

Servidor local

Transcrição

Como vimos, precisamos servir nossa aplicação através de um servidor web. Utilizaremos o [lite-server](https://github.com/JohnPapa/lite-server) (<https://github.com/JohnPapa/lite-server>). Além dele servir a pasta `alurabank/app` para nós, ele ainda suporta livereloading através do `BrowserSync` que traz embutido. Isso é perfeito, pois toda vez que os arquivos `.ts` forem modificados e os arquivos `.js` gerados nosso navegador automaticamente será recarregado.

Dentro da pasta `alurabank`, vamos instalar o `lite-server`:

```
npm install lite-server@2.3.0 --save-dev
```

Agora, em `alurabank/package.json` vamos adicionar a chamada do servidor através do script "server":

```
{
  "name": "alurabank",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "compile": "tsc",
    "start": "tsc -w",
    "server": "lite-server --baseDir=app"
  },
  "author": "",
  "license": "ISC",
  "devDependencies": {
    "@types/jquery": "^2.0.42",
    "lite-server": "^2.3.0",
    "typescript": "^2.3.2"
  }
}
```

Quando executamos no terminal `npm run server` nosso servidor subirá e podemos acessar nossa aplicação através de `localhost:3000`. No entanto, é necessário abrir outro terminal e dentro de `alurabank` executar o comando `npm start` para que o compilador em tempo real do TypeScript continue rodando.

No entanto, abrir dois terminais não é nada prática. Que tal se pudermos executar os dois processos paralelamente?

Rodando scripts paralelamente com o módulo `concurrently`

Vamos instalar o módulo `concurrently` (<https://www.npmjs.com/package/concurrently>). Ele nos permitirá rodar os dois scripts que criamos em paralelo nas plataformas Windows, MAC e Linux.

Dentro da pasta `alurabank` vamos executar o comando:

```
npm install concurrently@3.4.0 --save-dev
```

Agora, vamos renomear o script "start" para "watch" e adicionar novamente o script "start" que chamará o módulo concurrently:

```
{
  "name": "alurabank",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "compile": "tsc",
    "watch": "tsc -w",
    "server": "lite-server --baseDir=app",
    "start": "concurrently \"npm run watch\" \"npm run server\""
  },
  "author": "",
  "license": "ISC",
  "devDependencies": {
    "@types/jquery": "^2.0.42",
    "concurrently": "^3.4.0",
    "lite-server": "^2.3.0",
    "typescript": "^2.3.2"
  }
}
```

Excelente, agora, no terminal, basta executarmos o comando `npm start` para termos os dois serviços rodando em paralelo em um único terminal.

```
npm start
```