**Nuxt.js**

Nuxt.js es un framework que está basado en vue.js y escrito en JavaScript es Open Source (Código Abierto) facilita la creación pequeños medianos y grandes proyectos.

**Explicación presentación**

**Diapositiva 2:** Concepto de nuxt.js

**Diapositiva 3:** Para crear un proyecto en nuxt.js abrimos una ventana de comandos de Windows o una terminal de Linux y escribimos:

**npx create-nuxt app <Project-name>**  en nuestro caso el proyecto se llama connectapi

Mientras se genera el proyecto se nos pide que elijamos algunas opciones como lenguaje de programación en nuestro caso JavaScript, en gestor de paquetes elegimos npm, en el framework ui escogimos Bootstrap-vue porque es muy fácil de utilizar y nos brinda una cantidad de componentes que son muy útiles como cards, botones, alerts, navbar,etc.

**Diapositiva 4:** Una vez generado el proyecto en la terminal se pueden observar los comandos para ejecutarlo en modo desarrollo, con **npm run dev**

**Diapositiva 5:** En el panel lateral izquierdo podemos observar cada una de las carpetas del proyecto, del lado derecho tenemos el package.json con el script correspondiente para ejecutar el proyecto, también podemos observar la terminal indicándonos el puerto en el que va a correr el proyecto en este en **http://localhost:3000**

**Diapositiva 6:** Al escribir en la barra de direcciones de nuestro navegador favorito **http://localhost:3000** se pinta la información contenida en el archivo principal del proyecto, el siguiente paso es editarlo a nuestra conveniencia.

**Diapositiva 7:** En la parte izquierda tenemos dos carpetas una llamada componentes y una llamada pages que tiene las vistas y rutas de nuestro proyecto en ese caso User.vue y User2.vue

**Diapositiva 8: EN EL ARCHIVO USER.VUE DENTRO DE LA ETIQUETA TEMPLATE MEDIANTE EL COMPONENTE NUXT-LINK HACEMOS REFERENCIA AL HOME O RUTA PRINCIPAL DEL PROYECTO UTILIZAMOS LA CLASE BTN BTN-SUCCESS DE BOOTSTRAP PARA QUE SE NOS CREE UN BOTON PARA QUE AL PULSARLO NOS LLEVE AL HOME.**

**DIAPOSITIVA 9:** Dentro de tamplate creamos una rejilla básica con Bootstrap y dentro de una card (TARJETA ) pintamos la información que extraemos de la api randomuser. Mediante un ciclo v-for recorremos todos los datos que se encuentran en el arreglo users y los pintamos en la etiqueta h5 de html mediante la notación de puntos.

**Diapositiva 10:** Dentro del archivo user pero en la sección de script importamos axios, también le asignamos un nombre al componente en este caso User, dentro de data creamos un arreglo llamado users. Dentro de mounted hacemos el llamado a la api mediante axios y la información obtenida la almacenamos en el arreglo users.

**DIAPOSITIVA 11 Y 12:** Hacemos lo mismo que en la diapositiva 9 la única diferencia es que nos conectamos a una api llamada reqres.in

**Diapositiva 13:** EN LA CARPETA COMPONENTE EDITAMOS EL ARCHIVO TUTORIAL.VUE, AÑADIMOS UNA IMAGEN Y UNA DESCRIPCIÓN CORTA DEL PROYECTO ADATAR.

**DIAPOSITIVA 14:** EN EL ARCHIVO INDEX.VUE QUE SE ENCUENTRA DENTRO DE LA CARPETA PAGES IMPORTAMOS EL COMPONENTE TUTORIAL.VUE QUE CONTIENE LA INFORMACIÓN QUE SE VA A MOSTRAR EN LA VISTA PRINCIPAL. TAMBIÉN IMPORTAMOS EL COMPONENTE NAVBAR.VUE.

**DIAPOSITIVA 15:** AL EJECUTAR EL PROYECTO CON EL COMANDO NPM RUN DEV, PODEMOS VISUALIZAR EL HOME O PANTALLA PRINCIPAL

**DIAPOSITIVA 16:** VISTA USER, DONDE SE PINTA LA INFORMACIÓN QUE ESTAMOS EXTRAYENDO DE LA API RANDOM USER