



TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS
DESENVOLVIMENTO PARA DISPOSITIVOS MÓVEIS

ENRICO DA LAQUA DOS REIS

PRÁTICA DO GIT E GITHUB + PROJETO HTML/CSS

VITÓRIA
2021

ENRICO DA LAQUA DOS REIS

PRÁTICA DO GIT E GITHUB + PROJETO HTML/CSS

Atividade avaliativa apresentada à disciplina do Projeto Integrador IV-A do Centro Universitário Faesa, sob tutoria da Professora Lorena Piza Arndt.

VITÓRIA
2021

SUMÁRIO

1 INTRODUÇÃO.....	2
2 DIFERENÇA ENTRE GIT E GITHUB.....	3
3 PRINTS DO REPOSITÓRIO.....	7
4 PROJETO.....	16

1 INTRODUÇÃO

É de extremamente importante sabermos nos organizar e planejar nossas rotinas diárias, na programação não é diferente, antes de começarmos a aprender a programar devemos nos importar como gerenciar nossos arquivos, para isso é recomendável criarmos um ambiente de estudos para guardar nossos códigos de práticas e exemplos do que estamos aprendendo. Atualmente contamos com ferramentas que possibilitam a utilização de recursos que facilitam a vida dos programadores. Para isso será necessário cumprir alguns requisitos de sistema: instalar o Git em nossa máquina e criar uma conta no GitHub.

Como proposta o presente trabalho tem por objetivo colocar diferenciar o Git do GitHub e colocar em prática a criação de um repositório no GitHub e enviar a proposta do projeto que consiste numa página HTML/CSS com a divulgação de um ponto turístico.

2 DIFERENÇA ENTRE GIT E GITHUB

Git:

Sistema de controle de versionamento de arquivos. Usado principalmente para desenvolvimento de softwares, mas pode ser usado para registrar o histórico de edições de qualquer tipo de arquivo.

Nos permite ter controle das alterações dos arquivos salvas em diferentes versões. Com isso, é possível retomar uma versão anterior (ou quantas necessário for), caso necessitemos corrigir alguma parte do código que não funcionou após alguma alteração ou até mesmo no caso de por algum motivo, não precisar mais dar continuidade àquela alteração.

Com o Git, é possível que diversas pessoas de um grupo de trabalho realizem alterações do mesmo arquivo, incrementando, removendo e alterando partes do código sem causar impacto sobre o outro integrante do grupo de trabalho. O arquivo alterado pode ser salvo de forma local na máquina, e posteriormente pode ser encaminhado para a um repositório remoto (com a validação de um aprovador), de forma que os outros membros do grupo tenham acesso àquela alteração.

Permite também a alteração ou criação de vários arquivos de um mesmo projeto de forma que cada participante deste grupo ao finalizar seu bloco de alterações (separados por branches), poderá testar seu código separadamente, e confirmar o funcionamento de sua parte. Com isso, ele poderá descrever as alterações realizadas em cada arquivo criado/alterado e enviar para o repositório remoto, que conforme informado anteriormente, após a validação do aprovador, este poderá compilar todas as partes do projeto em uma única “branch” (master/main), tendo então o projeto inteiro guardado.

Cito então, algumas ferramentas e funções básicas e essenciais para utilização: Branch: São “*marcos*” nas ramificações contendo a “linha do tempo” de um projeto, onde é possível visualizar o marco do momento em que cada usuário

“salvou” (deu commit) suas alterações. Com ela, é possível acompanhar e reunir várias alterações diferentes feitas simultaneamente no mesmo arquivo (contanto que a alteração de um código não afete a funcionalidade criada em outro) ou em diferentes arquivos do projeto.

Commit: Este é utilizado para *salvar* alterações realizadas em uma branch local. Os arquivos deverão conter uma breve descrição/comentário da alteração realizada separadamente, de forma que facilite a identificação da alteração realizada.

Push: Esta função é utilizada para *publicar* as modificações locais (que sofreram o commit) para um repositório central (remote) de forma a compartilhar com os membros da equipe (caso realizem o “pull”) o código alterado.

Fetch: *Baixa* o conteúdo do repositório remoto selecionado para o local.

Pull: *Busca e baixa* as alterações de códigos do repositório remoto selecionado e atualiza imediatamente o repositório local para que os conteúdos sejam iguais.

O pull/merge, ele já tenta fazer o merge das alterações baixadas direto no seu código (caso não haja conflito).

Merge: *Mescla* as alterações realizadas, e publicadas (commit) de forma que todas as partes do código(arquivos/código/projeto) criadas/alteradas sejam reunidas, tornando-se então, uma nova versão do projeto.

Checkout: Basicamente permite a *navegação entre as branches*. Para iniciar as alterações em uma determinada branch, faz-se necessário primeiramente fazer checkout na mesma.

Remote: Repositório salvo em um local remoto (basicamente um serviço em nuvem) onde ficarão guardadas as alterações do projeto que poderão ser públicas ou não. Como exemplo de repositório remoto, temos o GitHub.

No site do Git, em “GUI Clientes” existem algumas interfaces para melhor visualização e gerenciamento dos arquivos e códigos dos projetos. Porém, esta não é a proposta do documento. Sendo assim, utilizaremos o próprio prompt de

comando do Git para realizar as funções da ferramenta, enquanto o controle de versões faremos pelo próprio GitHub.

