

Recuerda:

- ¡Los videos de YouTube pueden ser acelerados para ahorrar tiempo de visualización!
- Algunos videos en inglés tienen subtítulos incorporados.
- Proyecto Marketplace:
https://github.com/agustinruatta/marketplace_tecnicatura_2023/
- Trabajo final:
<https://docs.google.com/document/d/1Bd7xIIH7FiRX-j5nOczL6iRKWBOXWQG64ah9F2J3et4/>

Clase 1

- Fecha: 08/03/2023
- Temario: introducción a la materia, a internet y a la WWW.

Recursos: Español

¿Cómo funciona internet? explicación breve y simplificada de:

- Medio principal usado por internet
- IP
- ¿Qué es un router?
- Estructura y topología de internet
- DNS
- Protocolos
- ICANN

Más sobre Internet

¿Cómo funciona la web?

Recursos: Inglés (recomendada)

¿Cómo funciona internet?: explicación completa de cómo funciona internet:

- IP
- ¿Qué es un switch?
- ¿Qué es un router?
- Estructura y topología de internet
- WAN
- ISP

¿Cómo funciona la web?: explicación de qué es la web y cómo funciona:

- DNS
- Servidor
- HTTP
- HTML
- CSS
- Javascript
- Diferencia de frontend y backend. Tecnologías usadas en ambas.

Ejercicios

- Investiga el comando ping. Usandolo haz una consulta a la IP 8.8.8.8 . ¿Cuánto tarda cada paquete? ¿Cuántos han llegado?

- Investiga el comando [comando tracert \(Windows\)](#) o [traceroute \(GNU/Linux\)](#) y aplícalo sobre la IP 8.8.8.8. ¿Qué camino ves? También puede usar la [versión online](#).
- Apretando F12, abre la pestaña network del navegador e ingresa a <https://www.wikipedia.org/>. ¿Qué requests aparecen? ¿Cuales son los headers de petición y respuesta? ¿Cuál es el cuerpo que devuelve?

Clase 2

- Fecha: 15/03/2023
- Temario: HTML

Recursos

- ¡Elige e instala tu IDE! Puedes usar el que prefieras. **Lo importante es que lo tengas instalado y listo para usar para ésta clase.**
- [Aprende HTML](#): ver desde el principio hasta el minuto 2:09:22 (todo lo referido a HTML).

Recursos complementarios (opcionales):

- https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/HTML_basics
- https://developer.mozilla.org/es/docs/Learn/Forms/Your_first_form
- <https://web.dev/learn/html/>

Ejercicios autoevaluación

Crea un HTML con tu currículum vitae. El mismo debe contener los siguientes datos:

- Tu nombre y apellido
- Email de contacto
- Teléfono
- Una imagen tuya
- Experiencia laboral en diferentes puestos, y un link a la página de esas empresas
- Estudios académicos
- Idiomas hablados
- Proyectos personales
- Intereses o hobbies

Puedes ver [un ejemplo de CV aquí](#). Aunque este CV se muestra con un cierto estilo, céntrate solamente en generar el HTML (no importa que se vea mal o feo).

Video de Resolución:

<https://youtu.be/mQThm444ydl>

Proyecto

Introducción

Durante la materia trabajaremos sobre un proyecto en donde aplicaremos lo aprendido. Antes de comenzar, unas cosas a tener en cuenta:

1. Aunque se detallarán las cosas a hacer, hay muchas cosas sin definir. Y esto es buscado. Usa tu imaginación para decidir lo mejor, crea lo que te parezca mejor.
2. Siempre se dividirá en 2 partes:

- a. Obligatorio: a esto deberás tenerlo hecho obligatoriamente, ya que sino no podrás avanzar en la próxima clase.
- b. Opcional: si eres muy rápido y terminas todo lo obligatorio, puedes hacer la parte opcional (más tareas), ¡o agrégale lo que tu quieras!

