# ArcMap

Inicialmente foi escolhido o ArcGis ArcMap para manipulação e edição dos mapas. O ArcGis oferece uma versão de testes (<http://desktop.arcgis.com/en/arcmap/>)gratuita por dois meses. Com esta versão é possível manipular os dados que geram os mapas, permitindo acrescentar, editar ou excluir valores da tabela de atributos.

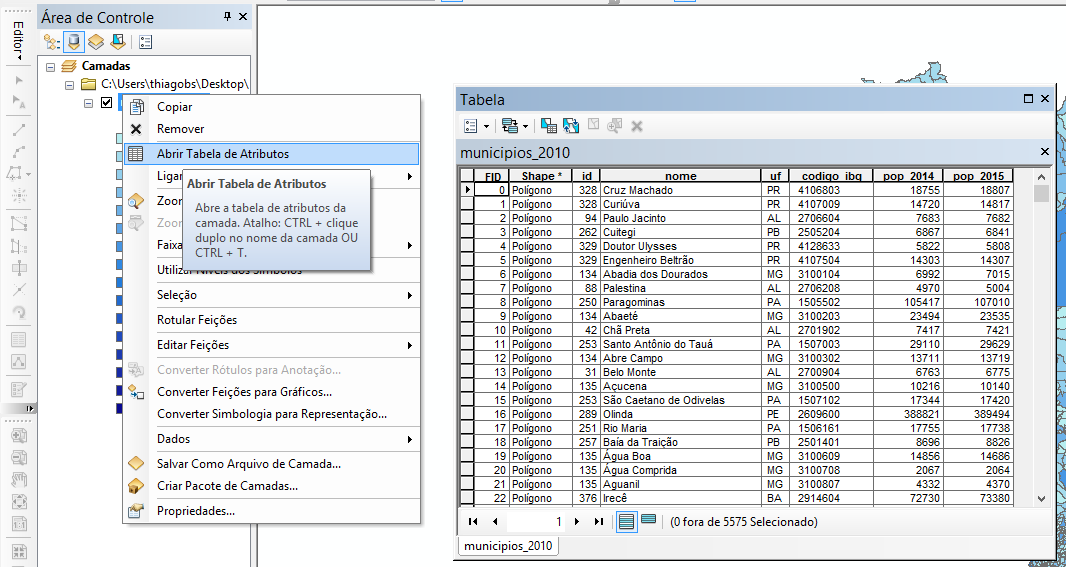


Figura - Exemplo Tabela de Atributos ArcMap

Com a tabela de atributos completa com todos os dados necessários para criação dos mapas é preciso converte-la em um arquivo no formato GeoJSON, que será utilizado posteriormente. Para isso, utilizamos o serviço online do ArcGis (<https://www.arcgis.com/home/signin.html?returnUrl=http%3A%2F%2Fdesktop.arcgis.com%2Fen%2Farcmap%2F>), que nos permite exportar o mapa criado com sua tabela de atributos para diversos formatos, incluindo o GeoJSON. Este serviço online também é disponibilizado de forma gratuita para testes por dois meses.

Para editar os valores da tabela de atributos, entre em *Editor -> Iniciar Edição.*

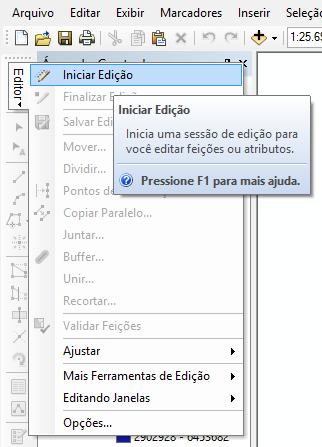


Figura - Iniciar edição na tabela de atributos

A importação da tabela de atributos para o ArcGis Online é feita dentro do próprio software ArcMap, é preciso apenas fazer *login* com a conta do ArcGis Online no ArcMap.