Alguns comandos Git:

git init: Cria o diretório “.git” dentro da pasta indicada tornando-a um repositório (master).

git add Readme.md: Envia um arquivo (especificado pelo nome) para a área de "staging" (encenação - uma espécie de espera). Local onde o arquivo fica aguardando o commit.

git add . : Envia todos os arquivos salvos (na área staging) para a área de commit.

git status: informa se existe algum arquivo aguardando commit na branch master. Caso haja, informa em "new file" o arquivo aguardando commit.

git commit "comentário descrevendo as alterações realizadas": realiza o commit do arquivo com o comentário contendo a descrição da alteração entre aspas.

git branch -M "main": altera o nome da branch principal de "master" para "main".

git remote add origin: <https://github.com/EnricoLaqua/BestTrip.git>: Faz a conexão do repositório local com o repositório remoto.

git push -u origin main: solicita o push das alterações locais para a remota. Abre a tela de login para acessar o github e permitir o envio. Envia um e-mail confirmando o push e sobe a mensagem no prompt do Git informando o push.

git push origin main: (sem -u), quando realizar apenas fazer o push pois o repositório já foi criado anteriormente, configurando a origin como upstream no Git.

git checkout -b "nome-da-branch": Cria a branch "nome-da-branch" e já faz checkout nela.

git checkout nome-da-branch: realiza o checkout na branch (alternando entre as branches) para que possa realizar alterações nela e visualizar as alterações que foram feitas. Exemplo: “git checkout master” faz com que eu entre na branch master.

GitHub:

Plataforma para hospedagem de arquivos e código-fonte. Pode ser utilizado para guardar os arquivos com o controle de versionamento usando o Git.

O GitHub trabalha com repositórios (diretórios), usados para armazenar os projetos. Dentro destes repositórios (intitulados normalmente com o nome do projeto), ficam os respectivos arquivos do projeto.

A interface do GitHub permite que o usuário consiga visualizar o conteúdo do arquivo (script do código) dentro da própria página do arquivo apenas selecionando-o.

A ferramenta permite ainda o compartilhamento de arquivos e projetos, de forma a contribuir em projetos privados ou Open Source com qualquer usuário no mundo. Os usuários podem também mesclar repositórios remotos. É muito utilizada também para a divulgação de trabalhos e como portfólio.

Assim como Git, o GitHub também trabalha com branches.

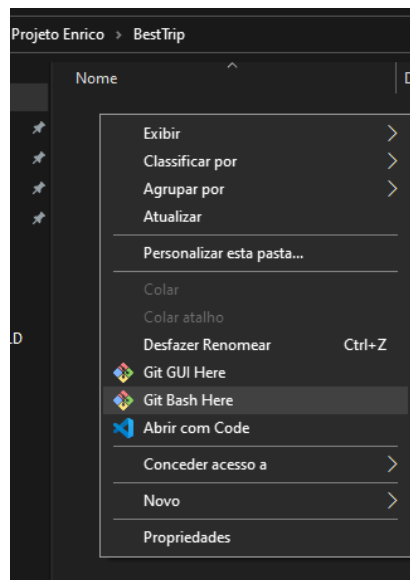
Em suma, ambos os sistemas são facilitadores do dia a dia de muitos desenvolvedores, uma vez que reduz a morosidade e o tempo de entrega dos projetos (visto que vários usuários podem trabalhar simultaneamente), além do controle de versões que reduz muito retrabalho.

3 PRINTS DO REPOSITÓRIO

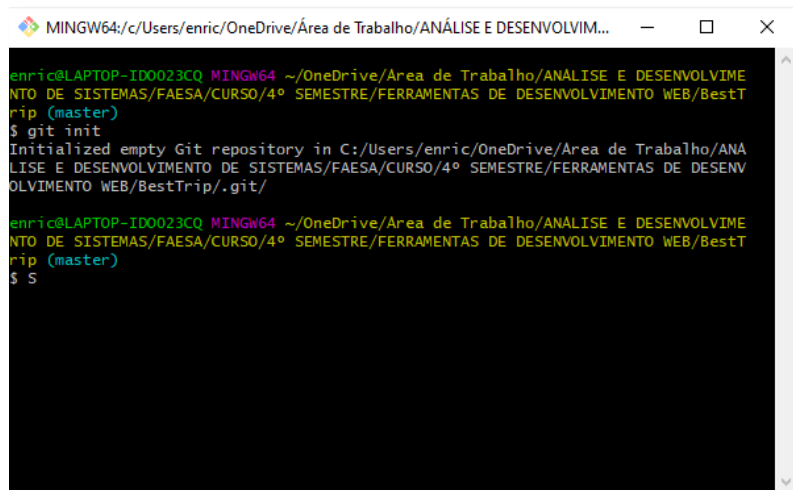
Projeto do Site Best Trip:

Inicialmente, precisaremos criar um diretório local na máquina, onde armazenaremos todos os arquivos do projeto. Nele, será gerado o arquivo “.git”, inicializando um novo repositório Git a partir deste repositório local. No caso do nosso site, utilizaremos o diretório “BestTrip”.

1) Dentro do diretório com o nome do projeto (BestTrip), clique com o botão direito do mouse, e selecione “Git Bash Here”.



