**RELATÓRIO**

1. **ATIVIDADE PROPOSTA:**

Criação de um código que fornecesse informações de forma iterativa para o usuário em relação as informações de produção e acidente. Essas informações deverão ser apresentadas através de gráficos.

1. **METODOLOGIA UTILIZADA**

Baseando-se nas premissas da atividade, foi proposto pelo Thiago a utilização da linguagem de programação Python, utilizando mais especificamente a documentação Dash.

Essas ferramentas foram utilizadas para a filtragem dos dados e a plotagem dos mesmos, optando-se pela apresentação dos dados de produção através de gráficos com linhas e pontos e os dados de acidentes através de gráficos de barras.

1. **DADOS UTILIZADOS**
   1. **Produção**

Os dados utilizados para a Produção foram todos retirados do site do SAFF através do seguinte comando:

**Siade 🡪 Consulta 🡪 Produção 🡪 Produção de Transporte (Avançado)**

E nessa página, foram escolhidos os seguintes filtros:

* **Período: 01/2006 a 09/2019**
* **Informações Adicionais:** 
  + - **Mês/Ano;**
    - **Ferrovia;**
* **Valores:**
  + **TU;**
  + **TKU;**
  1. **Acidentes**

Os dados para os gráficos de Acidentes foram todos retirados do site do SAFF através do seguinte comando:

**RAAF 🡪 Consulta 🡪 Registro de Acidente (Ferrovia)**

E nessa página foi filtrado o seguinte intervalo de tempo:

* **Ocorrência: 01/01/2006 a 31/12/2019**
  1. **Índice de Acidentes**

Acessando o site <http://www.antt.gov.br/ferrovias/arquivos/Anuario_Estatistico.html> no dia 17 de outubro de 2019, foi realizado o download das tabelas referente ao anuário estatístico de cada ferrovia.

Uniu-se as informações referente aos índices anuais dessas tabelas em um único arquivo, sendo este último o arquivo utilizado para a plotagem dos gráficos.

**Observações:** as bases de dados foram geradas em formato Excel (xlsx) e trabalhadas em python.

Outro detalhe importante, é em relação aos dados de Acidentes, foi criado uma nova coluna nomeada ‘Acidentes’ e numerou-se assim os acidentes.

1. **CÓDIGO**

Apenas uma breve observação em relação ao código, em que o detalhe que possibilitou a resolução dessa atividade que foi a criação de um loop nas ferrovias selecionadas. Dessa forma, esse loop auxiliou na filtragem dos dados e na criação dos parâmetros a serem utilizados para a plotagem dos gráficos.

1. **SITE**

Seguindo os objetivos da atividade, que consiste na criação de um template de site para apresentação gráfica referente aos dados de produção, acidentes e índices de acidentes. Baseando-se nisso, foi construído a estrutura do site apresentando os dados na seguinte ordem:

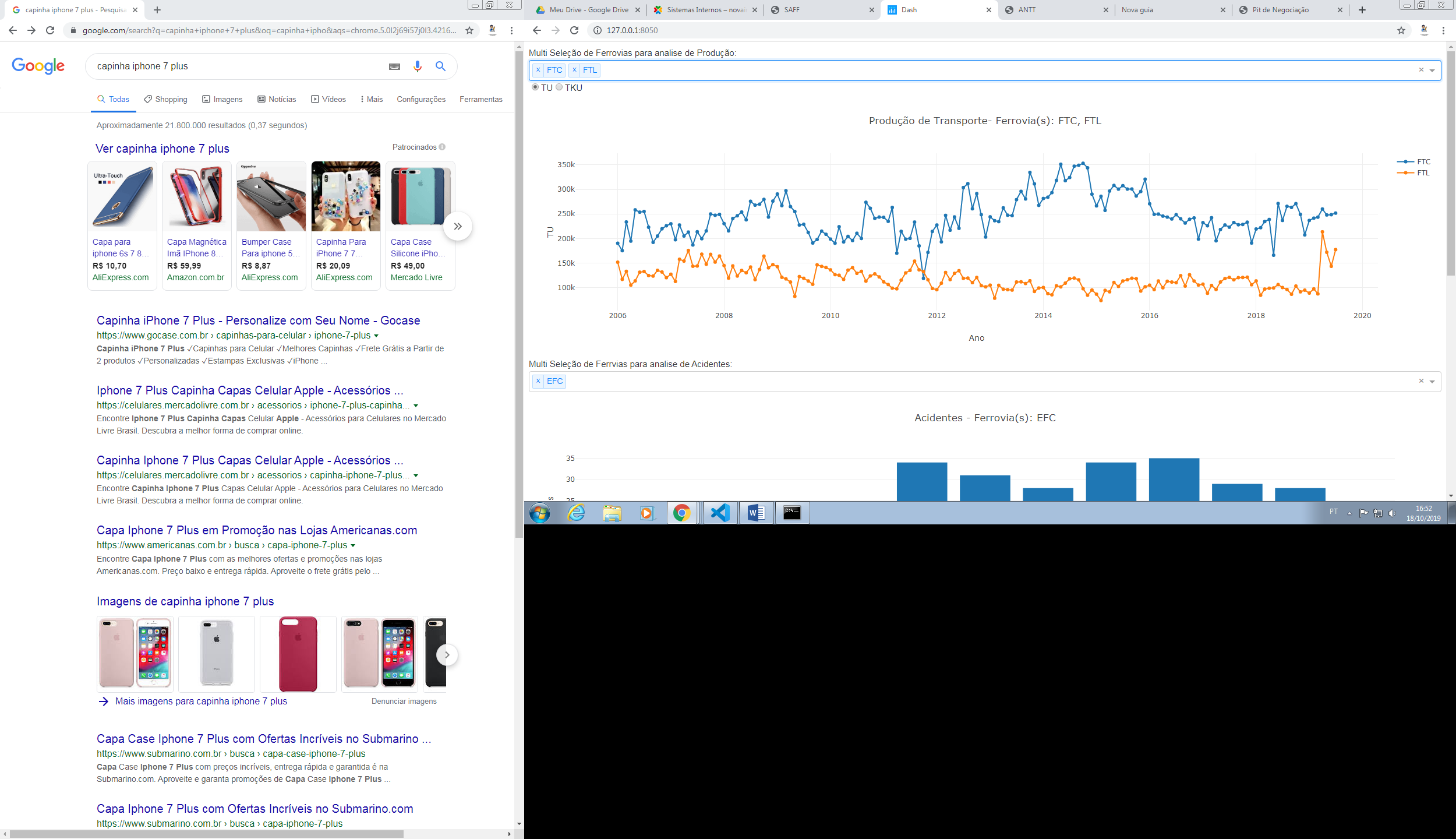
* **Produção**
* **Acidentes**
* **Índices de Acidentes** 
  1. **Produção**

