LABORATORIO DE  
**Instalación y Configuración**

https://s3.amazonaws.com/nextu-content-production/Desarrollador_Web/08_Herramientas_Build_Systems_Tecnologias_Emergentes/includes/images/clock.png

45 MINUTOS

La empresa de conciertos **Luz Azul** cuenta con un sitio web estático donde ponen la información relevante a los instrumentos que utilizan y, en el futuro, pondrán la información de los conciertos realizados. El director de la orquesta quiere empezar a promocionar los conciertos de manera internacional y ha encomendado al director de tecnología que se implemente una estrategia de internacionalización para la página. También continuamente el equipo de fotografía e imagen cambia los recursos que se muestran en la página web y esto implica esfuerzos por parte del equipo de tecnología de la empresa.  
El director de tecnología te ha pedido que, como desarrollador web, crees una estrategia de internacionalización en la cual se produzca el sitio web de **Luz Azul** en cada uno de los idiomas que se planean introducir. Además ha resaltado otros aspectos: debes crear una manera en la que los recursos sean definidos por el equipo de fotografía e imagen y se integren al sitio web de manera automática.  
En una revisión de código del sitio actual se encontró que usa foundation, jquery y what-input.js como dependencias de código javascript / css. Debes instalar estas dependencias en el sitio y la infraestructura de tareas que diseñamos.

Contenido:

Para el desarrollo de este laboratorio debes descargar el **código base** que contiene la página estática de Conciertos **Luz Azul.**

Contenido

A continuación te presentamos el archivo base:

[Descargar archivo base](https://s3.amazonaws.com/nextu-content-production/Desarrollador_Web/08_Herramientas_Build_Systems_Tecnologias_Emergentes/Ejercicios/WEB16S_C8U1L1_Laboratorio1/solucion/codBase.zip)

Instrucciones

* 1. **Estructura del proyecto:**  
       
     Descarga el código base e inspecciónalo.  
     Indica la estructura del proyecto y cuál es la función de cada carpeta. Luego describe la estructura del archivo index.html, indicando donde se incluyen las hojas de estilo, los scripts de Javascript y las imágenes que se muestran en la página.  
       
     El proyecto consiste de un archivo index.html donde está el código html del proyecto. Adicionalmente tenemos 3 carpetas css, js y img.  
     + **Carpeta js:** Contiene los archivos Javascript del proyecto, adicionalmente contiene una carpeta vendor con las dependencias en Javascript.
     + **Carpeta img:** Se encuentran todas las imágenes que provee el equipo gráfico
     + **Carpeta css:** Contiene las dependencias de css de la librería Foundation y el css de la aplicación en app.css

* 1. **Manejo de scripts:**  
       
     Diseña las tareas que se deben ejecutar para los manejos de scripts de las librerías usadas.  
     + Los scripts deben ser descargados por un **manejador de paquetes** que permite, a su vez, descargar las dependencias de manera local.
     + Se debe mantener un archivo para saber qué librerías y en qué versión deben estar instaladas. **(Package.json)**
     + Se debe hacer una **tarea** que revise qué archivos son dependencias de la aplicación y los contenga en una carpeta haciendo **copias de los archivos** para la versión que se va a construir.
     + Estos archivos .js pueden juntarse en un solo archivo **vendor-bundle.js,** por ejemplo, utilizando una **tarea que los concatene.**
     + Los scripts de la aplicación también pueden juntarse con **vendor-bundle.js** en **app.js** para servir un sólo archivo Javascript con otra **tarea de concatenación.**
     + El nombre del archivo a importar debe ser inyectado en el **index.html** por una tarea de **reemplazamiento.**

* 1. **Manejo de recursos:**  
     Diseña las tareas que implementan una infraestructura para el manejo de los recursos gráficos, creados por el equipo de fotografía.  
     Para eso, realiza una **tarea de reemplazamiento,** tomando las llaves desde el archivo json y las reemplaza por las correctas en todos los html.
  2. **Estrategia de localización del sitio web:**  
     Diseña la infraestructura de tareas requerida para generar las versiones localizadas del sitio web. Debes tener archivos html esqueleto que sean los que se tomarán de base, utilizando un diccionario de etiquetas reemplazar en cada bundle para cada sitio web estático generado. Index.html finalmente detectara el lenguaje a cargar y luego inyecta el html necesario según el lenguaje.
     + Carpeta con plantillas html
     + Carpeta con diccionarios en JSON
     + Construcción del bundle html para cada lenguaje con tareas de reemplazamiento y luego copia.
     + Meta Plantilla de index.html que reemplaza el index.html y mediante javascript detecta qué lenguaje debe ser cargado para luego, cargarlo.

Buenas Prácticas

* Recuerda que puedes guardar varias versiones de archivos intermedios a medida que tu cadena de procesamiento de tareas se va completando. Usualmente es mejor servir al final la menor cantidad de archivos posibles para reducir la cantidad peticiones.