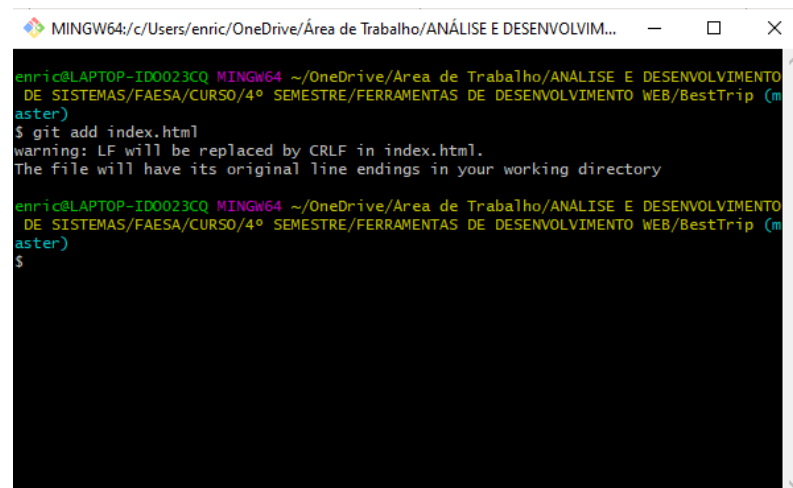
2) Abrirá o prompt de comando do Git, que utilizaremos para realizar as alterações do projeto. O primeiro comando que iremos digitar será o “git init”, que criará o diretório “.git” com todos os arquivos necessários do git, indicando que aquele será o repositório do projeto.



```
MINGW64:/c:/Users/enric/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIM...
enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIM...
NTO DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestT
rip (master)
$ git init
Initialized empty Git repository in C:/Users/enric/OneDrive/Área de Trabalho/ANA
LISE E DESENVOLVIMENTO DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENV
OLVIMENTO WEB/BestTrip/.git/
enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIM...
NTO DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestT
rip (master)
$
```

3) Com o repositório criado, podemos iniciar o desenvolvimento do site, salvando todos os arquivos (".html", ".css", ".jpg"...) dentro desta pasta "BestTrip".

Com os arquivos criados, o próximo passo é voltar ao prompt do Git e digitar o comando "git add index.html" (ou simplesmente "git add ." para adicionar todos os arquivos do projeto de uma só vez). Conforme descrito anteriormente, este comando encaminhará os arquivos do projeto para a área de "staging", onde ficará aguardando o commit.



```
MINGW64:/c:/Users/enric/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIM...
enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIM...
DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestTrip (m
aster)
$ git add index.html
warning: LF will be replaced by CRLF in index.html.
The file will have its original line endings in your working directory
enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIM...
DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestTrip (m
aster)
$
```

4) O comando "git status" mostrará todos os arquivos aguardando commit na área de "staging".

```
MINGW64:/c/Users/enric/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIME...
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   index.html

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        "Diferen\303\247a entre Git e GitHub.docx"
        "atra\303\247\303\265es.html"
        dicas.html
        estilo.css
        img/
        "p\303\241gina_turismo.zip"
        "v\303\255deo.html"

enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIMENTO DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestTrip (master)
$
```

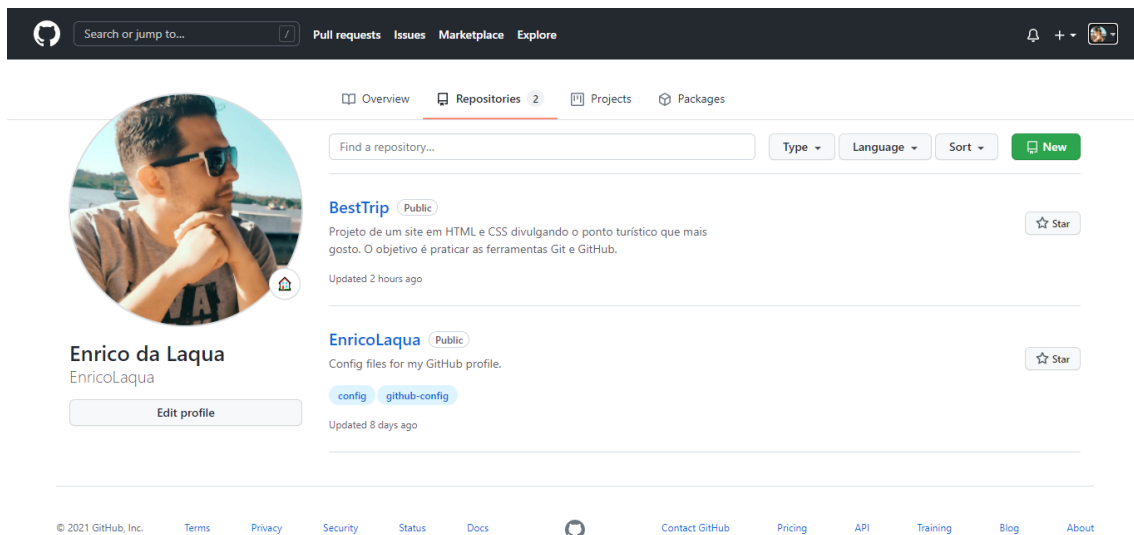
5) No próximo passo, usaremos o comando “git commit -m “primeira versão da home””, para dar commit no arquivo index.html e comentar as alterações que foram feitas no arquivo.

```
MINGW64:/c/Users/enric/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLV...
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        "Diferen\303\247a entre Git e GitHub.docx"
        "atra\303\247\303\265es.html"
        dicas.html
        estilo.css
        img/
        "p\303\241gina_turismo.zip"
        "v\303\255deo.html"

enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIMENTO DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestTrip (master)
$ git commit -m "primeira versão da home"
[master (root-commit) fc699aa] primeira versão da home
1 file changed, 50 insertions(+)
create mode 100644 index.html

enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIMENTO DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestTrip (master)
$
```

6) Agora precisamos criar um repositório no GitHub para que possamos salvar os arquivos de forma remota e fazer o controle de versionamento. Para isso, no site do GitHub, dentro da página principal do seu perfil, clique em “Repositories” e depois em “New”.

