



Certified Tech Developer

The Ultimate Degree

Front End III

Tips y recomendaciones sobre enrutamiento

Para repasar lo visto esta semana, te dejamos los conceptos importantes que debes dominar sobre el enrutamiento.

Importante

- Gracias a la potencia de los navegadores, a las bibliotecas y frameworks de front end (como React, Angular o Vue) y a tecnologías propias de JavaScript (como la API de Fetch y AJAX) podemos conectar el navegador con el back end sin necesidad de recargar la página. Esto dio paso a las single page applications (SPAs).
- Los frameworks modernos reintrodujeron el concepto de enrutador, pero ahora en el lado del front end.
- React Router utilizó el concepto tradicional de enrutamiento estático, pero a partir de la versión 5 introdujo el concepto de enrutamiento dinámico.
- Es muy importante entender que una URL no es en realidad una ruta. Es más bien una indicación de cómo se deben armar y conseguir los recursos. Esta indicación es interpretada por el enrutador de la aplicación.

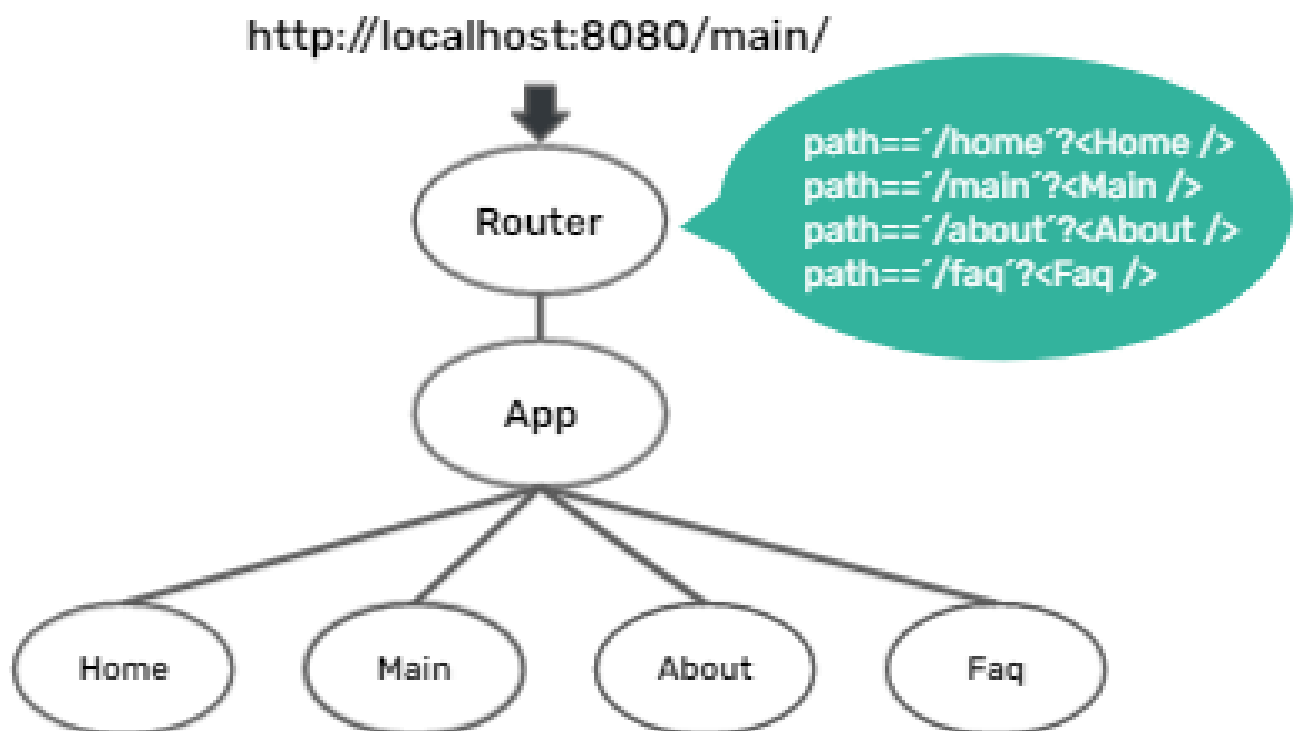
Enrutamiento estático

En el enrutamiento estático se deben definir de antemano todas las rutas en una ubicación centralizada. Luego, estas rutas están disponibles en el nivel más externo de la aplicación, antes de que suceda cualquier renderizado.

En el enrutamiento estático las rutas se definen juntas en un solo archivo, por ejemplo:

```
<Router history={browserHistory}>
  <Route path="/" component={Home}>
    <Route path="/main" component={Main}>
      <Route path="/main/article/" component={Article}/>
    </Route>
    <Route path="/faq" component={Faq}/>
    <Route path="/about" component={About}/>
  </Route>
</Router>
```

Flujo de una aplicación que utiliza enrutamiento estático:



Enrutamiento dinámico

Con el enrutamiento dinámico ya no necesitamos un archivo con rutas predeterminadas. Además, podemos usar componentes `Route` y `Link` según lo necesitemos en cualquier componente.