Desse modo, os dados de produção foram apresentados com as seguintes iterações com o usuário:

* Escolha das Ferrovias a serem analisadas;
  + Vale ressaltar que optei (juntamente com uma dica do Thiago) pela opção de ‘*multi select’* que permite a seleção de uma ou mais ferrovias, podendo analisa-las simultaneamente.
* Escolha entre os dados de TU ou TKU;

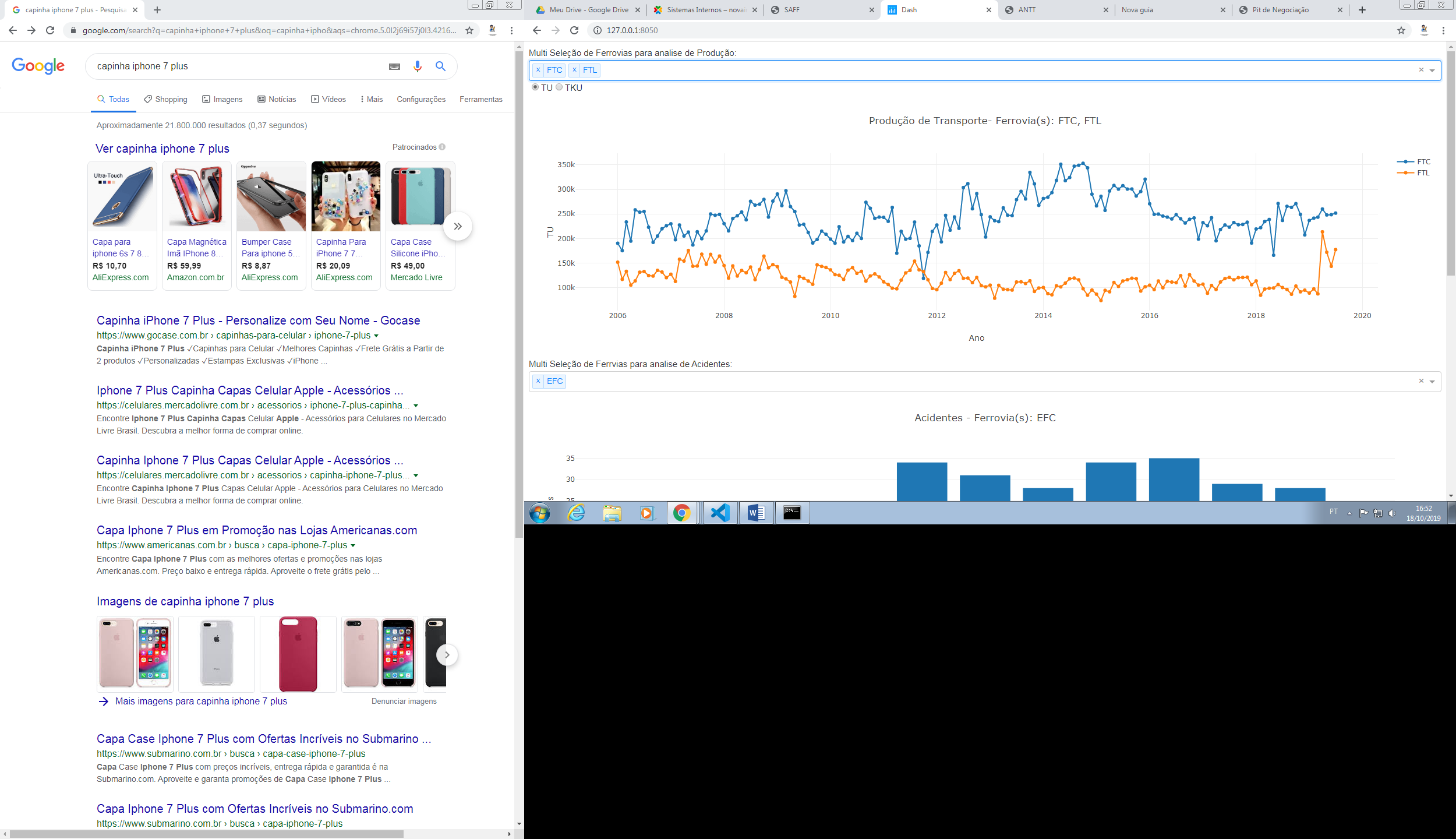
Após a escolha dessas opções, é apresentado o gráfico referente as escolhas do usuário. Assim, é apresentado um gráfico em que no eixo x varia no tempo, enquanto no eixo y varia a TU ou TKU (de acordo com a escolha do usuário).