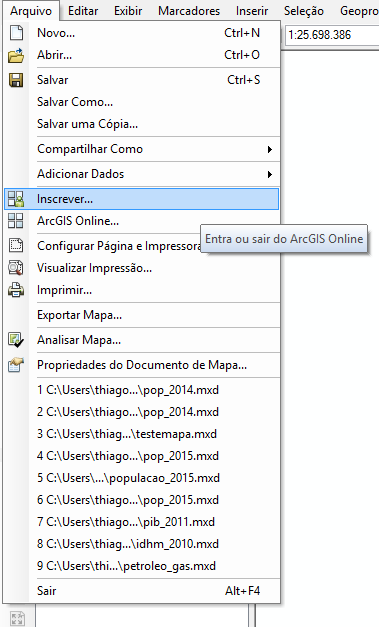


Figura - Login no ArcGis Online pelo ArcMap

Com o login efetuado, basta compartilhar o mapa como um serviço que será publicado no ArcGis Online.

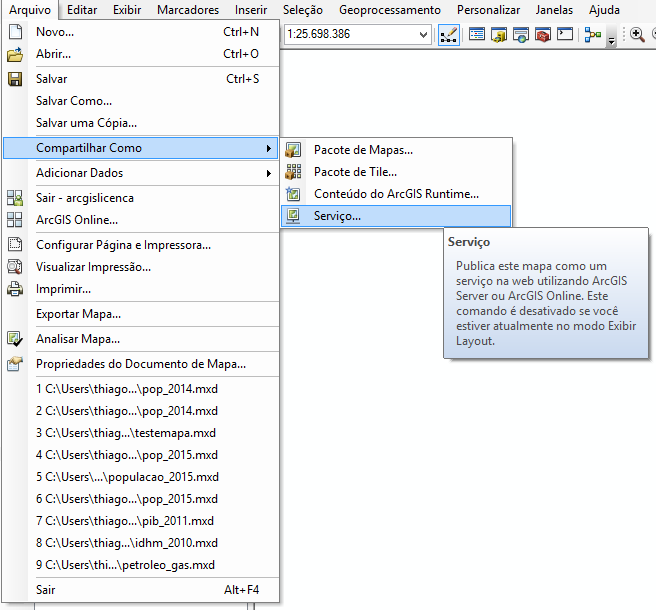


Figura - Compartilhar mapa como serviço

Feito isso, selecionar *Publicar um serviço* e *Avançar*.

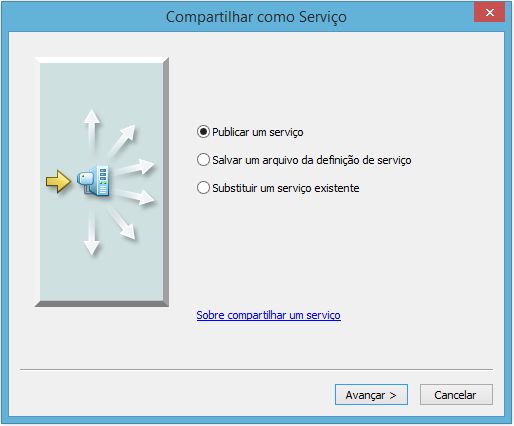


Figura - Primeira etapa para publicar o mapa

Fornecer um nome para o serviço e selecionar uma conexão. A conexão este relacionada a conta criada no ArcGis Online.

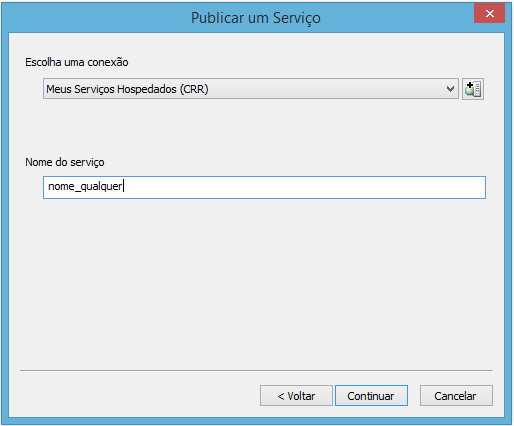


Figura - Selecionar nome e conexão

Então configure os parâmetros e outras informações necessárias para a publicação.