El proyecto consta de un marketplace sobre productos tecnológicos (como un Mercado Libre simplificado de computadoras y similares). Tendrá varias vistas (o páginas), las cuáles serán:

- La página principal (o home).
- La vista del producto, que contendrá toda la información.
- Administración de los productos a vender (creación, edición, eliminación y lectura).
- Buscador

Descripción de la tarea

Obligatorio

Lo primero a hacer es la página del producto. En la misma el usuario podrá ver toda la información del mismo (por ejemplo, de una Notebook). Mínimamente deberá tener los siguientes campos:

1. El título indicando el producto (por ejemplo, "Notebook Dell Inspiron 3505").
2. Una foto del producto.
3. Una descripción del producto (ej: indicando su procesador, tamaño de pantalla, etc).
4. Link con el sitio web del fabricante (por ejemplo, <https://www.dell.com>)
5. El precio.
6. Extra: si te sobra tiempo, añádele una sección con los comentarios de personas que ya compraron este producto, y una valuación del 1 al 5. Debajo de eso, deberá haber un formulario para ingresar una calificación.

¡Recuerda! Si quieres agregar o modificar algo, ¡siéntete libre!

Opcional

¿Terminaste los puntos anteriores? Entonces empieza a trabajar en la página principal. La misma deberá tener al menos la siguiente información:

1. Un input para poder ingresar un producto a buscar
2. Una grilla de algunos productos que están a la venta. Al hacer click sobre cada producto, debe llevar a la página del producto (explicada en el punto anterior)

Clase 3

- Fecha: 22/03/2023
- Temario: CSS

Recursos

Aprende CSS: de forma obligatoria deberás ver desde el minuto 2:09:22 hasta el minuto 4:12:50. Sin embargo, te recomiendo que lo veas hasta el final.

Grid Layout: recomendado para tener una mayor facilidad en cómo ordenar tus elementos.

Recursos complementarios (opcionales):

- https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/CS_S_basics
- <https://web.dev/i18n/es/learn/css/>
- <https://css-tricks.com/snippets/css/complete-guide-grid/>
- <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>
- <https://cssgridgarden.com/> (Minijuego para practicar CSS grid)
- <https://flexboxfroggy.com/> (Minijuego para practicar CSS flexbox)

Ejercicios autoevaluación

Añade estilos con CSS al CV creado la clase pasada. Recuerda que puedes ver [un ejemplo de CV aquí](#) si necesitas inspiración.

Video de Resolución:

<https://youtu.be/840HY-aJpiY>

Proyecto

Agrega CSS a tu marketplace. Al finalizar la clase, haremos una competencia para elegir el mejor.

Clase 4

- Fecha: 29/03/2023
- Temario: Git y GitHub

Recursos

- [Introducción a Git a GitHub](#): ver vídeo completo
 - Opcional - [Configurar GitHub con claves SSH](#): la mejor forma, y la más segura, es conectarse a GitHub a través de claves SSH. Por lo que te recomiendo que lo configures en tu máquina
 - Opcional: <https://git-scm.com/book/es/v2>
 - Opcional: <https://www.youtube.com/watch?v=jSJ8xhKtfP4&list=PLTd5ehIj0goMCnj6V5NdzSIHBgrIXckGU>

IMPORTANTE: crea una cuenta en GitHub. La usaremos en la próxima clase.

Ejercicios autoevaluación

Crea un repositorio Git y añade el CV que has creado. Posteriormente súbelo a GitHub.

Video de Resolución:

<https://youtu.be/BmgrhBuxEqE>

Proyecto

Git

1. Aplica Git al proyecto del marketplace
2. Súbelo a un repositorio de GitHub.
3. Crea una nueva rama en donde añadirás un cambio (por ejemplo, un agregar un nuevo producto a la página principal, o usar CSS Grid Layout para darle orden).
4. Pushear esa rama y crear un PR (Pull Request).
5. Asignar ese PR a un compañero.
6. El compañero deberá hacer el review, indicando los comentarios si es necesario.
7. En caso de que el PR esté aprobado, mergearlo a master.
8. Deployar el marketplace usando GitHub pages

CSS



Si todavía no lo has hecho, aplica CSS Grid Layout al marketplace.

Clase 5

- Fecha: 05/04/2023
- Temario: Javascript

Recursos

- Opcional: Antes de comenzar, te recomiendo que leas [este artículo](#) en el que te dará consejos para aprender Javascript de forma más rápida y correcta.