A seguir é apresentado um exemplo onde foram selecionadas as ferrovias FTC e FTL com os dados referentes a TU.



Observando a imagem, pode-se notar que na região superior direita encontra-se a legenda referente à cada ferrovia selecionada. (Um detalhe: o próprio sistema automaticamente cria as legendas e cores para cada ferrovia).

Para uma melhor visualização:



Analisando uma imagem ampliada, pode-se notar a região de seleção de ferrovia(s) e logo abaixo os pontos de seleção referente à TU ou TKU.

Observação: o gráfico por ser variável, ele automaticamente selecionar a melhor opção de escala para a ferrovia escolhida.

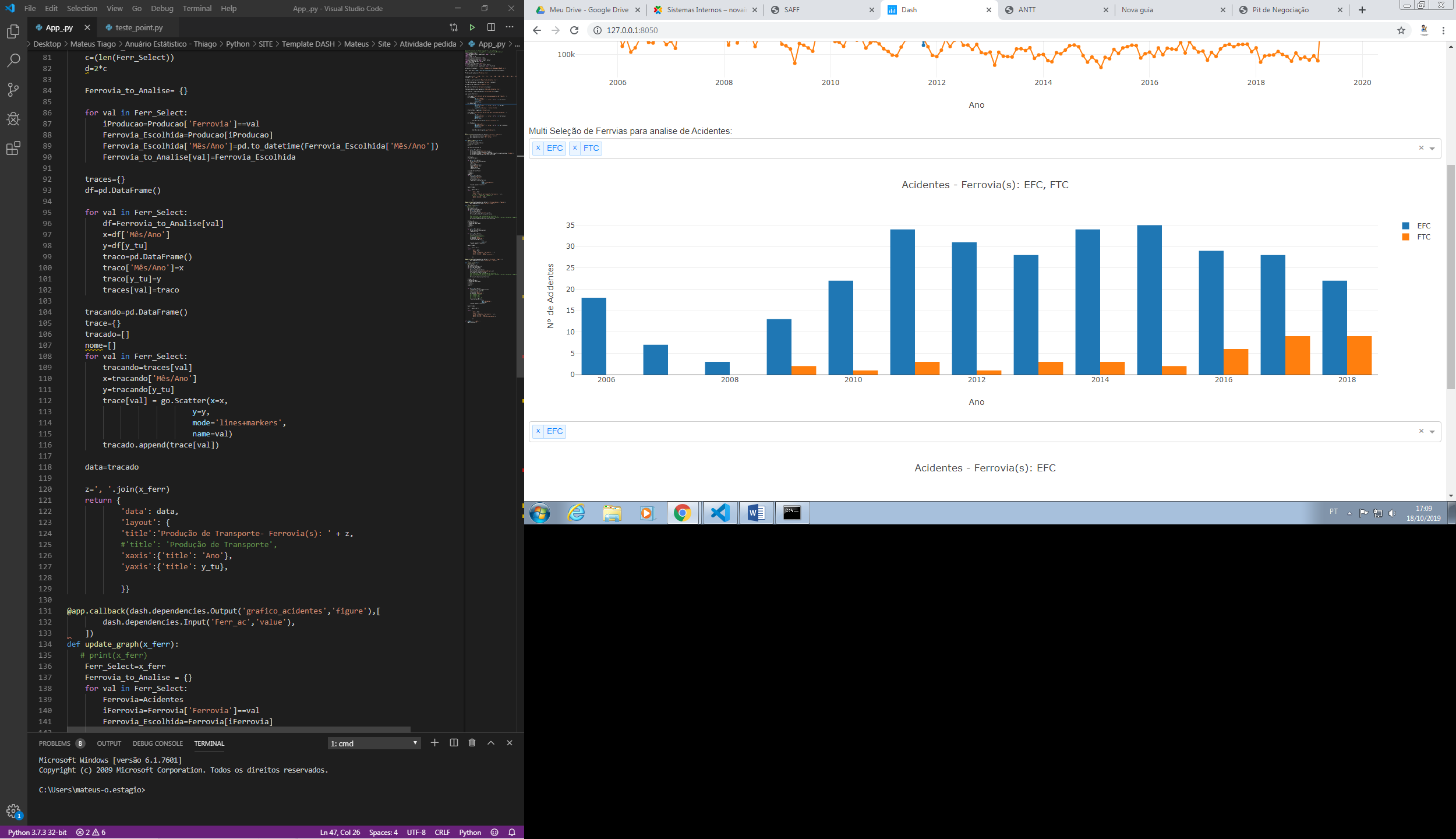
* 1. **Acidentes**

Desse modo, os dados de acidentes foram apresentados com a seguinte iteração com o usuário:

* Escolha das Ferrovias a serem analisadas;
  + Vale ressaltar que optei pela opção de ‘*multi select’* novamente, que permite a seleção de uma ou mais ferrovias, podendo analisa-las simultaneamente.

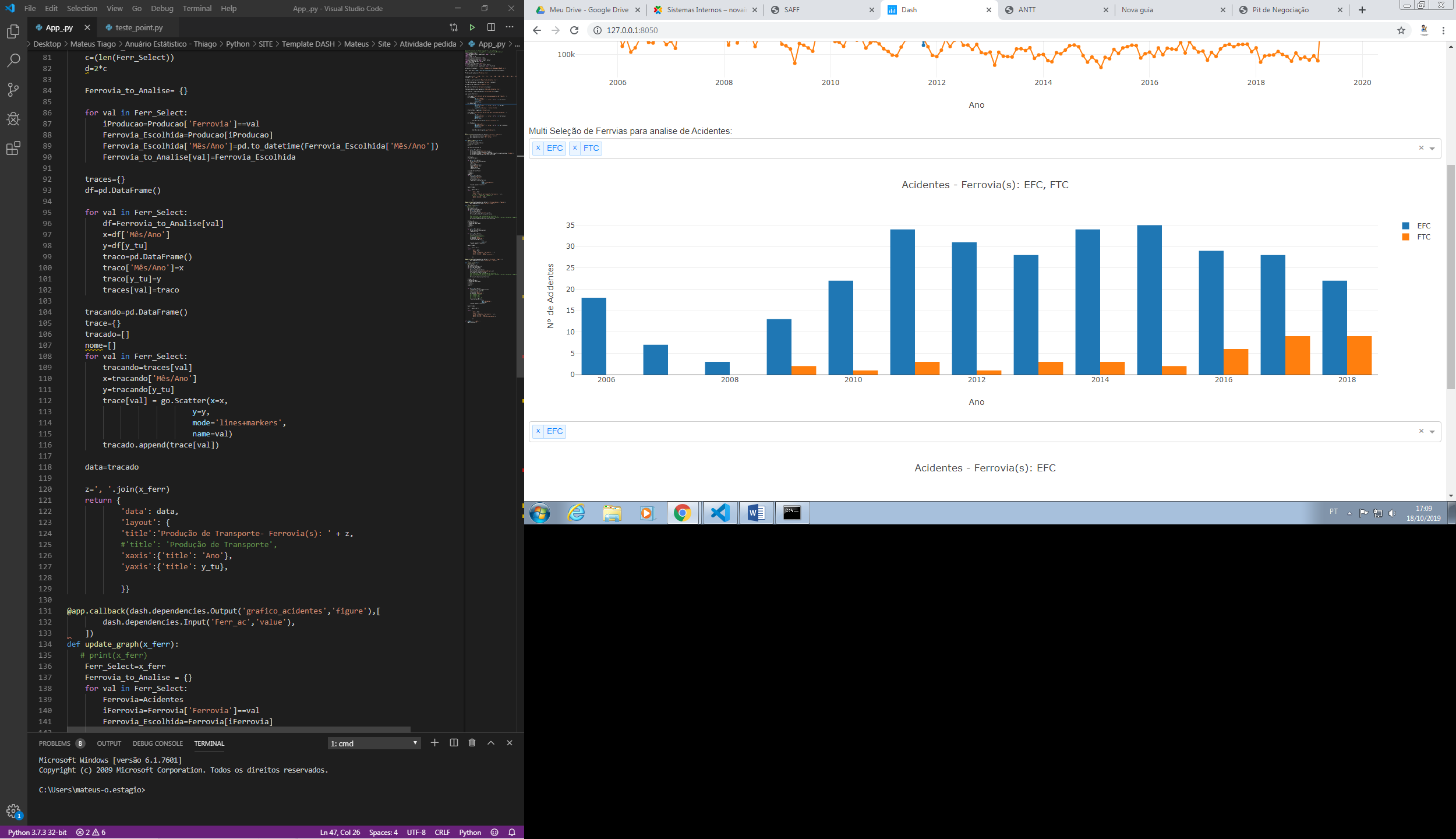
Após a escolha da(s) Ferrovia(s), é apresentado automaticamente os dados de acidente.