Antes de publicar o serviço, é importante verificar se não possui nem um erro ou restrição no mapa (clicar em *Análise*). Caso isso ocorra, é preciso corrigir todos os erros antes de publicar. Se tudo estiver correto, clicar em *Publicar* e aguardar o mapa ser publicado no ArcGis Online.

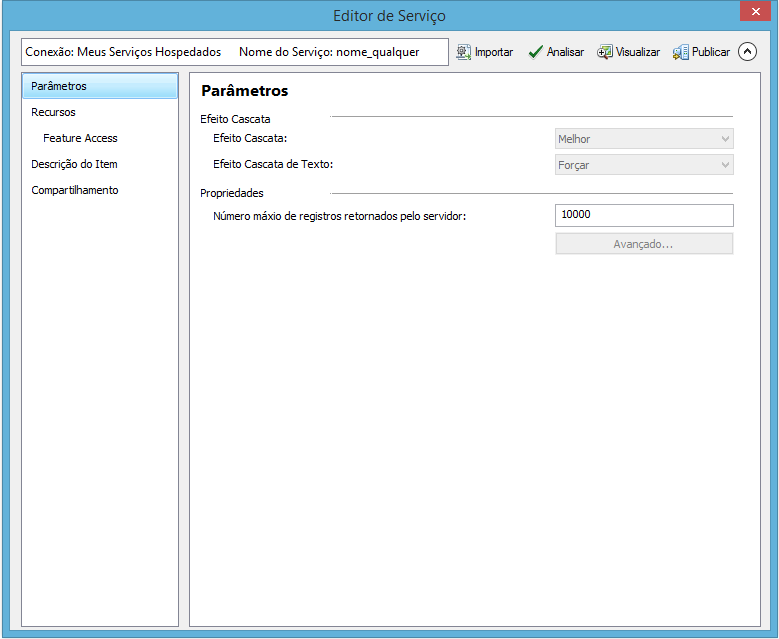


Figura - Publicação do serviço

Após a publicação, o mapa poderá ser encontrado na aba *Meu Conteúdo* no ArcGis Online.

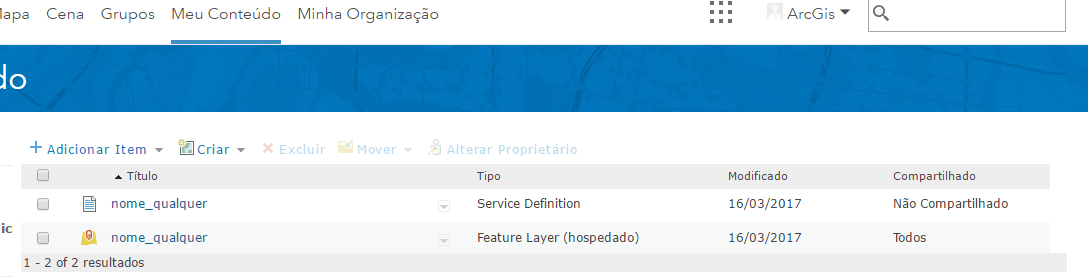


Figura - Mapa publicado no ArcGis Online

Ao entrar no mapa publicado, clicar na opção *Exportar Dados* -> *Exportar para GeoJSON*.



Figura - Exportar para GeoJSON

Feito isso, preencha os dados necessários para a exportação e clique em *Exportar*.