Con este enrutamiento, las rutas se pueden definir en el componente donde tenga sentido hacerlo. Por ejemplo:



```
<BrowserRouter>

  <Switch>

    <Route exact path="/"><Home /></Route>

    <Route path="/main"><Main /></Route>

    <Route path="/about"><About /></Route>

    <Route path="/faq"><Faq /></Route>

  </Switch>

</BrowserRouter>
```

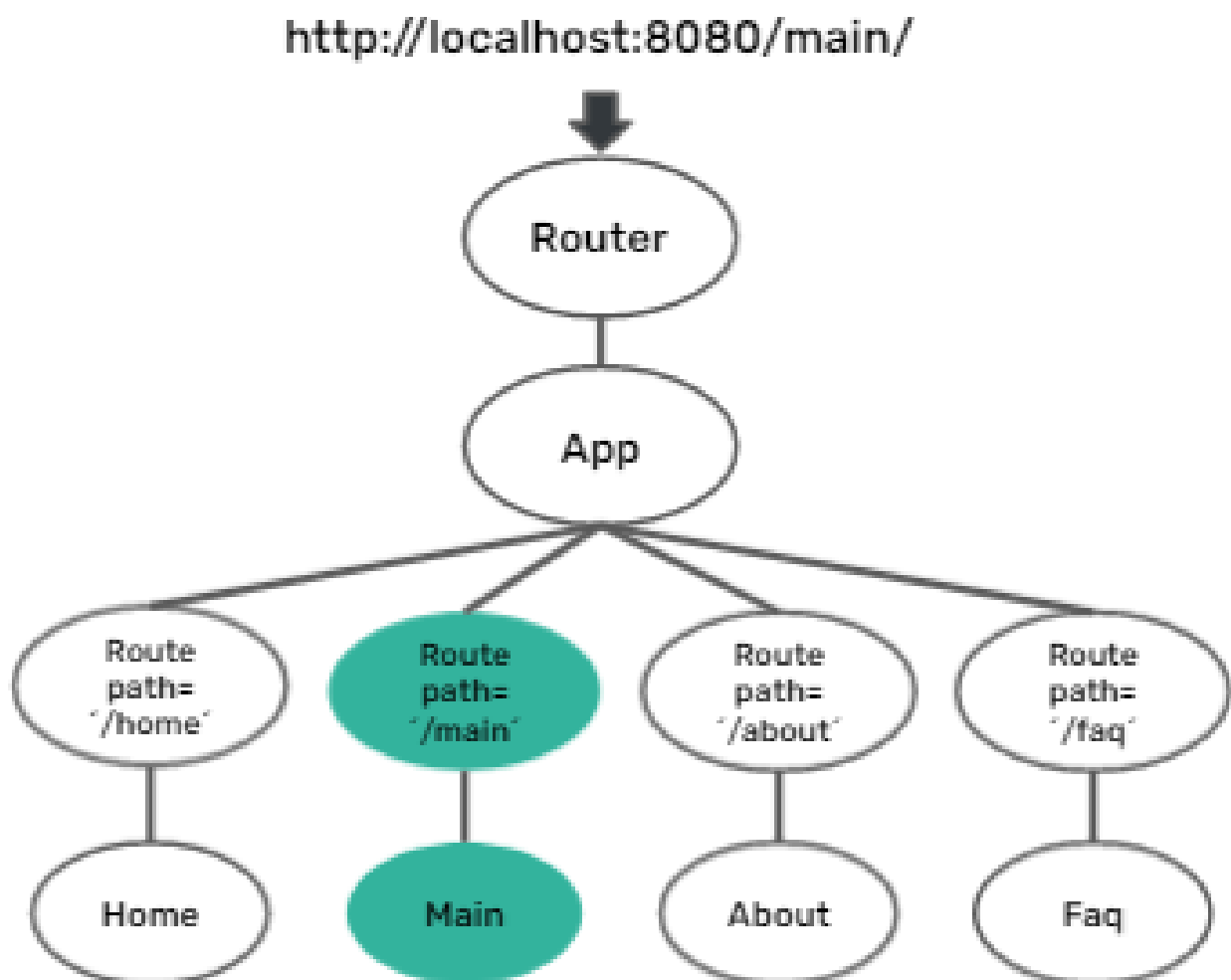
Y en otro componente, por ejemplo en el componente Main, también podemos definir rutas. Por ejemplo:

```
<Route path={'/main/blog'}> <Blog /> </Route>

<Route path={'/main/vlog'}> <Vlog /> </Route>
```



Flujo de una aplicación que utiliza enrutamiento dinámico:



Diferencias entre enrutamiento estático y dinámico

La siguiente tabla resume las diferencias clave entre ambos enfoques.

	Enrutamiento estático	Enrutamiento dinámico
Versión	Hasta la versión 3.	Desde la versión 4 en adelante.
Componentes de React Router.	Router, Route, IndexRoute.	BrowserRouter, HashRouter, Switch, Route, Link.
Rutas responsivas.	No.	Sí. Lo que se renderiza puede decidirse según las dimensiones de la pantalla.
Rutas semánticas y SEO-friendly.	Sí.	Sí.
Permite la separación clara de aspectos.	No, las rutas se ponen todas juntas en un archivo simplemente porque son rutas.	Sí, las rutas quedan agrupadas por aspectos del negocio, no por motivos de implementación técnica.