**Digital Innovation One**

***Introdução à lógica e à programação***

**Algoritmos:** é simplesmente uma "receita" para executarmos uma tarefa ou resolver algum problema.

**Pseudocódigo:** um modo de escrever um código com linguagem simplificada.

**Fluxograma:** é uma ferramenta utilizada para representar uma sequência lógica e coerente do fluxo de dados. Sendo um diagrama, uma representação esquemática.

**Variável:** Local utilizado para armazenar informações. Sendo um espaço na memória que pode ser alterado durante a execução. (Normalmente declarada no início)

* **Constantes:** Valores que não serão alterados

**Expressões Aritméticas:** São expressões que utilizam operadores aritméticos, envolvendo Constantes e Variáveis. (- + / \* %)

**Operadores relacionais:** Expressões composta por outras expressões ou variáveis numéricas com operadores relacionais, retornam verdadeiro e falso (valores lógicos). ( < > >= <= == !=)

**Tomada de decisão:** Se trata de uma decisão baseada em uma condição encontrada dentro da execução.

**Concatenação:** É um termo usado para designar a operação e o unir o conteúdo de duas strings(sequência de caracteres). Agrupamento de duas ou mais células incluindo formulas ou textos em um único resultado. ( &, + )

***Introdução ao Portugol***

**Estrutura/Laços de Repetição:**  Permite executar mais de uma vez o mesmo comando, de acordo com uma condição ou um contador.

**Linguagem de Programação:** é uma linguagem escrita e formal que especifica um conjunto de instruções e regras usadas para gerar programas. A função é servir como meio de comunicação entre computadores e humanos.

**Alto Nível:** A sintaxe se aproxima mais da nossa linguagem e se distancia da linguagem da máquina. (C, php,, C++, sharp, phayton)

**Baixo nível:** Está mais próxima da linguagem da máquina. É necessário ter conhecimento direto da arquitetura do computador para fazer alguma coisa. (sensui)

**Compiladas:** É uma linguagem em que o código fonte, é executado pelo sistema operacional ou pelo processador, após ser traduzido por meio de um processo chamado compilação. (C-sharpe, visual base, c==, delfi)

**Interpretadas:** O código fonte é executado por um programa de computador chamado interpretador, que em seguida é executado pelo sistema operacional ou processador. (JS,php, phyton)

**Portugol:** é uma pseudo linguagem que permite ao leitor desenvolver algoritmos estruturados em portugues de forma simples e intuitiva, independente de linguagem de programação. permite ao programador pensar no pproblema em si, nãao no equipamento que irá executar o código.

Site para o portugolStudio:

<https://github.com/UNIVALI-LITE/Portugol-Studio/releases/>

**Desvios Condicionais:**

**Se:**  é utilizada a palavra **se** a condição está entre parentes ( ) e o que deve acontecer deve estar entre { }

**senão:**  se acondição do  **se**  for falsa, um outro conjunto de comandos deve ser executado.

**caso:** Similar aos comandos **se** e **senão** reduz a complexidade na escolha de diversas opções. Não utiliza operadores lógicos, apenas trabalha com valores definidos.

**Comentários:** são instruções que o commputador não irá considerar. Serve para organizar o codigo

**Matriz:** Uma coleção de variáveis, acessiveis com um unico nome e com sequencia

**Índice:** a individualização de cada variável de um vetor é feita através do uso de índices

**Vetores:** são matrizes de uma só dimensão

***Introdução ao Git e ao GitHub***

**Comandos no terminal: (Prompt)**

**dir -** Lista as páginas do usuário

**cd -** Change direction, leva para o diretório desejado

**\** - foi para a base do local

**cd .. -** Retroceder uma pasta

**cls -** Clear Screen, limpar o terminal

**TAB -** Auto Completar as informações

**mkdir -** Make Directory , criar pasta. Adiciona o nome após o espaço.

**echo -** Printa no diretório a mensagem

**del -** Deletar os arquivos dentro do diretório, pasta.

**seta para cima -**Ver o histórico de comandoss

**rmdir -** Deletar o diretório, pasta

mv - mover

**Comando Git**

**ls** - Listar arquivos

**cat** - Visualizar o conteúdo da chave

**ctrl + l** - Limpa o diretório do git bash

**git init** - cria um novo repositorio da git

**ls - a** - mostrar pastas ocultas

git status - trás informações

***Instalação do Git***

Git é um sistema de controle de versão distribuído [gratuito e de código aberto projetado para lidar com tudo, desde projetos pequenos a muito grandes com velocidade e eficiência.](https://git-scm.com/about/free-and-open-source) Se trata de uma forma segura para garantir a não alteração dos dados. É um sistema distribuído Seguro.

