

 01

## Adicionando o app na Traybar

### Transcrição

Dando continuidade ao nosso curso, o que queremos fazer agora é permitir que o usuário consiga trocar de curso, já que no momento, só trabalhamos com um curso na aplicação. Para alterar o curso, devemos mexer na mão no HTML da página `index.html`.

Se o nosso objetivo é fazer uma aplicação integrada com o Desktop, aproveitando as funções do sistema operacional, a nossa integração atual não é nada prática. Então, vamos nos aproveitar de uma funcionalidade do sistema operacional para facilitar a troca de cursos.

Os ícones do sistema operacional servem para facilitar o acesso a algumas configurações do software. Por exemplo, através do ícone do wi-fi, conseguimos facilmente trocar a rede que estamos conectados.

### Adicionando um ícone na tray bar

Esses ícones são chamados de *tray icons*, e ficam na *tray bar*, que no macOS e no Ubuntu fica no canto superior direito, e no Windows no canto inferior direito.

Aproveitaremos a integração do Electron com os sistemas operacionais e colocaremos um ícone do **Alura Timer** na *tray bar*, e quando clicarmos nesse ícone, aparecerá os cursos já disponíveis, assim poderemos escolher um e trocar facilmente o curso.

Se formos consultar a [documentação \(`https://electron.atom.io/docs/`\)](https://electron.atom.io/docs/) do Electron, há um módulo **Tray** ([\(`https://electron.atom.io/docs/api/tray/`\)](https://electron.atom.io/docs/api/tray/), que nos permite criar ícones na *tray bar*. Ele pertence ao **\*Main Process**, logo devemos trabalhar com ele no processo principal, o nosso `main.js`.

Então vamos começar, primeiro importando o módulo e criando uma variável nula:

```
// main.js

const { app, BrowserWindow, ipcMain, Tray } = require('electron');

let tray = null;

// restante do código omitido
```

E agora, logo após a aplicação iniciar, nós instanciamos um `Tray`, que recebe o ícone que ficará na *tray bar* como parâmetro. Para isso, fazemos o download do zip ícone [aqui \(`https://s3.amazonaws.com/caelum-online-public/electron/cap05/imgs/download-icon-tray.zip`\)](https://s3.amazonaws.com/caelum-online-public/electron/cap05/imgs/download-icon-tray.zip), o extraímos dentro da pasta `alura-timer/app/img/` e implementamos o código:

```
// main.js

app.on('ready', () => {
  console.log('Aplicação iniciada');
  mainWindow = new BrowserWindow({
```

```
width: 600,  
height: 400,  
});  
tray = new Tray(__dirname + '/app/img/icon-tray.png');  
  
mainWindow.loadURL(`file://${__dirname}/app/index.html`);  
});
```

Agora, assim que a aplicação é iniciada, o ícone fica posicionado na *tray bar*. Assim que a aplicação é fechada, o ícone some.

Ótimo, conseguimos posicionar corretamente o ícone, vamos ver a partir do próximo vídeo o que fazer com ele.