesse e fornecer informações de forma acessível e coerente. O desenvolvimento do site sem temas pré-configurados reforça a capacidade de adaptar o conteúdo às necessidades específicas, possibilitando maior controle sobre cada aspecto da página.

3. METODOLOGIA

Neste projeto, o framework **Hugo** foi utilizado para a construção de um website estático. Hugo é uma ferramenta poderosa para a criação de sites, pois gera páginas estáticas rapidamente a partir de arquivos de conteúdo e layouts predefinidos. A seguir, será descrito o funcionamento básico do Hugo, incluindo a utilização de **front matter**, organização do sistema de arquivos, definição de **partials**, e a dinâmica de construção e reutilização de componentes como o header.

3.1. Front Matter e Metadados

Cada página ou post criado no Hugo começa com uma seção chamada **front matter**, que é utilizada para definir metadados sobre o conteúdo. Esses metadados permitem configurar diversas propriedades e comportamentos da página, como o título, data de publicação, rótulos (tags), categorias, entre outros. No caso deste projeto, utilizamos a sintaxe TOML, delimitada por **+++**.

Exemplo de front matter em TOML:

```
+++

title = "Página Inicial" #refere-se ao título da página, pode ser obtido por {{.Title}}

date = "2024-10-09" #refere-se à data de criação da página

draft = false #define se a página irá ou não aparecer ao rodar o servidor hugo, caso true, a página não aparece

description = "Bem-vindo ao meu site pessoal" #descrição

tags = ["Currículo", "Sobre"] #etiquetas

layout = single_exemplo #Faz referência ao layout para o qual a página redireciona

image = exemplo.png # Faz referência à imagem que pode ser obtida pelo método {{.Params.image}}
```

```
+++
```

3.2. Organização do Sistema de Arquivos

O sistema de arquivos do Hugo segue uma estrutura bastante organizada, o que facilita o desenvolvimento e a manutenção do site. Os diretórios principais incluem:

- **content/**: onde ficam os arquivos de conteúdo do site, como páginas e posts. Cada arquivo pode ser estruturado em subdiretórios, representando diferentes seções do site (ex. `content/blog/`, `content/about/`).
- **layouts/**: pasta onde são armazenados os templates de layout, que definem como o conteúdo será exibido. Nesta pasta, encontramos templates para páginas específicas e também componentes reutilizáveis, como o header e footer.
- **static/**: armazena os arquivos estáticos, como imagens, arquivos CSS, JavaScript, e outros recursos que não mudam frequentemente e são servidos diretamente ao usuário.
- **config.toml**: arquivo de configuração principal, onde são definidas as configurações globais do site, como o título, a baseURL e outros parâmetros de personalização.

3.3. Partials e Templates Base

Uma das grandes vantagens do Hugo é a possibilidade de reutilizar código através de *partials*. Um *partial* é um bloco de código que pode ser incluído em diferentes partes do site, como o header, footer, ou qualquer componente repetido. Isso facilita a manutenção do site, pois é possível centralizar as alterações em um único arquivo.

Os *partials* são definidos no diretório `layouts/partials/` e podem ser chamados em outros templates utilizando a sintaxe `{{ partial "nome-do-partial.html" . }}`. Por exemplo, o header do site pode ser definido como um *partial* e incluído em todas as páginas: `{{ partial "header.html" . }}`.

Além dos *partials*, Hugo também utiliza uma estrutura chamada *base template* para definir layouts gerais, permitindo que diferentes páginas compartilhem a mesma estrutura visual (ex. header, footer, barra lateral). Um exemplo comum de *base template* é o `baseof.html`, onde os componentes globais do site são definidos. Dentro de `baseof.html`, usamos o comando “*block*” para definir áreas específicas que serão preenchidas pelos templates das páginas individuais, garantindo assim que todas as páginas herdem a mesma estrutura básica.

3.4. Utilização da Sintaxe {{ }}

A sintaxe `{{ }}` é fundamental no Hugo, pois é utilizada para chamar variáveis, funções e templates. Essa sintaxe, baseada na linguagem Go, é o que permite o dinamismo na construção de páginas estáticas.