\*Estamos utilizando o terminal Git Bash

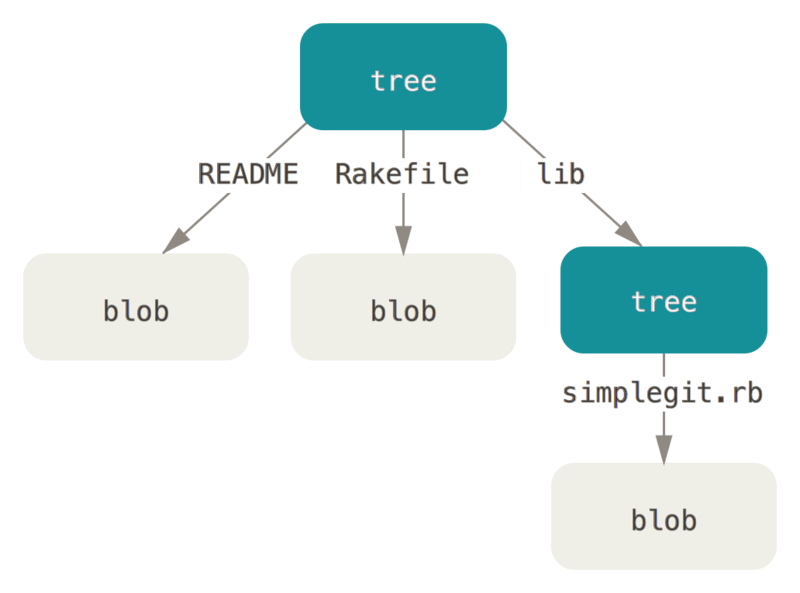
https://git-scm.com/download/win

**SHA1 -** Secure Hash Algoritim, conjunto de funções hash criptográficas projetadas pela NSA. Pega o arquivo e embaralha de uma forma específica. A encriptação gera conjuntos de caracteres identificador de 40 dígitos. É uma forma curta de representar um arquivo. Ao alterar o arquivo o código é alterado

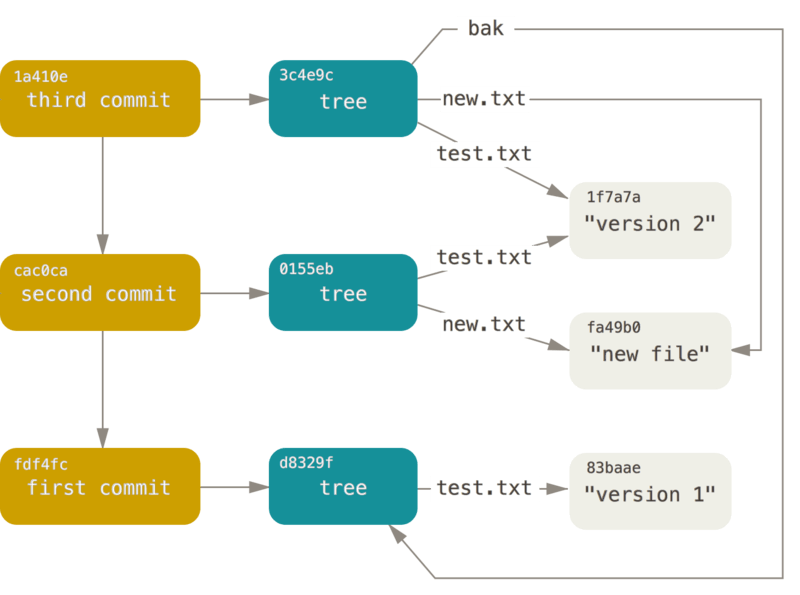
**ex:** *openssl sha1 (nome do arquivo)* -gera o código

**BLOBS -** É um objeta que armazena os metadados, como o *sha1 do arquivo*,

*‘blob tamanho\0conteudo’*

**TREE -**  Armazena BLOBS e nome do arquivo, tem os metadados dos blobs e das ouutras arvores.

**COMMIT -** Ele aponta para uma árvore, parente, autor, mensagem, timestamp. Significa uma alteração. Eles também possuem SHA1.