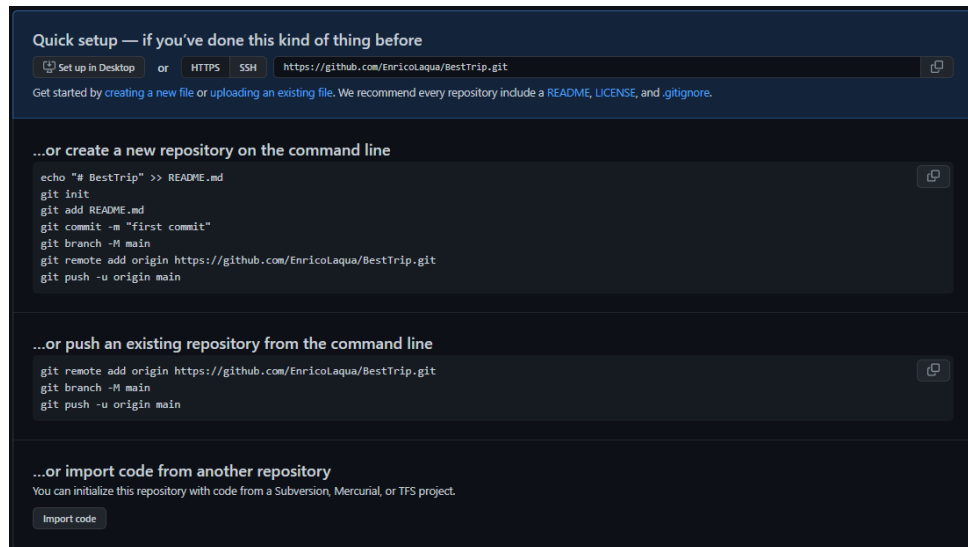


7) Na tela a seguir, preencheremos o “Repository name” com o nome do projeto e em “Description”, uma breve descrição do projeto. Os outros campos manteremos como “default” e por fim, clicar em “Create Repository”.

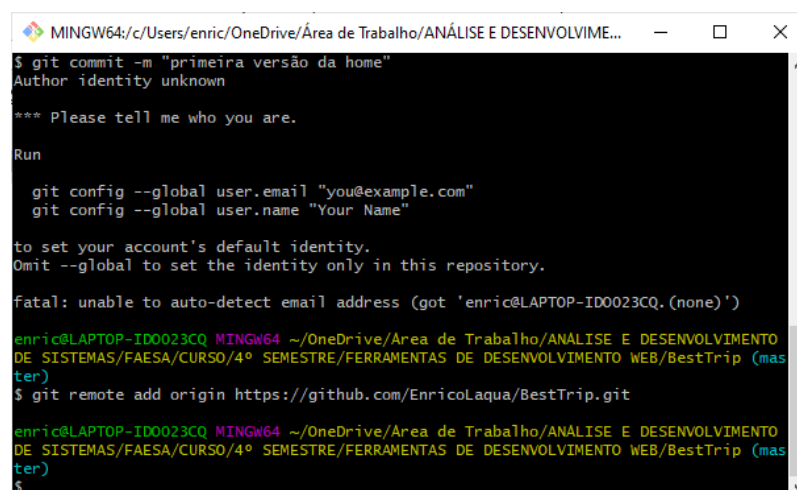
The image shows the 'Create a new repository' form in GitHub. The form has a dark background. At the top, it says 'Create a new repository' and provides a brief explanation of what a repository is. Below this, there are two main sections: 'Owner' and 'Repository name'. The 'Owner' is set to 'EnricoLaqua' and the 'Repository name' is 'BestTrip', which has a green checkmark next to it. Below these fields, there is a suggestion for repository names. The 'Description' field is optional and contains the text 'Projeto de um site em HTML e CSS divulgando o ponto turístico que mais gosto. O objetivo é praticar as ferrai'. Below the description, there are two radio buttons for visibility: 'Public' (selected) and 'Private'. The 'Public' option is described as 'Anyone on the internet can see this repository. You choose who can commit.' and the 'Private' option is described as 'You choose who can see and commit to this repository.' Below these options, there is a section titled 'Initialize this repository with:' which includes three checkboxes: 'Add a README file', 'Add .gitignore', and 'Choose a license'. Each checkbox has a brief description and a 'Learn more' link. At the bottom of the form, there is a green button labeled 'Create repository'.

8) Ao criar o repositório, o próprio GitHub informa a lista de comandos para auxiliar na manipulação dos arquivos entre Git e GitHub. Observe que na parte de cima, existe uma linha com um link <https://github.com/EnricoLaqua/BestTrip.git>.

