

Front End

George Hamilton Hamann



Apresentações

Vamos nos conhecer



Seu nome:

Idade:

Qual atividade faz hoje:

Qual a motivação para fazer o curso

Possui experiência com desenvolvimento

O que espera aprender no final do curso

O que veremos



- Conceitos básicos de desenvolvimento web
- HTML
- CSS
- Javascript
- Typescript
- Angular Basic
- Angular Project
- Security

Recapitulando os projetos

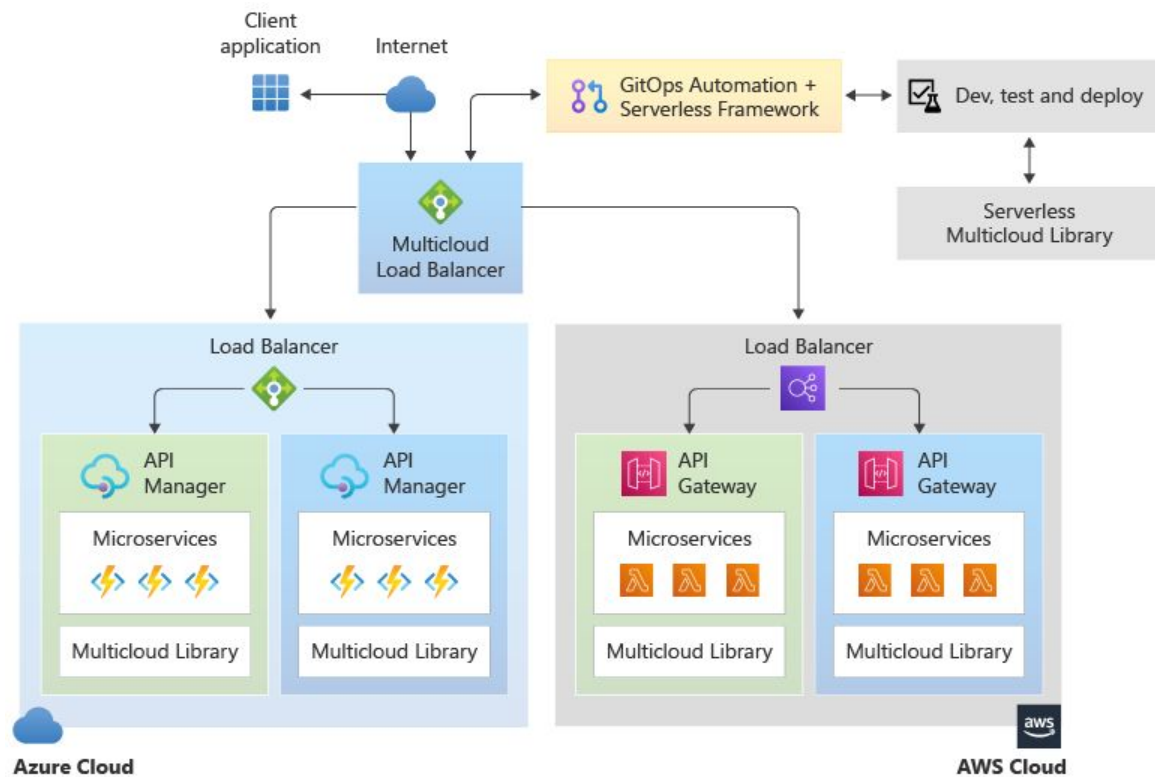
Gostaria se possível que fosse apresentado a criação dos projetos que fizeram até o momento.

Desenvolvimento Web



Quando pensamos em desenvolvimento web, precisamos conhecer o ciclo de vida das suas aplicações. O mais conhecido quando pensamos em Web é o conceito “cliente-servidor”.