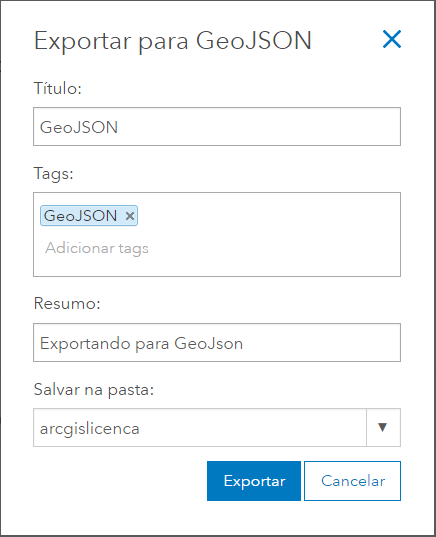


Figura - Preencher dados e exportar

Com o arquivo GeoJSON publicado é só fazer o download e salvá-lo. Este arquivo será muito importante para a criação do mapa online.

# Banco de Dados

O banco de dados foi criado em MySQL a partir de uma tabela do Excel. Este banco de dados é utilizado para realizar as consultas das buscas realizadas pelo usuário no site.



Figura - Trecho da tabela no Excel

Para fazer a exportação dos dados da tabela foi utilizado o plugin *MySQL for Excel* (<https://www.mysql.com/why-mysql/windows/excel/>). Este plugin permite exportar os dados do Excel para o MySQL sem que precisamos fazer isso manualmente.

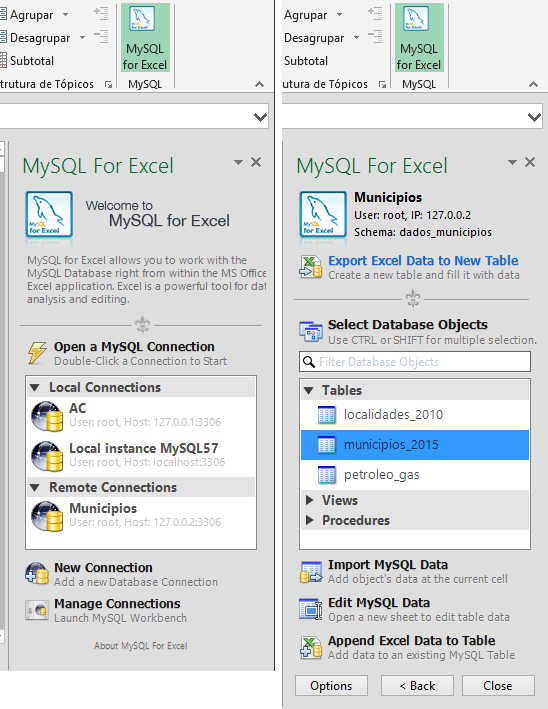


Figura - Exemplo plugin MySQL for Excel

Com o banco criado, é possível realizar consulta e conectá-lo na aplicação web.

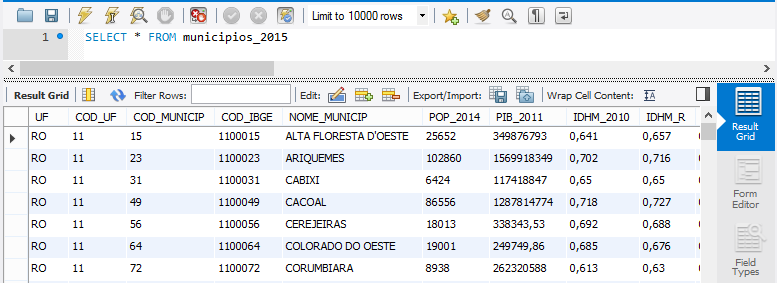


Figura - Select realizado no banco

Após a criação do banco, foi preciso conectá-lo a aplicação web. A conexão com o banco foi feita na linguagem PHP, utilizando a classe PDO (PHP Data Object) para isso.