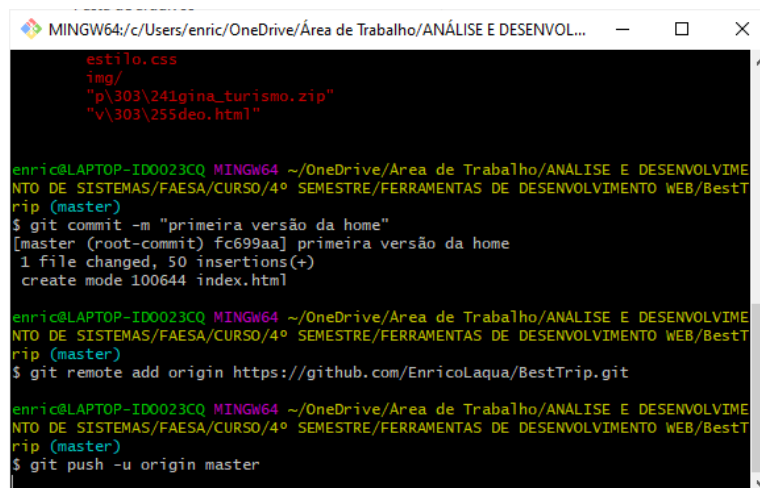
Este é o caminho onde será guardado o repositório remoto no GitHub e com ele faremos a conexão entre o repositório local e remoto.



9) Conforme mencionado acima, o repositório foi criado na máquina local e no GitHub, porém ainda não existe um link de conexão entre eles. Para fazer essa conexão, no prompt de comandos do Git, será inserida a entrada “git remote add origin https://github.com/EnricoLaqua/BestTrip.git” (colar -tecla “insert” ou “ctrl + shift + v”- o link do repositório que foi copiado na tela mencionada acima). Neste comando, estamos fazendo uma conexão do repositório local com o repositório do GitHub através do link e dando o nome do repositório do GitHub de “origin” (origem).



10) Neste próximo passo, faremos um “Push” para enviar o arquivo “index.html” para o GitHub, através do comando “git push -u origin master”. Conforme descrito acima, este comando envia as alterações locais para o repositório remoto (origin), na branch principal (main).



```
MINGW64~/c/Users/enric/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLV...
estilo.css
img/
"p\303\241gina_turismo.zip"
"v\303\255deo.html"

enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIME
NTO DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestT
rip (master)
$ git commit -m "primeira versão da home"
[master (root-commit) fc699aa] primeira versão da home
1 file changed, 50 insertions(+)
create mode 100644 index.html

enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIME
NTO DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestT
rip (master)
$ git remote add origin https://github.com/EnricoLaqua/BestTrip.git

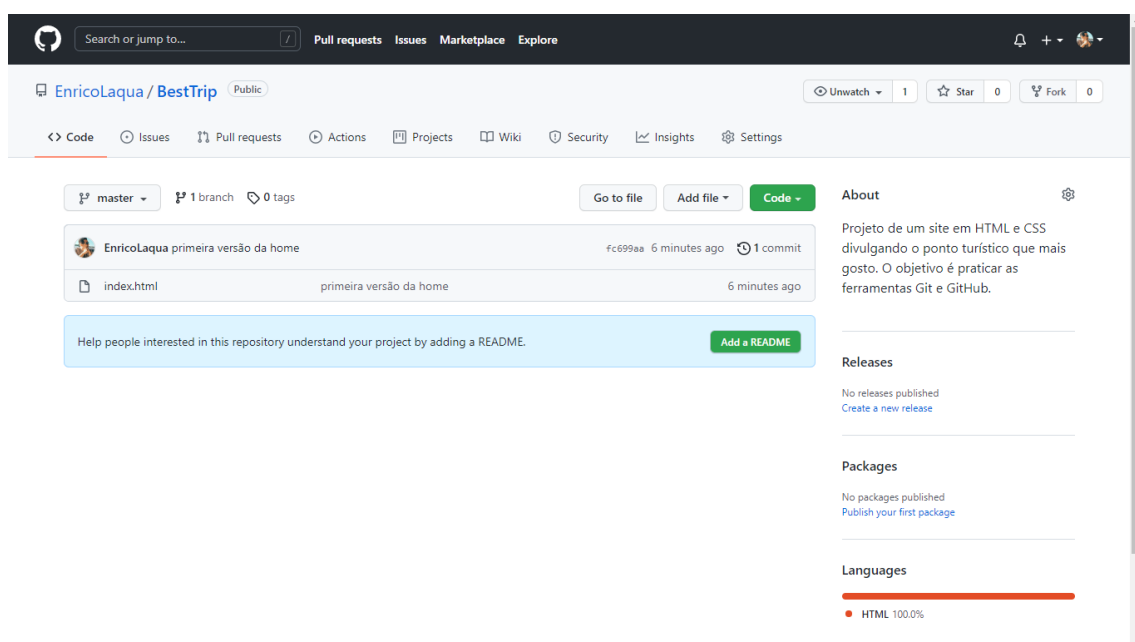
enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIME
NTO DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestT
rip (master)
$ git push -u origin master
```

11) Abrirá uma tela de autenticação para que seja realizado login no GitHub. Clicaremos em “Sign in with your browser” e realizaremos o login. Após este passo, no prompt do Git irá retornar uma mensagem informando o “push” realizado. Ao atualizar a página do GitHub, será possível visualizar o arquivo index.html no repositório.



```
MINGW64/c/Users/enric/OneDrive/Área de Trabalho/ANÁLISE E DESENVOL...
enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIM
NTO DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestT
rip (master)
$ git remote add origin https://github.com/EnricoLaqua/BestTrip.git

enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIM
NTO DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestT
rip (master)
$ git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 1.25 KiB | 1.25 MiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/EnricoLaqua/BestTrip.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```



12) Por fim, enviaremos os demais arquivos do projeto para o GitHub simultaneamente através do comando “git add .”. Conforme já descrito acima, ele envia todos os arquivos para aguardar o commit.