- Por exemplo, para chamar o título da página, utilizamos:
`<h1>{{ .Title }}</h1>`
- Ou, para criar um link dinâmico para uma página específica:
`Leia mais`

Essa sintaxe também é usada para carregar partials e outros recursos compartilhados, como o exemplo do header mencionado anteriormente. A reutilização de componentes como o header e o footer é essencial para manter a consistência visual e estrutural do site, além de facilitar futuras atualizações.

A metodologia aplicada neste projeto foca na compreensão completa do funcionamento do Hugo, desde a estrutura de arquivos e organização do conteúdo até a definição de layouts reutilizáveis. A utilização de partials e templates base garantiu um desenvolvimento mais eficiente e modular, enquanto a exploração da front matter e da sintaxe `{{ }}` permitiu uma personalização fina de cada página do site, contribuindo para a criação de um website pessoal robusto e fácil de manter.

4. RESULTADOS

O resultado deste projeto foi o site:

<https://hugowebste-gurza-mateus-20c71c39c14def7a4a0e4fe33807464828fdca.e.usp.br/>

Publicado por meio do GitLab Pages.

O código fonte do site e a arquitetura completa do site no framework hugo podem ser encontrados tanto no GitLab quanto no GitHub associados ao e-mail gurza.mateus@usp.br.

- GitLab: <https://gitlab.uspdigital.usp.br/gurza.mateus/hugowebste>
- GitHub: <https://github.com/MatzkinGurza/MyWebProfile>

4.1 Páginas

O site possui, no total, 8 páginas, entre elas temos:

- Uma página inicial (home) no layout `index.html`.
- Uma página sobre mim (about) no layout default de `single.html`.
- Duas páginas de listagem de conteúdos no layout `list.html`.
 - A primeira se refere à formação acadêmica e se encontra dentro da path: `layouts/schooling/`. Esta lista aponta para 3 páginas de conteúdo distintas.
 - A segunda se refere a experiências práticas importantes e se encontra dentro da path: `layouts/experience/`. Esta lista aponta para 2 páginas de conteúdo distintas.
- Cinco páginas de conteúdo específico, as quais podem ser acessadas pelas listas, todas com seus respectivos layouts no formato `single_exemplo.html`.

4.2 Layouts

Abaixo segue uma listagem dos códigos de layouts (layouts):

- **default/baseof.html**

```
<html lang="en"></html>

{{ partial "head.html" . }}

<body>
  {{ partial "header.html" . }}
  <main>
    {{ block "main" . }}
    {{ end }}
  </main>
```

```

    {{ partial "footer.html" . }}
</body>
</html>

```

- default/single.html (página “sobre mim”)

```

{{ define "main" }}
<section class="exp-main" id="about">

    <div class="exp-box">
        <div class="exp-img">
            
        </div>
        <article class="exp-text">
            {{ with .GetPage "about" }}
                {{ .Content }}
            {{ end }}
        </article>
    </div>

</section>

{{ end }}

```

- partials/head.html

```

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>{{ .Site.Title }}</title>

    <link rel="stylesheet" href="{{ relURL "css/styles2.css" }}"?v={{ now.Unix
}}">

</head>

```

- partials/header.html

```

<header>

<h1>{{ .Site.Title }}</h1>

    <nav class="pill-nav">

        <a href="{{ relURL "" }}">Página Inicial</a>

        <a href="{{ relURL "about" }}">Sobre Mim</a>

        <a href="{{ relURL "experience" }}">Experiência</a>

        <a href="{{ relURL "schooling" }}">Formação</a>

    </nav>

</header>

```

- partials/footer.html

```

<footer>

    <div class="contact-info">

        <span class="primary-email">Email USP:<a
href="mailto:gurza.mateus@usp.br"> gurza.mateus@usp.br</a></span>

        <span class="secondary-email">Email secundário:<a
href="mailto:mateusgurza@gmail.com"> mateusgurza@gmail.com</a></span>

    </div>

    <div class="copyright">

        <p>© 2024 por Mateus Matzkin Gurza. Última atualização em 13/09/2024</p>

    </div>

</footer>