**Chave SSH -** Conexão segura e confiável entre máquina e GitHub. (github -> setting -> chavve SSH)

**Gerando a chave -**  No git Bash *ssh-keygen -t ed25519 -C* [*mariavitoriaogait@gmail.com*](mailto:mariavitoriaogait@gmail.com)

Cria a chave no git e põe o código criado.

E agora inicializar o ssh - *$ eval $(ssh-agent -s)*

**Criando um repositório**

**git init**  -O comando git init cria um novo repositório do Git. Ele pode ser usado para converter um projeto existente e não versionado em um repositório do Git ou inicializar um novo repositório vazio.

-configurações do git

**git config --global user.email "mariavitoriaogait@gmail.com"**

**git config --global user.name MaVik**

-add arquivo Markdown: é um jeito mais humano de escrever HTML, uma estrutura simples

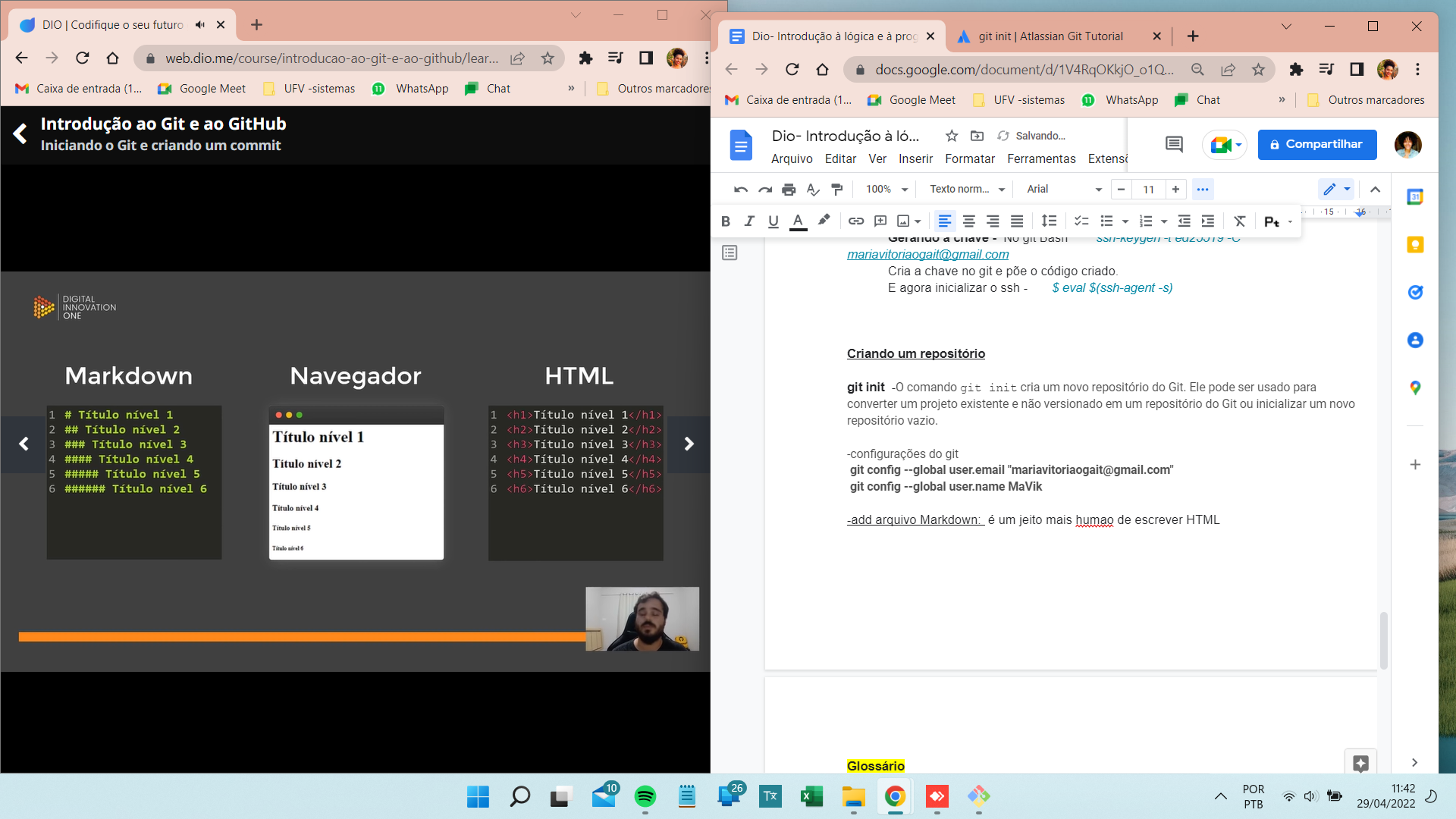
**Taypora -** é o programa que vai abrir o .md PAGO