Com advento de diversos provedores de soluções nuvem, também temos agora o conceito “serverless”



Exemplo de arquitetura serverless

Front-end vs Back-end



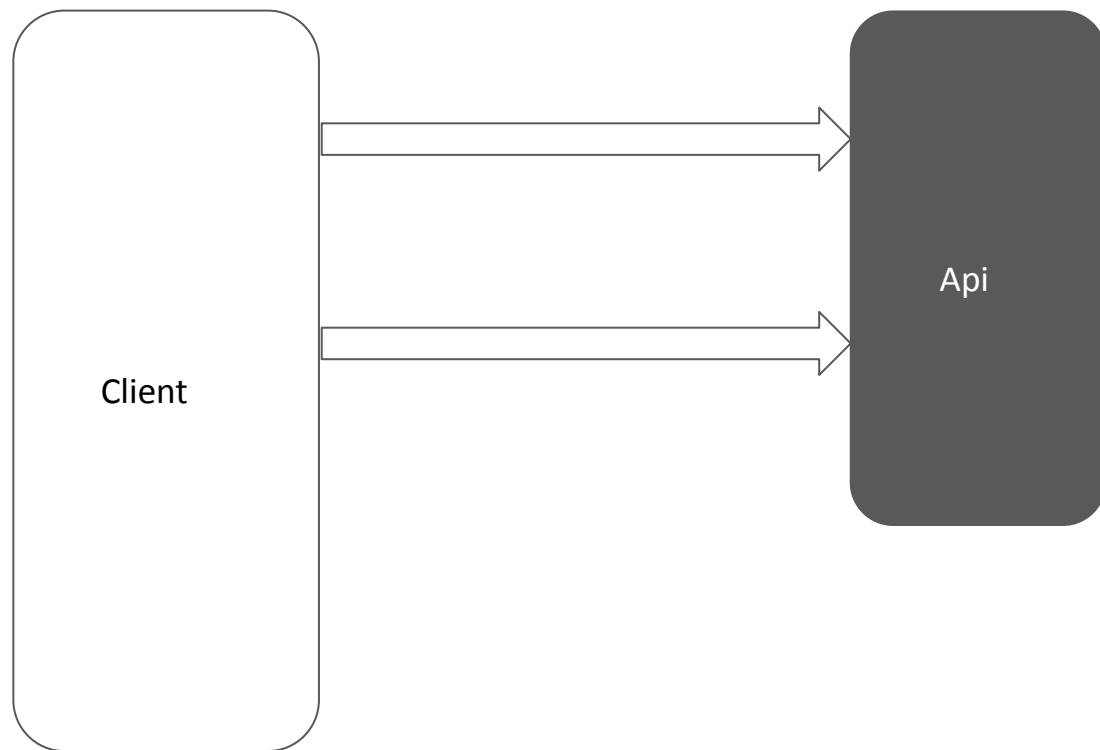
Front-end é a camada de apresentação que geralmente se tem interação com alguém ou algo, para se coletar entradas.

Back-end é a camada de serviço ou processamento, que recebe os dados ou que provêem informação para a camada de apresentação.

Front-end vs Back-end



Existem aplicações, que são desenvolvidas todas elas numa camada única, de back e front, altamente acoplado ou interdependente uma da outra. Porém existe uma abordagem onde as camadas atuam de formas independente ou desacoplada, pois hoje as aplicações que rodam no front-end possui recurso suficiente.



Exemplo de abordagem desacoplada

Ferramenta de controle de versão e gerenciamento de arquivo.

- Repositório, é um diretório que armazena o conteúdo a ser gerenciado.
- Branch, é uma ramificação que contém o estado de uma versão do diretório
- Principais comandos, git clone, git push, git pull , git init, git add, git commit
- GitFlow, master, develop, features

GIT- Exercício 01



Crie uma conta no github para que possa gerenciar os exercícios e seu códigos.

Crie um projeto no git hub, “Aprendendo Front End”

- Realize o clone
- Crie um “Readme.md”
- Crie um git ignote
- Realize um git Add
- Realize um git push
- Comece a seguir alguém no Git Hub

Html



Hyper text model language, mais conhecido como HTML ou como diz o nome, linguagem de marcação de Hypertexto, é código que se usa para estruturar uma página web, baseada em um série de elementos, onde contem as regras para que os navegadores possam interpretar e apresentar os dados na tela.