```

- experience/list.html

```
{{ define "main" }}

<h2 class="list-title">EXPERIÊNCIAS</h2>

<div class="list-wrapper">
  {{ range .Pages }}
    <div class="link-wrapper">
      
      <div class="link-text-wrapper">
        <a href="{{ .RelPermalink }}">{{ .Title }}</a>
        <div class="Description">
          {{.Content}}
        </div>
      </div>
    </div>
  {{ end }}
</div>

{{ end }}
```

- experience/single_IC.html

```
{{ define "main" }}

  <div class="exp-main">

    <div class="exp-box">
      <div class="exp-img">
        
      </div>
      <article class="exp-text">
        {{ with .GetPage "textfiles/IC_exp" }}
          {{ .Content }}
        {{ end }}
      </article>
    </div>

  </div>

{{ end }}
```

- experience/single_polijunior.html

```
{{define "main" }}
  <div class="exp-main">

    <div class="exp-box">
      <div class="exp-img">
        
      </div>
      <article class="exp-text">
        {{ with .GetPage "textfiles/pj_treinamentos" }}
          {{ .Content }}
        {{ end }}
      </article>
      <a href="https://polijunior.com.br/"><button>Site da Poli
Júnior</button></a>
    </div>

    <div class="exp-box">
      <div class="exp-img">
        
      </div>
      <article class="exp-text">
        {{ with .GetPage "textfiles/pj_case" }}
          {{ .Content }}
        {{ end }}
      </article>
    </div>

  </div>

{{ end }}
```



```

        <a href="https://polijunior.com.br/cases/zoetis/"><button>Case
Zoetis no site</button></a>
    </article>
</div>

    <div class="exp-box">
        <div class="exp-img">
            
        </div>
        <article class="exp-text">
            {{ with .GetPage "textfiles/pj_analista" }}
                {{ .Content }}
            {{ end }}
        </article>
    </div>

    <div class="exp-box">
        <div class="exp-img">
            
        </div>
        <article class="exp-text">
            {{ with .GetPage "textfiles/pj_gerente" }}
                {{ .Content }}
            {{ end }}
        </article>
    </div>

    <div class="exp-box">
        <div class="exp-img">
            
        </div>
        <article class="exp-text">
            {{ with .GetPage "textfiles/pj_coordenador" }}
                {{ .Content }}
            {{ end }}
        </article>
    </div>

    <div class="exp-box">
        <div class="exp-img">
            
        </div>
        <article class="exp-text">
            {{ with .GetPage "textfiles/pj_hunter" }}
                {{ .Content }}
            {{ end }}
        </article>
    </div>

</div>
{{ end }}

```

- schooling/list.html

```

{{ define "main" }}

    <h2 class="list-title">FORMAÇÃO ACADÊMICA</h2>

    <div class="list-wrapper">
        {{ range .Pages }}
            <div class="link-wrapper">
                
                <div class="link-text-wrapper">
                    <a href="{{ .RelPermalink }}">{{ .Title }}</a>
                    <div class="Description">

```

```

        {{.Content}}
    </div>
</div>
</div>
    {{ end }}
</div>
{{ end }}

```

- schooling/single_band.html

```

{{ define "main" }}

    <div class="exp-main">

        <div class="exp-box">
            <div class="exp-img">
                
            </div>
            <article class="exp-text">
                {{ with .GetPage "textfiles/band_ciencias" }}
                    {{ .Content }}
                {{ end }}
            </article>
        </div>

        <div class="exp-box">
            <div class="exp-img">
                
            </div>
            <article class="exp-text">
                {{ with .GetPage "textfiles/band_opt" }}
                    {{ .Content }}
                {{ end }}
            </article>
        </div>

        <a href="http://www.oba.org.br/site/index.php?p=conteudo&idcat=22&pag=conteudo&acao=mostra&idaluno=848866&olimp=oba&ed=2019">
            <button>Certificação de Medalha (OBA)</button></a>
    </div>
{{ end }}

```

- schooling/single_btc.html

```

{{ define "main" }}

    <div class="exp-main">

        <div class="exp-box">
            <div class="exp-img">
                
            </div>
            <article class="exp-text">
                {{ with .GetPage "textfiles/btc_curso" }}
                    {{ .Content }}
                {{ end }}
            </article>
        </div>

    </div>

{{ end }}