```
MINGW64; c:/Users/enric/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLV...
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 1.25 KiB | 1.25 MiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/EnricoLaqua/BestTrip.git
* [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.

enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIME
NTO DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestT
rip (master)
$ git add .
warning: LF will be replaced by CRLF in atrações.html.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in dicas.html.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in estilo.css.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in video.html.
The file will have its original line endings in your working directory

enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIME
NTO DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestT
rip (master)
$
```

13) Novamente utilizaremos o “git commit -m “Envio dos demais arquivos do site””.

```
MINGW64; c:/Users/enric/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLV...
rip (master)
$ git commit -m "Envio dos demais arquivos do site"
[master 1ee5595] Envio dos demais arquivos do site
15 files changed, 234 insertions(+)
create mode 100644 "Diferen\303\247a entre Git e GitHub.docx"
create mode 100644 "atra\303\247\303\265es.html"
create mode 100644 dicas.html
create mode 100644 estilo.css
create mode 100644 img/cachoeira.png
create mode 100644 img/catedral.png
create mode 100644 img/centrogramado.png
create mode 100644 img/circular.png
create mode 100644 img/gramado.png
create mode 100644 img/jardim.png
create mode 100644 img/lago.png
create mode 100644 img/natal.png
create mode 100644 img/passeio.png
create mode 100644 "p\303\241gina_turismo.zip"
create mode 100644 "v\303\255deo.html"

enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIME
NTO DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestT
rip (master)
$
```

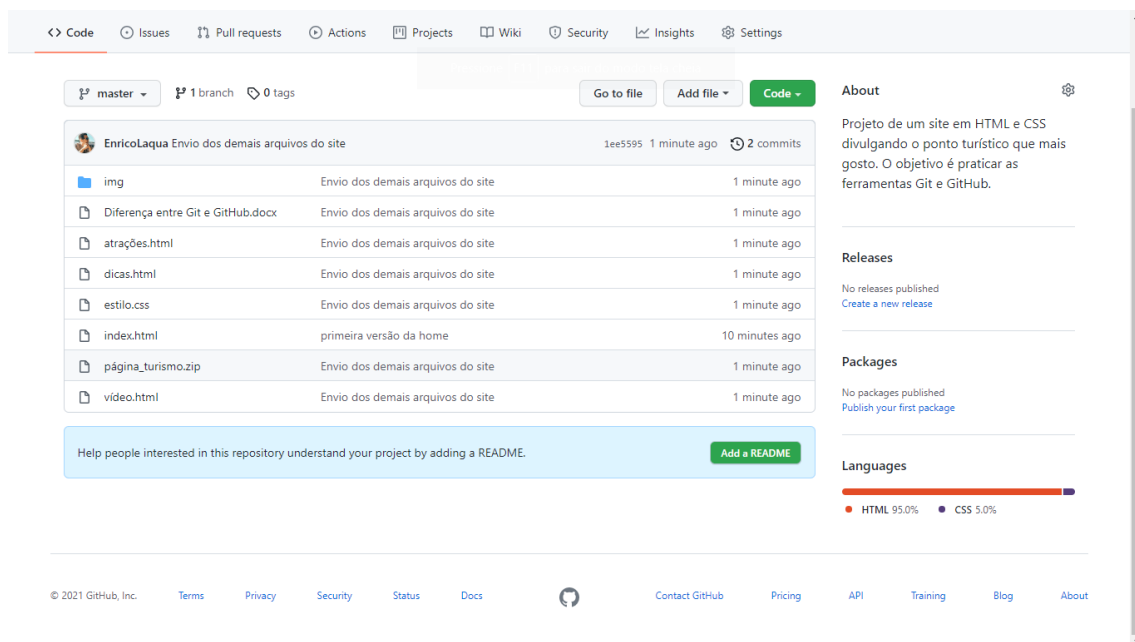
14) E desta vez, para o “push”, utilizaremos o comando “git push origin master”, sem utilizar o “-u” pois a “origin” já foi definida anteriormente (no passo #10) como repositório remoto upstream no Git.


```
MINGW64/c/Users/enric/OneDrive/Área de Trabalho/ANÁLISE E DESENVOL...
create mode 100644 img/lago.png
create mode 100644 img/natal.png
create mode 100644 img/passeio.png
create mode 100644 "p\303\241gina_turismo.zip"
create mode 100644 "v\303\255deo.html"

enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIM
NTO DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestT
rip (master)
$ git push origin master
Enumerating objects: 19, done.
Counting objects: 100% (19/19), done.
Delta compression using up to 8 threads
Compressing objects: 100% (18/18), done.
Writing objects: 100% (18/18), 15.15 MiB | 1.98 MiB/s, done.
Total 18 (delta 9), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (9/9), done.
To https://github.com/EnricoLaqua/BestTrip.git
fc699aa..1ee5595 master -> master