Linha do tempo, [HTML Standard \(whatwg.org\)](http://whatwg.org)

Html

```
<html>  
  <head>  
    <meta charset="utf-8">  
    <title>Título da página</title>  
  </head>  
    <body>  
      <p> Olá mundo HTML <p>  
    </body>  
</html>
```

Html - Exercício 02



Crie um HTML, onde
contenha:

- Nome
- Idade
- Nacionalidade
- Cidade

```
<html>
  <head>
    <meta charset="utf-8">
    <title>Título da
página</title>
  </head>
  <body>
    <p> Olá mundo
HTML <p>
  </body>
</html>
```

HTML



Vamos ver alguns dos elementos que o HTML

- Listas
- Tabelas
- Seções de páginas
- Formulários
- Caracteres especiais
- Multimédia

HTML - Listas



- Existem dois tipos, uma ordenado e outra desordenada
- A tag para uma lista desordenada é
- Uma lista ordenada é marcada por número
- A tag para uma lista ordenada é
- Os itens da lista são representados por
- Um item pode conter outros elementos
- Listas podem estar contidas dentro de outras listas

HTML - Listas ex

```
<html>
  <body>
    <ul>
      <li>Lages</li>
      <li>Londres</li>
    </ul>
    <ol>
      <li> George </li>
      <li> Hamilton</li>
    </ol>
  </body>
</html>
```

HTML - Tabelas



- Tag `<table></table>`
- é um conjunto de linhas e colunas
- linha `<tr></tr>` `<th colspan="3">Users</th>`
 - `<td rowspan="2">Free Time</td>`
- coluna `<td></td>`
- cabeçalho `<th colspan="3">`

HTML - Seções

- header, topo da página
- nav, navegação
- article e section
 - segregação de elementos
- aside, algo não ligado ao conteúdo principal
- footer, fim da página
- div, agrupamento genérico



HTML - Forms

- `<form> </form>`
- Iteração para controlar entrada de dados, através de elementos de texto, seleção, botões...
- `<input type="text" id="name">`
 - email, tel, text, number
- `label < label for="name">Nome</label>`
 - São clicáveis
- Exemplo de texto, placeholder
- Área de texto, `<textarea>`
- Drop down, `<select> <option>oi</option>`
- Botões, `<button>Confirmar</button>`
- `<input type="button" value="Send">`

HTML - Forms



```
<form>
  <h2>Contact Us</h2>
  <label for="type">Type: </label>
  <select id="type">
    <option>General</option>
    <option>Payment Issue</option>
    <option>Other</option>
  </select><br><br>
  <label for="name">Name: </label>
  <input type="text" id="name"><br><br>
  <label for="message">Message: </label><br>
  <textarea></textarea><br>
  <input type="button" value="Send">
</form>
```

HTML - Caracteres

Literal character	Character reference equivalent
<	<
>	>
"	"
'	'
&	&

HTML - Multimedia

- `<audio src="audio.mp3" controls loop>`
Audio not supported
`</audio>`
- `<audio controls>`
- `<source src="audio.mp3" type="audio/mpeg">`
- `<source src="audio.ogg" type="audio/ogg">`
- `</audio>`
- `<video controls>`
- `<source src="video.mp4" type="video/mp4">`
- `<source src="video.ogg" type="video/ogg">`
- Video is not supported by your browser
- `</video>`

Html - Exercício 03



Crie um HTML, onde contenha exemplo do que vimos até agora:

- Listas
- Tabelas
- Seções de páginas
- Formulários
- Caracteres especiais
- Multimédia

Html - Exercício 04



Crie um HTML, para termos uma lista de tarefas:

- To do
- Doing
- Done
- O card das tarefas deve ter
 - Data
 - Descrição
 - Status (Feito)