```

- schooling/single_poli.html

```

{{ define "main" }}

    <div class="exp-main">

```

```

        <div class="exp-box">
            <div class="exp-img">
                
            </div>
            <article class="exp-text">
                {{ with .GetPage "textfiles/poli_curso" }}
                    {{ .Content }}
                {{ end }}
            </article>
            <a href="http://www.pmr.poli.usp.br/" class="article-logo-link"></a>
        </div>

        <div class="exp-box">
            <div class="exp-img">
                
            </div>
            <article class="exp-text">
                {{ with .GetPage "textfiles/poli_optativas" }}
                    {{ .Content }}
                {{ end }}
            </article>
        </div>

    {{ end }}

```

- index.html

```

{{ define "main" }}
    <div class="home-main">
        <div id="index-center-image">
            
        </div>
        <div class="home-main-subdivision">
            <div class="image-side-text">
                {{.Content}}
            </div>
            <div class="button-grouping-row">
                <a href="{{relURL "documents/CV_ptbr_MateusGurza.pdf"
}}"><button>Currículo pt-BR</button></a>
                <a href="{{relURL "documents/CV_enus_MateusGurza.pdf"
}}"><button>Currículo en-US</button></a>
            </div>
            <div id="index-center-nav">
                <nav>
                    <a href="https://github.com/MatzkinGurza"></a>
                    <a href="https://www.linkedin.com/in/mateus-matzkin-gurza-a54115248"></a>
                    <a href="https://www.instagram.com/mateusgurza/"></a>
                </nav>
            </div>
        </div>
    </div>
{{ end }}

```

4.3 CSS

Abaixo segue o arquivo de CSS para ajuste da estética e disposição de elementos na página:

- **css/styles2.css**

```
@import
url('https://fonts.googleapis.com/css2?family=Oswald:wght@200..700&family=Roboto+Co
ndensed:ital,wght@0,100..900;1,100..900&family=Rubik:ital,wght@0,300..900;1,300..90
0&display=swap');

/* General Typography */
body, button {
    font-family: 'Roboto Condensed', 'sans-serif';
}

h1, h2, h3, h4, h5 {
    font-family: "Oswald";
    font-optical-sizing: auto;
    font-weight: 400;
    font-style: normal;
}

.list-title{
    color:white;
}

header a, footer, div.home-main-subdivision {
    font-family: "Oswald","Roboto Condensed", sans-serif;
    font-optical-sizing: auto;
    font-weight: 200;
    font-style: normal;
}

/* Header Design */
header {
    display: flex;
    box-sizing: border-box;
    justify-content:space-between;
    align-items: center;
    width: 100%;
    padding: 15px 5%;
    background-color: rgba(0, 0, 66, 0.604); /* Dark transparent navy */
    color: white;
    box-shadow: 0 0 20px rgba(0, 0, 66, 0.7); /* Softer shadow */
}

header a {
    color: white;
    text-align: center;
    text-decoration: none;
    font-size: 16px;
    padding:10px;
    margin: 10px;
    transition: transform 1.0s ease;
}

header a:hover {
    background-color: #536DFE; /* Bright techy hover color */
    padding: 10px;
    border-radius: 5px;
    transform: scale(1.1);
}

/* Buttons */
button {
    background-color: #397AF4; /* Techy blue */
    color: white;
    padding: 10px 20px;
```

```

border: none;
border-radius: 5px;
box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
transition: background-color 0.3s ease, transform 0.2s ease;
font-weight: 500;
cursor: pointer;
}

button:hover {
background-color: #2a62d4; /* Darker hover for buttons */
transform: translateY(-2px); /* Slight lift on hover */
}

/* Layout and Structure */
main {
width: 100%;
box-sizing: border-box;
margin: 0 auto;
padding: 20px 5%;
flex: 1;
/*background: linear-gradient(135deg, #397AF4 0%, #0E1C26 100%); *//*Gradient
background */
}

/* Main Section Flexbox */
div.home-main {
display: flex;
justify-content: space-between;
align-items: center;
max-width: 1200px;
margin: 40px auto;
background-color: inherit;
}

/* Image Container */
#index-center-image {
flex: 1;
display: flex;
justify-content: center;
margin-right: 40px;
}

#index-center-image img {
width: 100%;
max-width: 300px;
border-radius: 8%;
box-shadow: 0 0 30px rgba(0, 0, 0, 0.6); /* Blue shadow for techy vibe */
}

/* Content Container */
div.home-main-subdivision {
flex: 2;
background-color: rgba(0, 0, 66, 0.7); /* Transparent dark background */
color: white;
padding: 40px;
border-radius: 8px;
box-shadow: 0 4px 8px rgba(0, 0, 0, 0.429);
}

/* Social and Navigation Links */
#button-grouping-row {
display: flex;
justify-content: flex-start;
margin-top: 20px;
margin-right: 20px;
gap: 20px
}

#button-grouping-row button{
margin: 0 10px; /* Adds 10px space between each icon */
}