Formata com símboloss

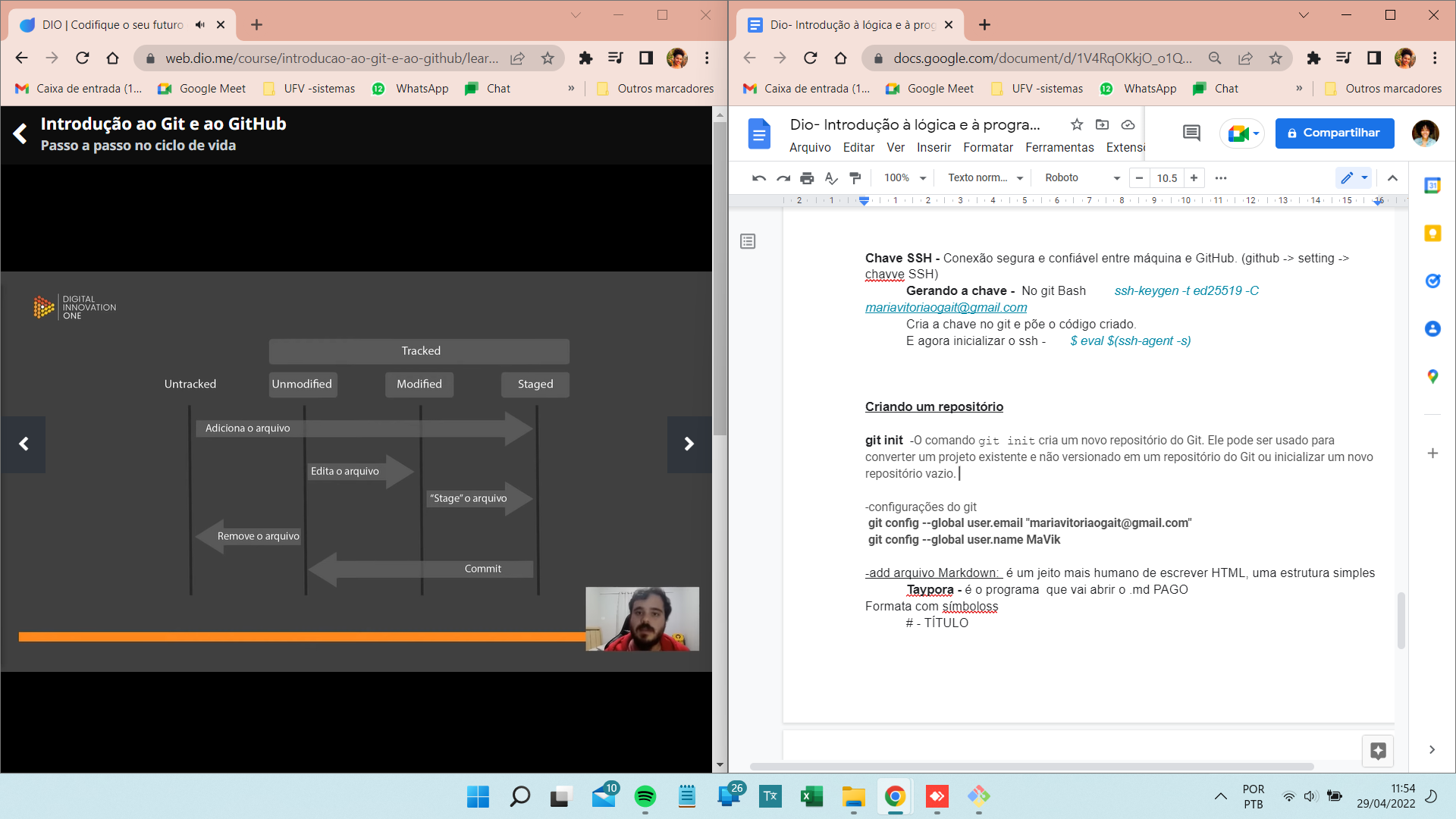
# - TÍTULO

\*\* \*\* - NEGRITO

\_ \_ - ITALICO



—----



**git add -**  a gente acabou de criar um arquivo “Untracked”

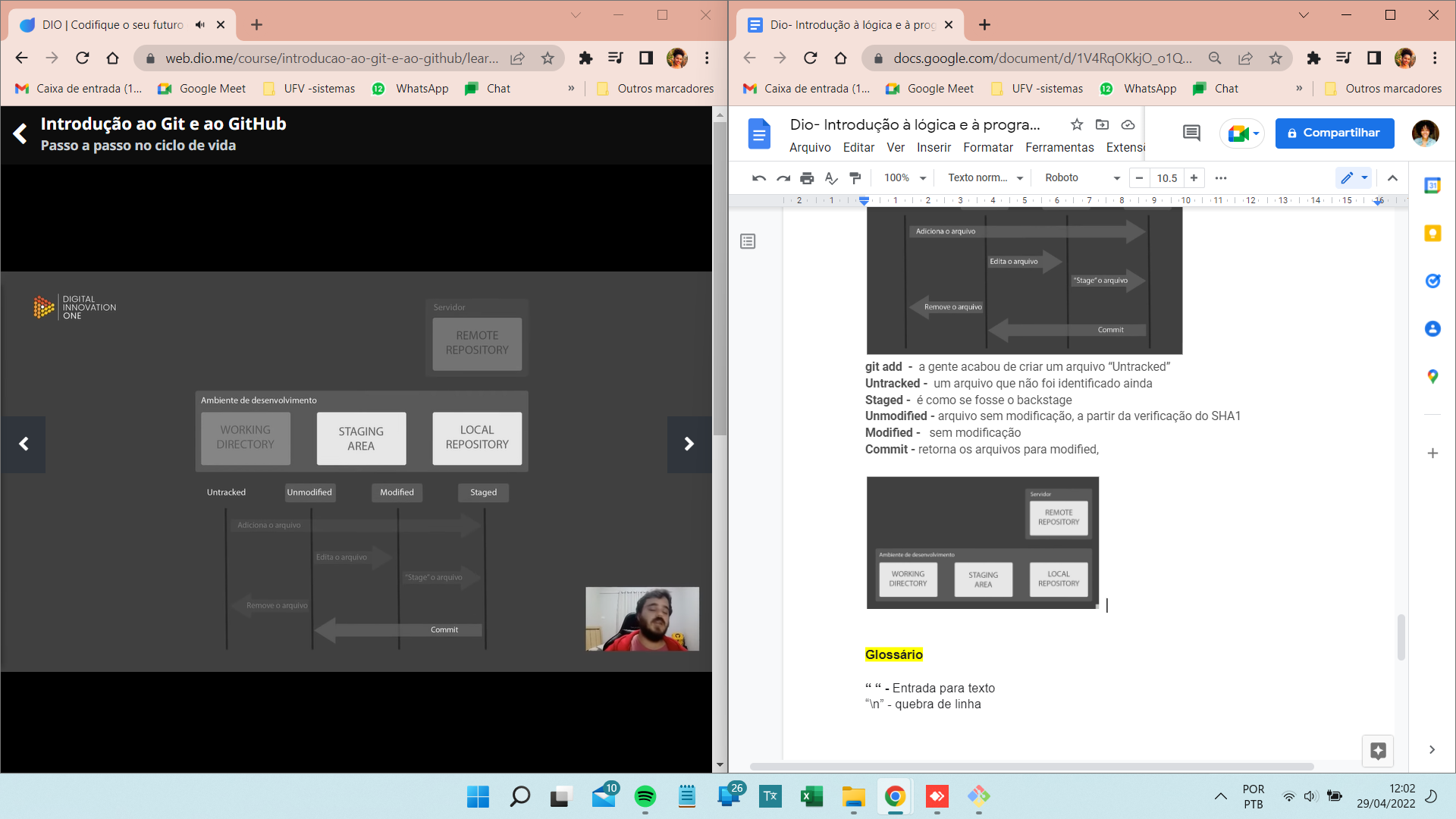
**Untracked -**  um arquivo que não foi identificado ainda

**Staged -**  é como se fosse o backstage

**Unmodified -** arquivo sem modificação, a partir da verificação do SHA1

**Modified -**  sem modificação

**Commit -** retorna os arquivos para modified,



De forma prática;

**Incluir pastas para o Github remoto.**

* Cria o repositorio c/ README
* Clona o link em HTTPS - lá onde faz download do .zip
* Cria uma pasta para receber
* Abre o “git bash”
* git clone <link>
* git status - verifica se tem alguma atualização
  + se tiver, vamos atualizar
  + git add .
  + git commit - m “comentário”
  + git push origin main

**Glossário**

**“ “ -** Entrada para texto

“\n” - quebra de linha