A seguir é apresentado um exemplo dos gráficos para uma seleção de dados das ferrovias EFC e FTC.



Observando a imagem, pode-se notar os mesmos detalhes que no gráfico anterior em que na região superior direita encontra-se a legenda referente à cada ferrovia selecionada. (Um detalhe: o próprio sistema automaticamente cria as legendas e cores para cada ferrovia).

Para uma melhor visualização:



Analisando uma imagem ampliada, pode-se notar a região de seleção de ferrovia(s).

Observação: o gráfico por ser variável, ele automaticamente selecionar a melhor opção de escala para a ferrovia escolhida.

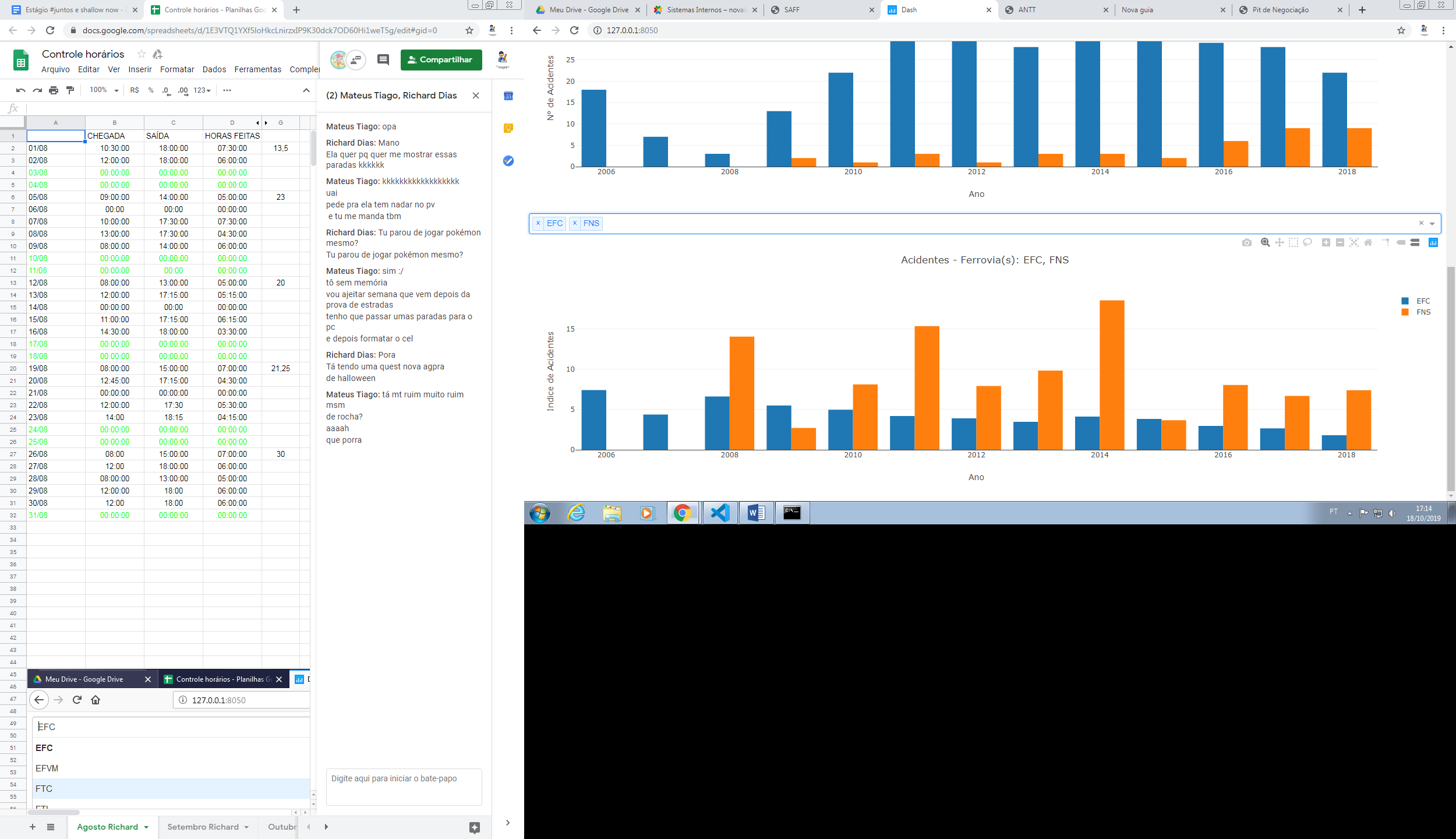
* 1. **Índice de Acidentes**

Desse modo, os dados de índice de acidentes foram apresentados com a seguinte iteração com o usuário:

* Escolha das Ferrovias a serem analisadas;
  + Vale ressaltar que optei pela opção de ‘*multi select’* novamente, que permite a seleção de uma ou mais ferrovias, podendo analisa-las simultaneamente.

Após a escolha da(s) Ferrovia(s), é apresentado automaticamente os dados referentes aos índices de acidente.

A seguir é apresentado um exemplo dos gráficos para uma seleção de dados das ferrovias EFC e FNS.



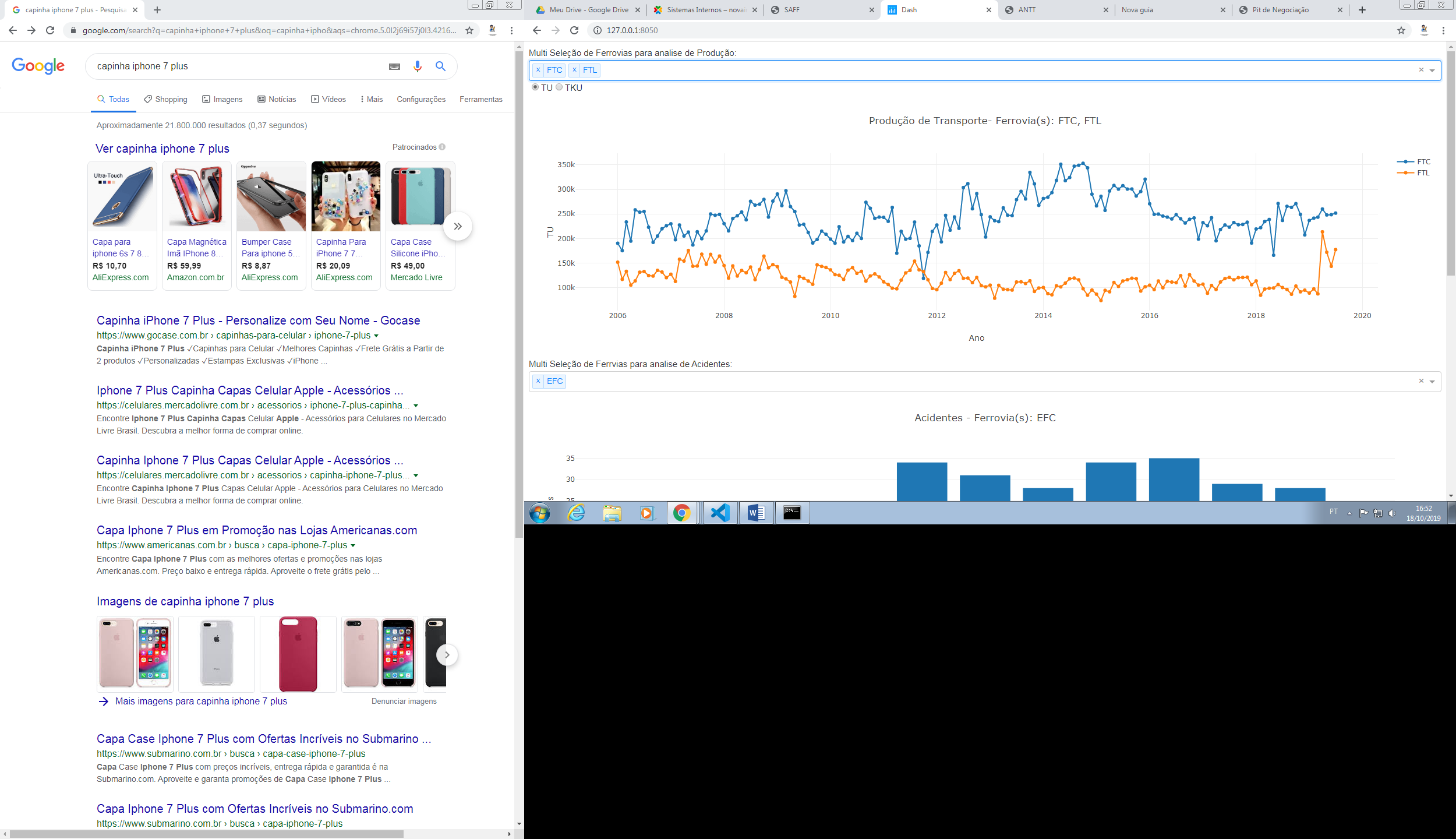
Observando a imagem, pode-se notar os mesmos detalhes que no gráfico anterior em que na região superior direita encontra-se a legenda referente à cada ferrovia selecionada. (Um detalhe: o próprio sistema automaticamente cria as legendas e cores para cada ferrovia).