```

```

#index-center-nav {
  margin-top: 30px;
  display: flex;
  justify-content: flex-start;
  gap: 20px;
}

#index-center-nav img {
  width: 40px; /* Slightly larger icons for a modern feel */
  height: 40px;
  transition: transform 0.3s ease;
}

#index-center-nav img:hover {
  transform: scale(1.2); /* Hover effect on icons */
}

#index-center-nav a {
  margin: 0 10px; /* Adds 10px space between each icon */
}

/* Footer Design */
footer {
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  padding: 20px;
  background-color: rgb(0, 0, 66); /* Darker footer */
  color: #CCCCCC; /* Softer text */
}

footer .contact-info {
  display: flex;
  justify-content: center;
  gap: 20px;
}

footer .copyright {
  margin-top: 10px;
  display: flex;
  justify-content: center;
  align-items: center;
  font-size: 0.85rem; /* Optional styling for text size */
}

/* General Body Styling */
body {
  min-height: 100vh;
  margin: 0;
  display: flex;
  flex-direction: column;
  /*background-color: #0E1C26; *//* Dark background for cleaner look */
  background: linear-gradient(45deg, #01060f 0%, #26a5ff 100%);
}

/* List pages */
.link-wrapper {
  flex-wrap: nowrap; /* Allow content to wrap to avoid overflow */
  flex-direction: row;
  background-color: #4B89F4;
  color: white;
  padding: 10px; /* Adjusted padding for a tighter, more consistent look */
  border-radius: 8px;
  box-shadow: 0 6px 12px rgba(0, 0, 0, 0.15);
  transition: background-color 0.3s ease, transform 0.2s ease;
  font-weight: 600;
  cursor: pointer;
  display: flex;
  align-items: center;
  gap: 15px; /* Slightly smaller gap for better visual consistency */
}

```

```

max-width: 1200px;
min-width: 1000px;
width: 100%;
flex-shrink: 0;
position: relative;
}

/* Image styling */
.link-wrapper img.list-side-img {
  width: 130px; /* Set a fixed width for the image */
  height: 130px; /* Set a fixed height for the image */
  object-fit: contain; /* Ensure the image fits within the given dimensions */
  border-radius: 5px; /* Keep the image rectangular with slightly rounded corners */
}

/* Add some space between the image and the text */
margin-right: 20px;
margin-left: 20px;
}

/* Wrapper for the text (link + description) */
.link-text-wrapper {
  display: flex;
  flex-direction: column; /* Stack the link and description vertically */
  justify-content: flex-start; /* Align text items to the top */
  flex-grow: 1; /* Make the text take the remaining space */
}

/* Link styling */
.link-wrapper a {
  color: white;
  font-size: 1.2em; /* Increased font size for links */
  text-decoration: none;
  margin-bottom: 5px; /* Space between link and description */
}

/* Hover effects */
.link-wrapper:hover {
  background-color: #3467D6; /* Darker blue on hover */
  transform: translateY(-5px); /* Slight lift effect on hover */
}

.link-wrapper a:hover {
  text-decoration: underline; /* Underline the link on hover for clarity */
}

/* List wrapper layout */
.list-wrapper {
  display: flex;
  flex-direction: column;
  gap: 20px; /* Gap between items */
  justify-content: flex-start;
  align-items: flex-start;
  max-width: 600px; /* Limit width for better readability */
}

/* Description styling */
.Description {
  margin-top: 5px; /* Less margin for tighter spacing */
  color: rgba(255, 255, 255, 0.85); /* Slightly lighter description text */
  font-size: 1em;
  font-style: italic;
}

/*single... sections of the site */
.exp-main {
  display: flex;
  flex-wrap: wrap; /* Permite que os blocos se reorganizem quando não couberem na
mesma linha */
  justify-content: space-between;
  gap: 20px; /* Espaçamento entre os blocos */
}

