

Mãos na massa: Habilitando CORS

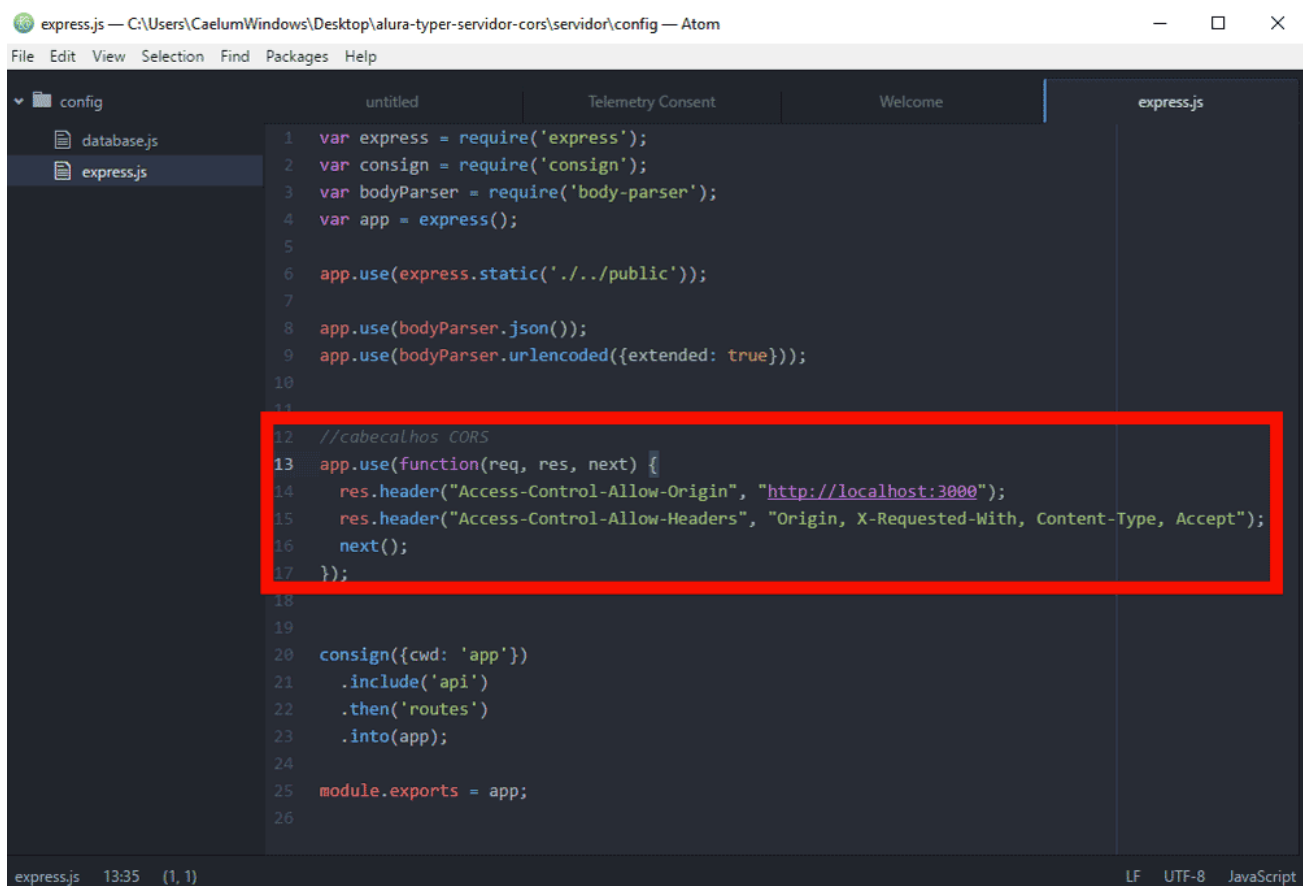
Agora vamos habilitar o CORS no projeto servidor que você acabou de instalar (da pasta `alura-typer-servidor-cors/servidor`). Tenha muito cuidado nesse exercício para não errar as pastas e configurações.

1) Primeiro pare a servidor da pasta `alura-typer-servidor-cors`.

2) Abra o arquivo `express.js` dentro da pasta `alura-typer-servidor-cors/servidor/config`.

Nesse arquivo você encontrará algumas configurações sobre o framework MVC Express do node.js. O que importa para gente são as configurações sobre o CORS!

Você encontrará alguns comentários nesse arquivo: **Descomente o código para habilitar os cabeçalhos da resposta:**



```
1  var express = require('express');
2  var consign = require('consign');
3  var bodyParser = require('body-parser');
4  var app = express();
5
6  app.use(express.static('../public'));
7
8  app.use(bodyParser.json());
9  app.use(bodyParser.urlencoded({extended: true}));
10
11
12 //cabeçalhos CORS
13 app.use(function(req, res, next) {
14   res.header("Access-Control-Allow-Origin", "http://localhost:3000");
15   res.header("Access-Control-Allow-Headers", "Origin, X-Requested-With, Content-Type, Accept");
16   next();
17 });
18
19
20 consign({cwd: 'app'})
21   .include('api')
22   .then('routes')
23   .into(app);
24
25 module.exports = app;
26
```

Por exemplo, a linha:

```
res.header("Access-Control-Allow-Origin", "http://localhost:3000");
```

faz que nosso navegador receba a permissão de para executar uma requisição cross-origin (CORS).

3) Reinicie o servidor (da pasta `alura-typer-servidor-cors/servidor`) na linha de comando:

```
C:\Users\CaelumWindows\Desktop\alura-typer-servidor-cors\servidor>npm start

> AluraTyper@1.0.0 start C:\Users\CaelumWindows\Desktop\alura-typer-servidor-cors\servidor
> node server

consign v0.1.2 Initialized in app
+ .\api\frases.js
+ .\api\placar.js
+ .\routes\frases.js
+ .\routes\placar.js
Banco data.db pronto para uso
Servidor iniciado
```

4) No navegador recarregue a sua aplicação:

<http://localhost:3000/principal.html> (<http://localhost:3000/principal.html>)

Tente carregar as frases aleatoriamente e também pela id. O erro do exercício anterior não deve mais aparecer, a requisição *cross-origin* deve funcionar!