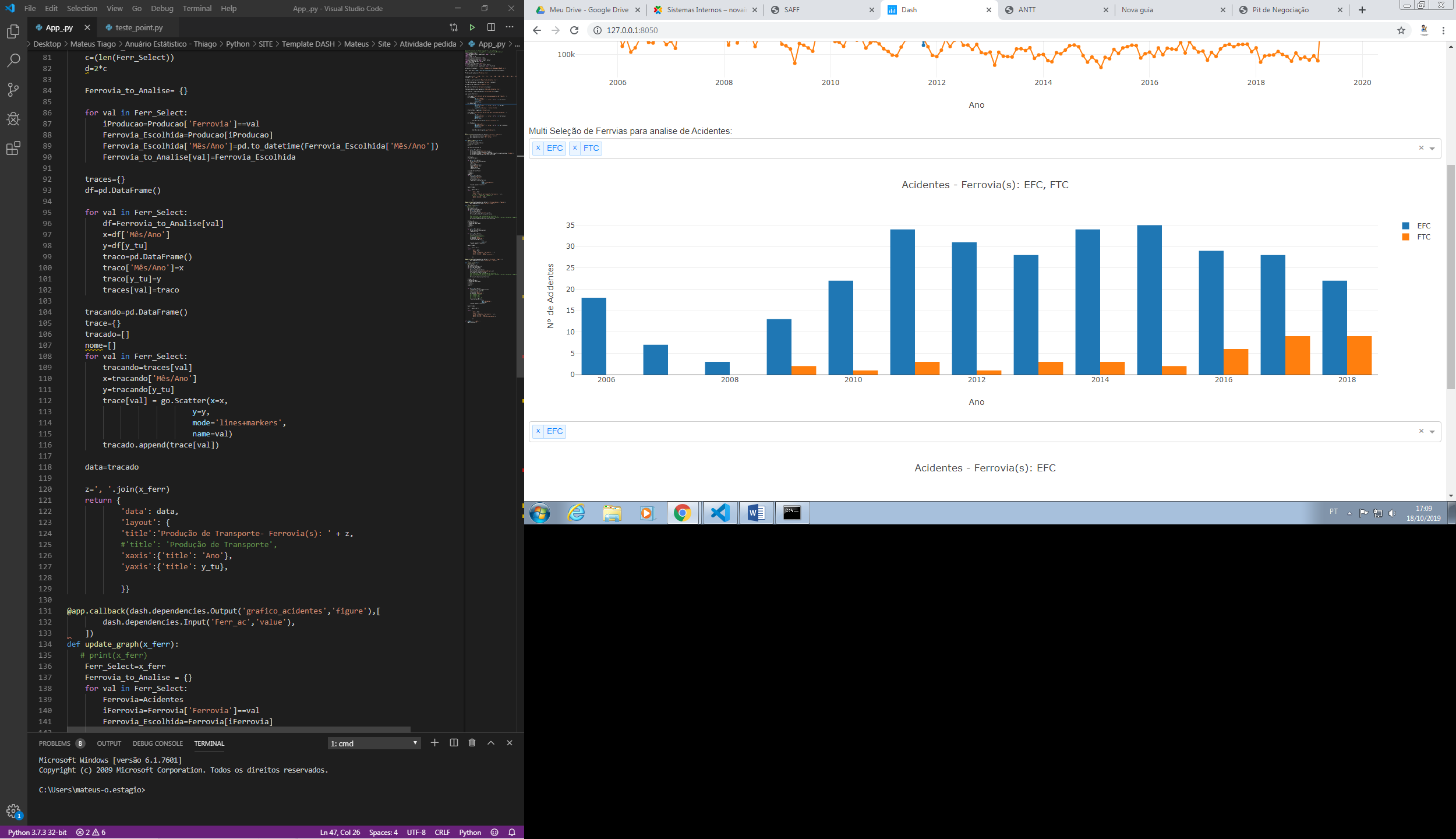
1. **CONCLUSÃO**

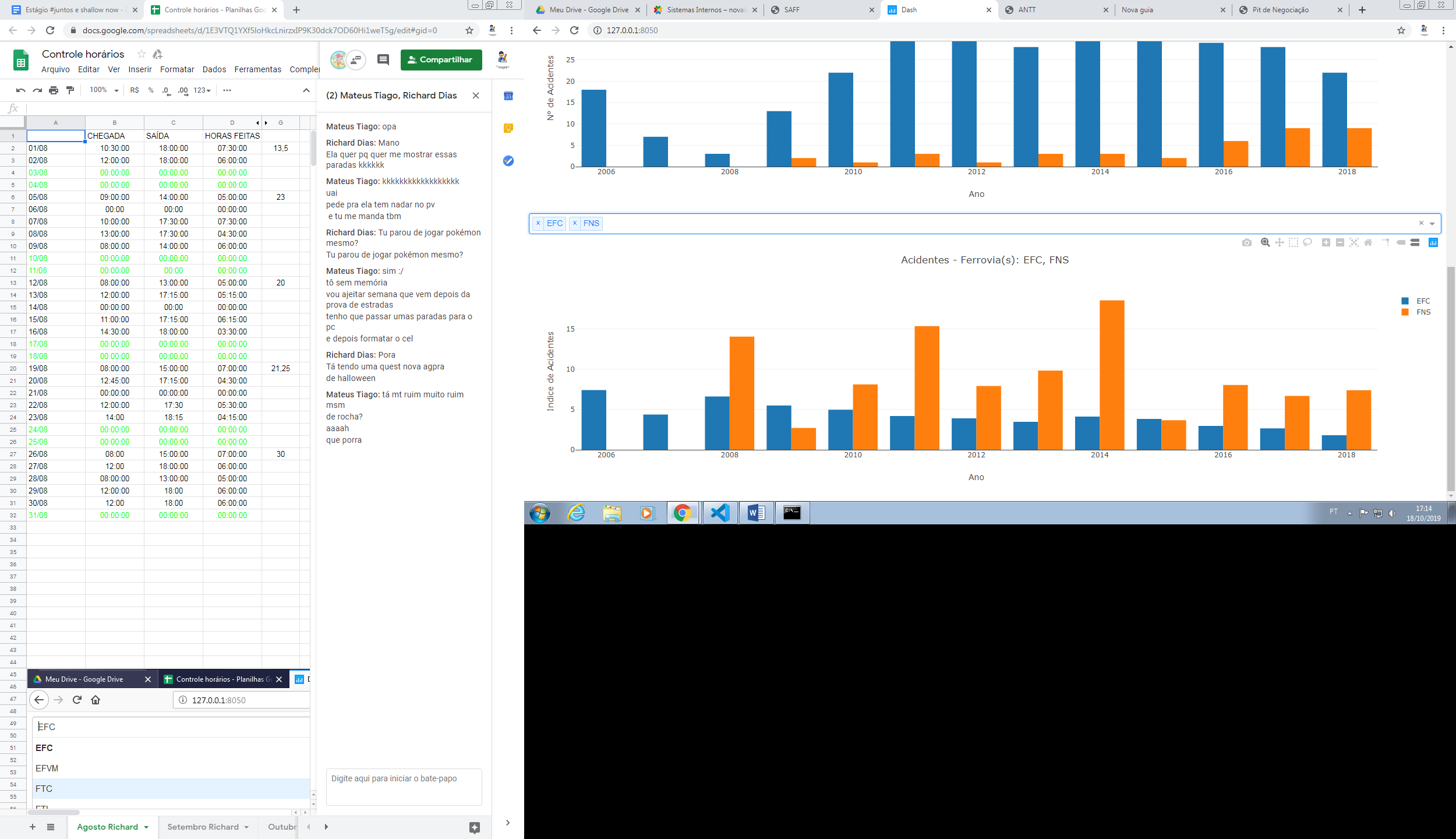
A apresentação das informações no formato de gráficos utilizando programação na linguagem Python, fazendo uso da biblioteca Dash foi um sucesso.

Assim, utilizando as bases de dados e o código pode-se gerar esses gráficos iterativos que atualizam quase que automaticamente no caso dos dados de acidentes e índices de acidentes enquanto que para os dados de produção possuem um tempo de carregamento de poucos milésimos de segundos. Isso ocorre devido os dados de produção estarem em Mês/Ano enquanto os dados de acidentes são apresentados em Ano. (Observação, ainda está sendo analisado essa parte do código para otimizá-lo e torna-lo mais eficiente).

O site ficou basicamente no seguinte estilo:







1. **REFERÊNCIAS**

<https://dash.plot.ly/>

<http://www.antt.gov.br/ferrovias/arquivos/Anuario_Estatistico.html>

<http://www.antt.gov.br/>

<http://intrantt/saff>

Além desses sites, foram utilizados inúmeros outros para solucionar determinados erros específicos relacionados a legenda do gráfico, manipulação de dicionários e listas e ainda como alterar/criar/editar DataFrames.

1. **BÔNUS**

A linguagem de programação em Python juntamente com a biblioteca Dash oferecem inúmeras possibilidades na geração de gráficos iterativos, podendo apresentar inúmeras informações de formas diferentes.

Devido a esses fatores, é possível a implementação de forma gráfica de outras informações.