```

```

.exp-box {
  background-color: #002855; /* Cor azul escura como fundo */
  border-radius: 10px; /* Bordas arredondadas */
  padding: 20px;
  flex: 1 1 calc(33.333% - 40px); /* Faz os blocos ocuparem 1/3 da largura,
ajustando conforme necessário */
  min-width: 300px; /* Largura mínima antes de quebrar para a próxima linha */
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2); /* Sombra leve para efeito de
elevação */
  color: white; /* Texto branco */
  position: relative;
}

.exp-box:hover {
  transform: translateY(-5px); /* Elevação ao passar o mouse */
  transition: 0.3s; /* Transição suave */
}

.exp-img {
  display: flex;
  justify-content: center;
  align-items: center;
  background-color: #004d9900; /* Azul mais claro para o fundo da imagem */
  width: 100%; /* Tamanho fixo da imagem */
  height: 200px;
  margin: 0 auto 20px auto; /* Centraliza no topo do bloco */
}

.article-img {
  height: inherit;
  border-radius: 5%;
}

.article-logo-link{
  height: 60px;
  position: absolute; /* A imagem será posicionada de forma absoluta */
  bottom: 20; /* Alinha no topo */
  right: 20; /* Alinha à direita */
}

.article-logo{
  width: 100%;
  height: 100%;
}

.exp-text {
  text-align: left; /* Texto alinhado à esquerda */
}

/* Responsividade para telas menores */
@media (max-width: 768px) {
  .exp-box {
    flex: 1 1 calc(50% - 40px); /* Ajusta para 2 blocos por linha em telas
médias */
  }
}

@media (max-width: 480px) {
  .exp-box {
    flex: 1 1 100%; /* Um bloco por linha em telas pequenas */
  }
}

```


4.4 O Site

Abaixo seguem imagens dos principais tipos de página no site:

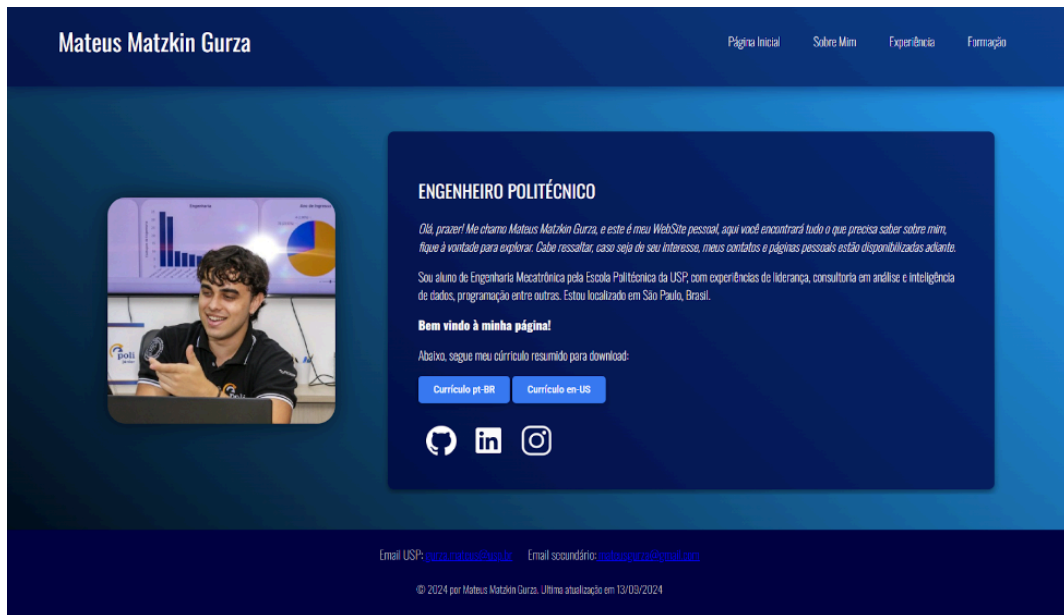


Figura 1: Home Page - Layout index.html

Observações:

- Clicando nos botões de currículo você baixa o documento em PDF na língua especificada.
- Se clicadas, as imagens das redes sociais direcionam o usuário para minhas páginas pessoais.



Figura 2: Lista de Conteúdos em “Experiência” - Layout list.html em layouts/experience. Similar à página de lista em “Formação”

Observações:

- Clicando no título das seções você é direcionado para páginas de aprofundamento acerca de cada temática. Quando o título estiver disponível para ser clicado ele será sublinhado em branco.

- Esta página de lista utiliza de arquivos em content/experience onde estão as descrições rápidas de cada atividade. Nessas páginas em markdown os front matters foram utilizados para apontar metadados que fazem referência às imagens na própria página de lista e também aos layouts para os quais cada item da lista redireciona.

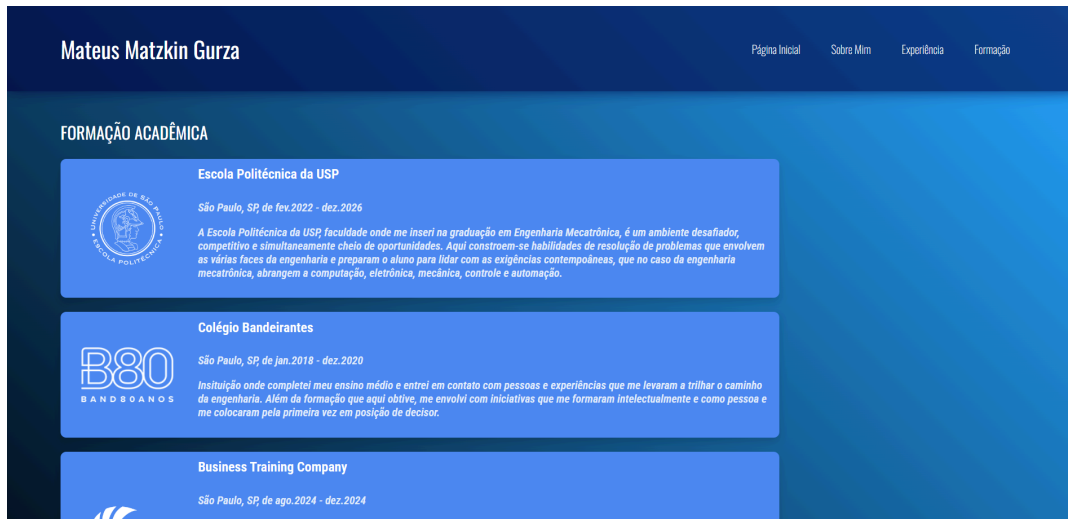


Figura 3: Lista de Conteúdos em “Formação” - Layout list.html em layouts/experience.



Figura 4: Página “Poli Júnior” na Lista da figura 2 - Layout single_polijunior.html. Similar às outras páginas que podem ser acessadas por meio das listas de conteúdo.

Observações:

- As páginas de conteúdo contêm blocos onde são explicadas diferentes vivências e aspectos do tema selecionado na página de lista. Esses blocos de conteúdo podem conter texto, imagens, links, e botões que redirecionam para páginas relevantes ao tema abordado.
- Esta página possui **media query** tal que ela se organiza funcionalmente para celulares e computadores e outras telas em função do tamanho da mesma. A página utiliza flexbox para dispor os blocos horizontalmente até um limite de largura mínimo, a partir do qual, para telas menores, os blocos passam a se organizar verticalmente. Veja a seguir a disposição vertical:

Mateus Matzkin Gurza
Página Inicial
Sobre Mim
Experiência
Formação



Poli Júnior - Visão Geral

A Poli Júnior, a maior empresa júnior do Brasil, vinculada à Escola Politécnica da Universidade de São Paulo, desempenhou um papel crucial na minha formação e na formação de profissionais altamente capacitados e preparados para enfrentar os desafios do mercado de trabalho. A empresa oferece aos seus membros a oportunidade de participar ativamente da gestão de projetos reais, atuando em diversas áreas, como engenharia civil, tecnologia, consultoria em inteligência de dados e consultoria empresarial. Ao proporcionar essa experiência prática, a Poli Júnior prepara seus membros para transitar com segurança no mercado de trabalho, desenvolvendo tanto habilidades técnicas quanto competências de gestão e liderança.