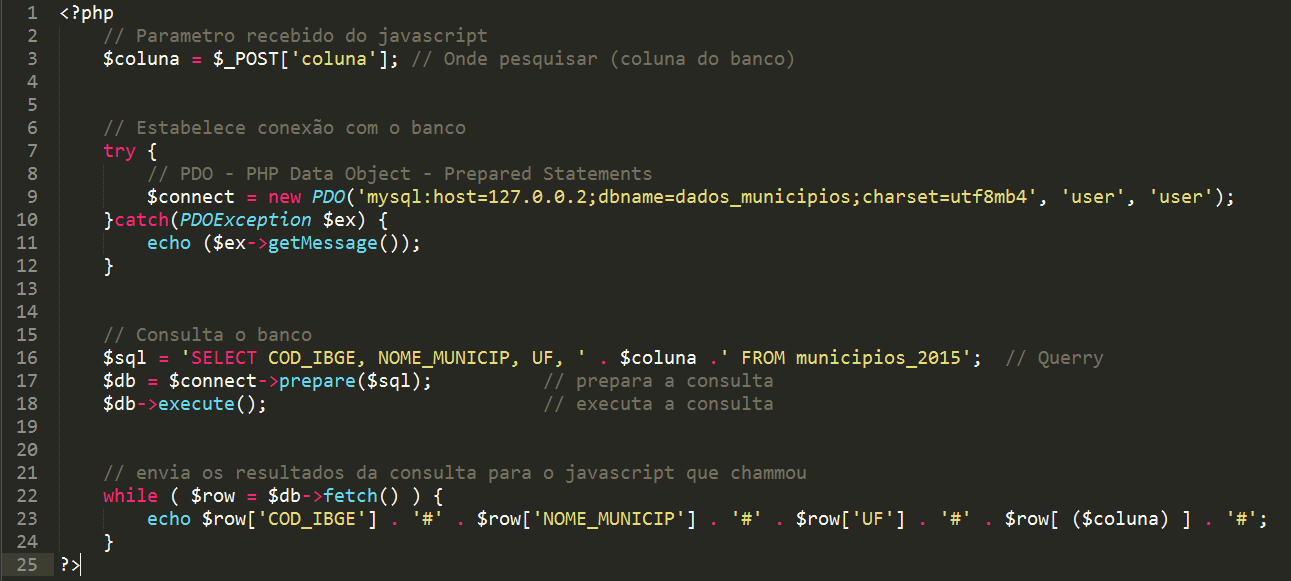


Figura - Código PHP para conexão com banco

Para chamar esta função no código JavaScript, foi utilizado Ajax com JQuery para poder acessar o PHP de forma assíncrona, sem a necessidade de recarregar a página a cada consulta do usuário.