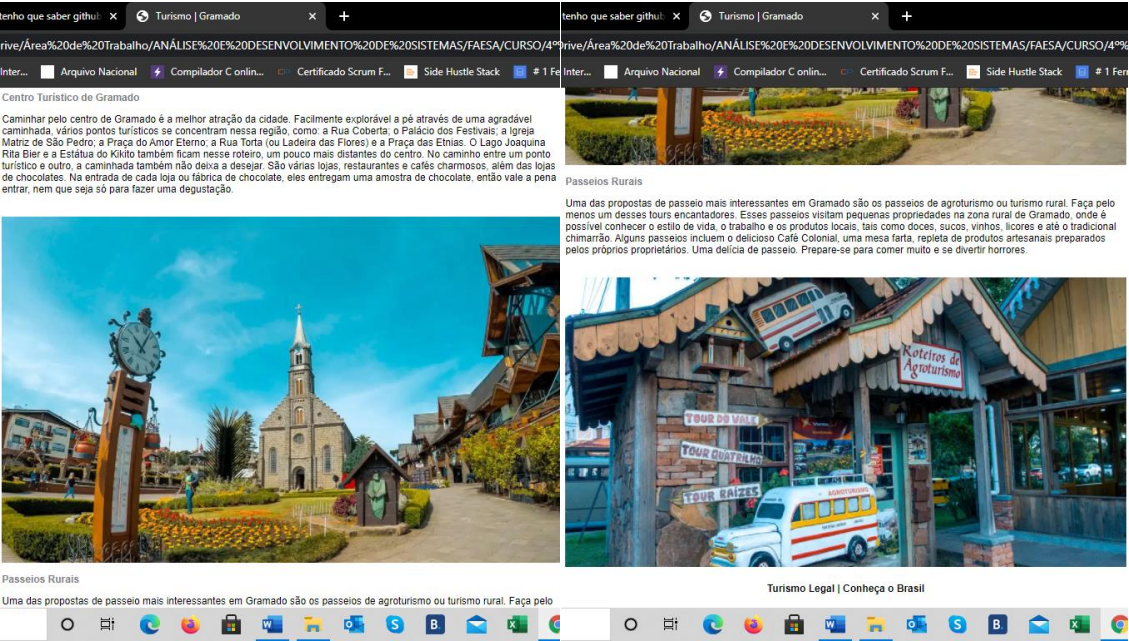
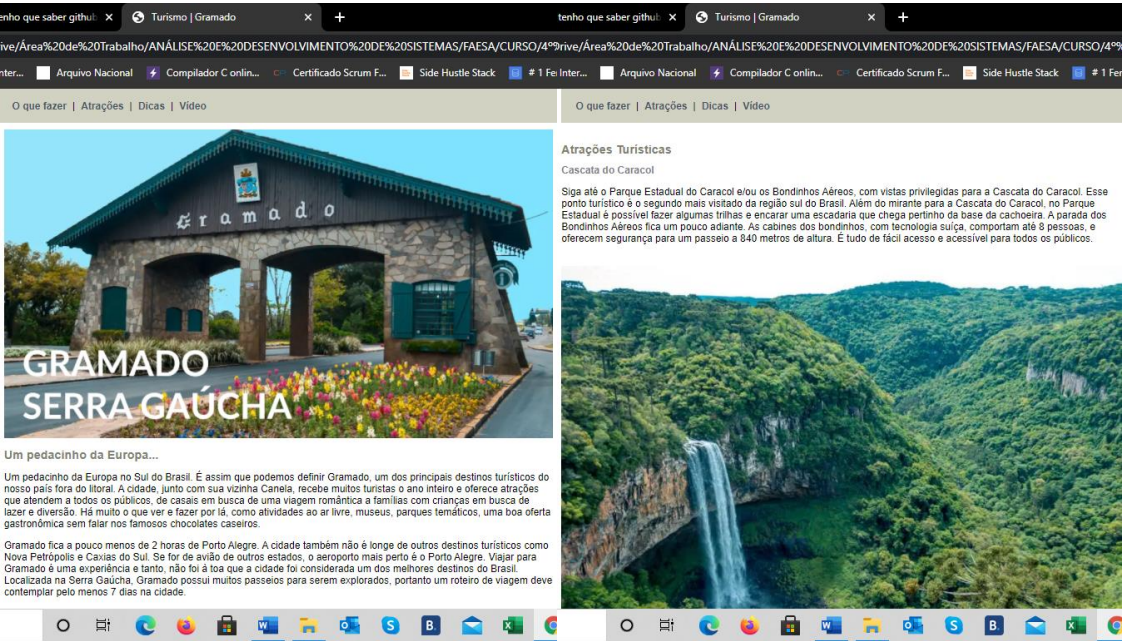
enric@LAPTOP-ID0023CQ MINGW64 ~/OneDrive/Área de Trabalho/ANÁLISE E DESENVOLVIM
NTO DE SISTEMAS/FAESA/CURSO/4º SEMESTRE/FERRAMENTAS DE DESENVOLVIMENTO WEB/BestT
rip (master)
$
```

15) No GitHub será possível visualizar as alterações do index.html e os demais no repositório, com seus respectivos códigos.



4 PROJETO

Telas:



tenho que saber github: x

Turismo | Gramado

x

+


rive/Area%20de%20Trabalho/ANÁLISE%20E%20DESENVOLVIMENTO%20DE%20SISTEMAS/FAESA/CURSO/4%


Inter... Arquivo Nacional Compilador C online... Certificado Scrum F... Side Hustle Stack # 1 Fer

O que fazer | Atrações | Dicas | Vídeo

De olho nas dicas...

Escolher a melhor época para visitar Gramado depende um pouco do seu objetivo de viagem. Se você quer curtir o frio em um destino romântico, deve apostar no inverno. Se quiser fugir do frio, melhor ir no verão. Vale ficar de olho no calendário de eventos da cidade. Em Agosto, por exemplo, acontece o Festival de Cinema de Gramado. Em Setembro o Festival Internacional de Cultura e Gastronomia tem uma programação de oficinas, degustações, shows musicais, peças teatrais e festas que permitem uma imersão cultural e gastronômica. Durante a temporada de inverno, vários eventos se distribuem pela cidade. O tradicional Estação Gramado é formado por atrações como feijoada, espetáculos como a versão pocket do Korvatunturi, saraus, bate-papos com jornalistas e oficinas gratuitas de pães e cucas, entre outros.





Chegando em Gramado

O Aeroporto de Porto Alegre é mais próximo para chegar em Gramado. O tempo de viagem é de cerca de 2h30 a 3h, mas imprevistos podem acontecer e estender ainda mais a viagem. Quem preferir uma opção mais confortável, pode optar por

tenho que saber github: x

Turismo | Gramado

x


+

rive/Area%20de%20Trabalho/ANÁLISE%20E%20DESENVOLVIMENTO%20DE%20SISTEMAS/FAESA/CURSO/4%

Inter... Arquivo Nacional Compilador C online... Certificado Scrum F... Side Hustle Stack # 1 Fer