Além do desenvolvimento prático, a Poli Júnior promove capacitações internas regulares, com treinamentos técnicos e gerenciais realizados por meio de parcerias com empresas de renome, como Bain & Company (especialmente através de sua unidade de consultoria digital e de tech, a Vector da Bain), McKinsey (com seu branch de AI, QuantumBlack), EY e Deloitte. Esses treinamentos permitem que os membros tenham contato direto com ferramentas e metodologias usadas por líderes de indústria, potencializando o aprendizado prático e estimulando os participantes das demandas reais do mercado.

A Poli Júnior também faz parte do Movimento Empresas Júnior (MEJ), uma rede nacional que promove a integração e o desenvolvimento de empresas júniores em todo o Brasil. Graças por eventos organizados pela Poli Júnior, como congressos e workshops, a Poli Júnior oferece aos seus membros uma visão global das tendências e inovações que estão moldando o futuro das empresas e da engenharia. Esses eventos reforçam o compromisso com a excelência e o aprendizado contínuo, conectando os participantes a uma comunidade vibrante de jovens empreendedores.

Por meio de suas parcerias com grandes empresas e sua integração com o MEJ, a Poli Júnior funciona como um verdadeiro campo de treinamento para futuros líderes, capacitando seus membros a lidar com os desafios de um mercado dinâmico e em constante evolução. A experiência adquirida na Poli Júnior, aliada aos conhecimentos com empresas de ponta e a exposição à gestão real, prepara a mim e prepara seus membros para atuarem com confiança em ambientes corporativos de alto nível.

[Site da Poli Júnior](#)



Caso de Sucesso: Zeetis - Análise de Cesta de Compras e Gestão de Estoque

Durante minha atuação como Analista de Soluções de Dados na Poli Júnior, conduzi um projeto estratégico para a Zeetis, líder na fabricação e distribuição de medicamentos para veterinários, analisando os dados e o comportamento de compra dos clientes. O objetivo era fornecer uma compreensão mais aprofundada sobre o comportamento de compra dos clientes e os efeitos de eventos de estoque, ajudando a empresa a otimizar suas campanhas de marketing e estratégias de gestão de estoque.

PROBLEMA INICIAL: A Zeetis enfrentava dificuldades para entender como os clientes compravam diferentes categorias de medicamentos em conjunto. Além disso, havia frustração da empresa sobre os impactos de eventos promocionais, relacionados ao estoque de medicamentos, que resultavam em uma redução no retorno financeiro, em detrimento da falta de clareza sobre a necessidade da empresa de entender suas estratégias de vendas e estoque de forma eficiente.

SOLUÇÃO: Foi desenvolvida uma Análise de Cesta de Compras (Market Basket Analysis - MBA), por meio de algoritmos de aprendizado de máquina como o Apriori, e utilizaram-se indicadores-chave de desempenho de negócios, como Support, Confidence e Lift, para identificar padrões de compra entre os produtos. Além disso, elaboramos um relatório comparando as Análises Exploratórias de Dados (EDA), que inclui uma análise de séries temporais, permitindo a projeção futura de demanda para investimentos mais precisos.

Essa solução forneceu uma visão abrangente sobre o comportamento de compra dos clientes, além de validar as hipóteses da Zeetis sobre os efeitos de eventos de estoque. Com essas informações, a empresa pôde ajustar suas campanhas de marketing, de maneira mais assertiva e rentável, e otimizar o nível de estoque, garantindo maior precisão na gestão de estoque e no planejamento estratégico.

Figura 5: Página com media query aplicada