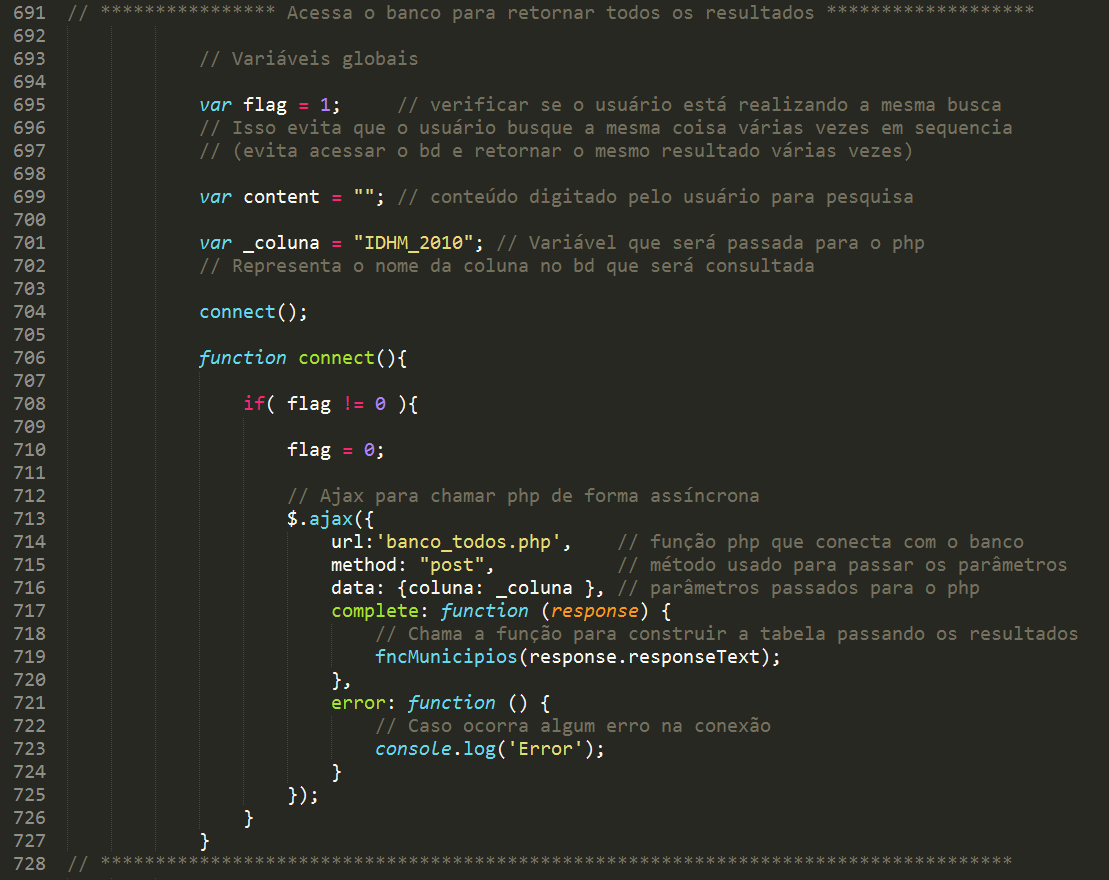


Figura - Exemplo Ajax para conexão com banco

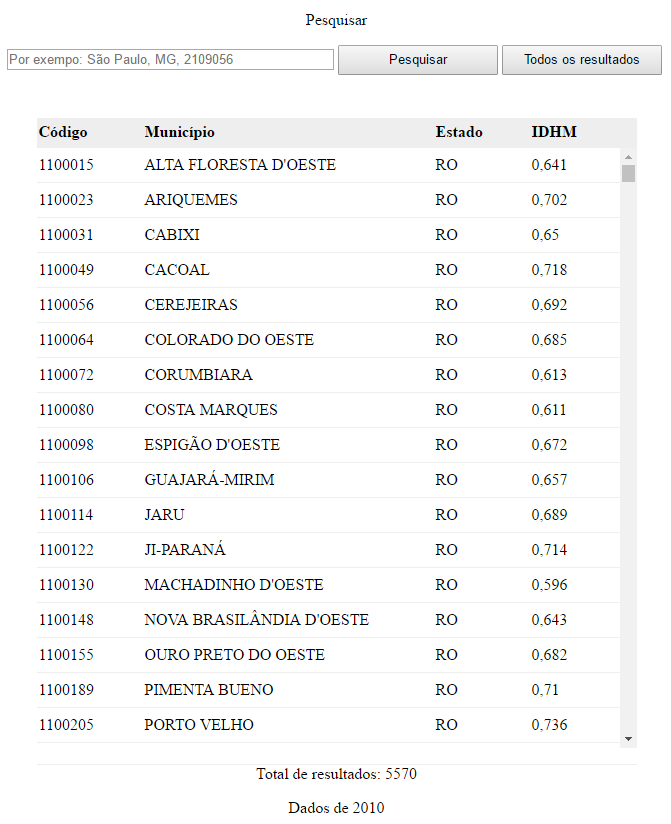


Figura - Exemplo tabela com resultados

# Leaflet

O Leaflet (<http://leafletjs.com/>) é uma biblioteca JavaScript para a elaboração de mapas interativos online, que nos fornece uma base para criação dos nossos mapas.

Primeiro é preciso importar os arquivos CSS e JavaScrip do Leaflet para podermos usá-los.