Chegando em Gramado

O Aeroporto de Porto Alegre é mais próximo para chegar em Gramado. O tempo de viagem é de cerca de 2h30 a 3h, mas imprevistos podem acontecer e estender ainda mais a viagem. Quem preferir uma opção mais confortável, pode optar por transfers, que irão pegá-lo no aeroporto e deixá-lo diretamente no hotel. Os preços podem ser bem maiores do que a viagem de ônibus. Finalmente, ir de carro também é uma opção. É possível alugar um veículo no aeroporto. No entanto, o carro não é imprescindível em Gramado, e dá para se locomover bem sem ele. Mas, para alguns pontos turísticos mais distantes, acaba agilizand o trajeto. Como você já deve imaginar, Gramado fica bem cheia nessa época, então pode ser mais difícil encontrar lugares para estacionar.

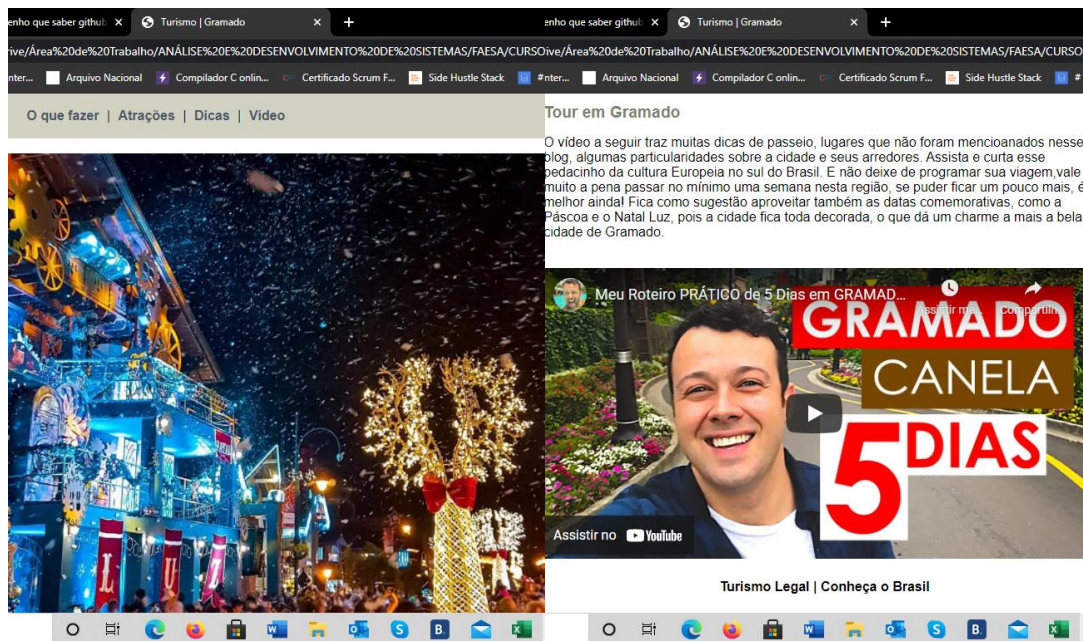


Hospedagem

Para quem precisa economizar, existem boas alternativas. Mas, nesse caso, é melhor evitar a alta temporada. Em épocas de eventos, como durante o Natal Luz, a taxa de ocupação dos hotéis de Gramado pode chegar a 95%. Os preços, obviamente, também sobem. Mas, como em todo destino, é possível encontrar preços a partir de R\$ 100,00 a diária. Para baratear a estadia, pode ser necessário ficar em lugares mais distantes do centro ou alugar apartamentos. Os valores de hotéis e pousadas em Gramado são acima de média, então se você quer economizar, uma dica é procurar opções de hospedagem pelo Airbnb. O site oferece opções de apartamentos ou quartos para alugar. É possível avaliar as opiniões de outros hóspedes anteriores, o que garante maior confiabilidade na hora de fazer a reserva. Na estrada entre Gramado e Canela, também é possível encontrar boas opções de hospedagem. Em Canela, cidade que fica distante cerca de 30 minutos de Gramado, também existem boas opções. Uma linha de ônibus liga as duas cidades e um trajeto de Uber custa em média R\$ 25,00, permitindo uma fácil locomoção entre as duas cidades.



Turismo Legal | Conheça o Brasil



Link do projeto: <https://github.com/EnricoLaqua/BestTrip.git>.