Curso NEXt.js

# Instalando NextJS

Requisitos

* + Node (>12,>10)
  + Git
  + Browser - Goggle Chrome - para utilizar la DevTools
* Instalacion
  + CLI - Automatica:
* yarn **create** **next**-app
  + Manual
* **mkdir** platzi-nextjs
* cd platzi-nextjs
* npm init -y
* yarn add **next** react react-dom
* npm add **next** react react-dom
* Agregar el siguiente codigo al "script"
* "scripts": {
* "dev": "next",
* "build": "next build",
* "start": "next start"

},

Inicializar NPM

npm init -y

Instalar con NPM

npm **install** **next** react react-dom

Scripts

"dev": "next",

"build": "next build",

"start": "next start"

Creamos directorio **pages**

mkdir pages

Iniciar con NPM

npm run dev

Routing

# Rutas básicas

<h4>Ideas/conceptos claves</h4>

**Rutas Estáticas:** Son las que son sencillas

**Rutas Dinámicas:** Son aquellas que se generan de una forma consecutiva dependiendo de variables

<h4>Apuntes</h4>

Todo lo que esta relacionado con paginas y navegación se entiende con el nombre de routing. En Next.js ya cuenta con una solución para no preocuparnos del tema. Cuenta con dos soluciones:

**Rutas Estáticas**

/about/

**Rutas Dinámicas**

/user/platzi

Si se quiere hacer rutas dinámicas con parámetros, nombramos el archivo de esta forma:

[parametro].js

para leer el parámetro se puede usar el hook de Next.js

**import** { useRouter } **from** "next/router";

**const** ProductItem = () => {

**const** {

query: { productId },

} = useRouter();

**return** <div>Esta es la pagina del producto: {productId}</div>;

};

**export** **default** ProductItem;

Los cambios que ejecutamos vienen con Hot Reload y fast refresh. Es decir que se aplican rápidamente en en el navegador.

Cada vez que creamos un archivo en pages se agregan a las rutas de la aplicación. Esta manera de hacer rutas se denomina routing basado en file system

**RESUMEN:** Crear rutas en Next.js es bastante sencillo debido a que usa routing basado en file system