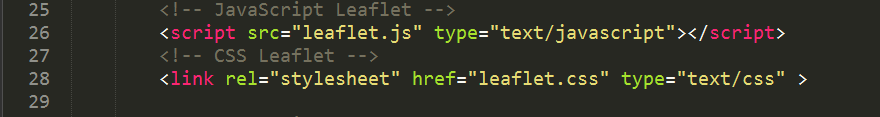


Figura - Importação do Leaflet

Para criar este mapa é bem simples, basta ter uma variável que receberá o mapa do Leaflet e tudo o que aplicarmos a ele utilizando o método *.addTo(map)*.

O *basemap* refere-se ao etilo do mapa utilizado no Leaflet. Vários estilos diferentes podem ser encontrados em <https://leaflet-extras.github.io/leaflet-providers/preview/> e podem ser trocados caso haja necessidade.

Foi utilizado o tutorial *Interactive Choropleth Map* (<http://leafletjs.com/examples/choropleth/>) para a criação das outras funcionalidades do mapa.



Figura - Criação mapa Leaflet

A documentação completa do Leaflet pode ser encontrada em <http://leafletjs.com/reference-1.0.3.html> para consulta e entendimento das funcionalidades do Leaflet.

# Testando a aplicação

Para realizar alguns testes na aplicação foi utilizado o Ngrok (<https://ngrok.com/>), que cria uma URL para um servidor local no computador, permitindo acessar remotamente a aplicação e testá-la. O comando do Ngrok para criar essa URL é *ngrok http 80.*

Para criar um servidor PHP localmente no computador, foi utilizado a ferramenta Xampp (<https://www.apachefriends.org/pt_br/index.html>), executando o módulo do servidor Apache na porta 80.

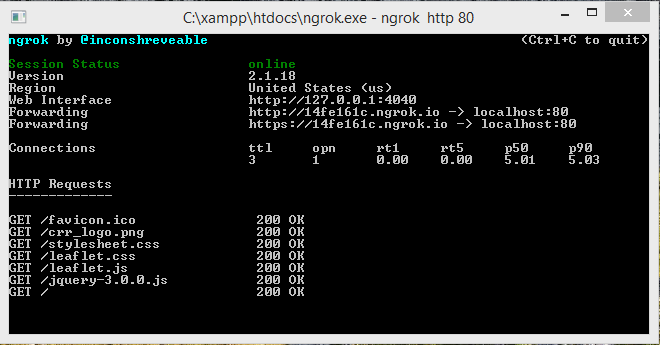


Figura - URL criada no Ngrok

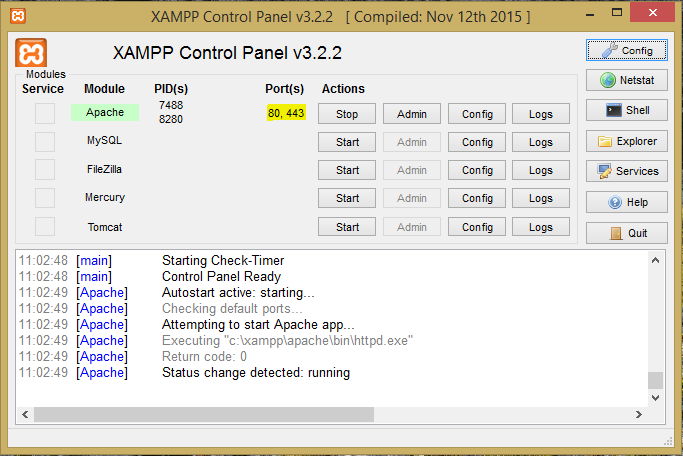


Figura - Xampp executando Apache

# Observações

Todos os códigos feitos e os relatórios estão disponíveis em uma pasta compartilhada no Dropbox (<https://www.dropbox.com/sh/8jrpkotywzqd2pe/AACT3sET_fmoRpqswFqjwn3Ka?dl=0>) e no Github (<https://github.com/tbsouza/CRR>).