- Obligatorio: Deberás ver el video  **Curso Completo de JAVASCRIPT para Principiantes (2021)**  . Debes ver hasta el minuto 02:33:31. Además, agrega la sección desde 3:30:14 hasta 3:36:30.
- Opcional: En YouTube [existe un curso de Javascript](#) (y muchas cosas relacionadas) que es excelente (un curso así valdría 100 dólares en Udemy). La única desventaja es que dura 19 horas. Si tienes tiempo de sobra, lo recomiendo para que lo vean. **No es obligatorio para la materia.**
- Opcional: en el sitio es.javascript.info obtendrás mucha información de forma escrita. Puedes usarlo como reemplazo del video obligatorio, o como un apoyo.
- Opcional: https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/JavaScript_basics

Ejercicios autoevaluación

Calculadora

Crea una calculadora. Esta se compone de dos *inputs* en donde podrás ingresar los números, un *input del tipo radio* donde seleccionarás la operación a realizar, y un botón “Calcular”, que al hacerle click mostrará el resultado usando un *alert*.

Video de Resolución:

<https://youtu.be/x7F1cLMQNMU>

Proyecto


Descubre el asesino en <https://scavenger-hunt-three.vercel.app/>

Clase 6

- Fecha: 12/04/2023
- Temario: Javascript

Recursos opcionales

Para esta semana no debes ver ningún material, ya que se usa para que todos se pongan al día. En caso que quieras adelantar para la próxima clase, mira la parte restante de

 **Curso Completo de JAVASCRIPT para Principiantes (2021)**  .

Ejercicios autoevaluación

Agenda de tareas

Crea una lista de tareas. Esta se compone de un *input* en donde podrás ingresar el nombre de una tarea, un *textarea* donde ingresarás la descripción, y un botón “Guardar”, que al hacerle click agregará la tarea a una lista. Un ejemplo muy sencillo de cómo se vería, sería:

Agenda de tareas

Nombre: Descripción:

Tareas

- Ir al supermercado: Comprar X, Y, Z
- Estudiar para lab de computación 3: el profesor es muy ortiva, mejor estudiarlo

Sin embargo, puedes darle la forma que quieras. Agregar CSS para una mejor presentación. Usar Grid Layout.

Mejora a la lista de tareas

Si terminaste lo anterior y querés practicar más, haz lo siguiente:

- Además del nombre y la descripción, agrega que puedan seleccionar el estado de la tarea (Pendiente, En Progreso, Finalizada). En la lista de las tareas, dividirla en 3 secciones y, dependiendo de qué estado selecciona el usuario, agregarlo a esa sección.
- Permite editar las tareas que aparecen en las listas.

Proyecto

Obligatorio

En la página de producto, ya agregaron una sección de feedback. Ahora agrega la característica para que el usuario pueda ingresar un feedback y que, presione el botón “Enviar”, el feedback se muestre en la página.

Opcional

1. En la página principal estamos mostrando la previsualización de una lista de productos. Sin embargo, tenemos que modificar el HTML por cada nuevo producto que queramos agregar. Modifica la página para que puedas guardar un array de objetos productos, y que “renderice” ese array de productos en la página. Entonces, cada vez que necesitemos agregar un producto, en vez de modificar el HTML, añadiremos un nuevo objeto en ese array.
2. En la parte superior de la página principal hay un buscador de productos. Añade la posibilidad de que si el usuario ingresa un término de búsqueda y presiona en el botón “Buscar”, solamente se muestre en esa lista los productos que contengan ese término. Si borra todo el término y presiona buscar, vuelve a mostrar todos los productos.

Opcional

En la [presentación usada para las clases](#) podrá encontrar diversos ejercicios para practicar javascript dentro del navegador.

Clase 7

- Fecha: 19/04/2023
- Temario: Javascript

Recursos opcionales

Para esta semana no debes ver ningún material, ya que se usa para que todos se pongan al día. En caso que quieras adelantar para la próxima clase, mira la parte restante de

 [Curso Completo de JAVASCRIPT para Principiantes \(2021\)](#) .

Ejercicios autoevaluación

En caso que no lo hayas hecho, resuelve el ejercicio de autoevaluación de la “Agenda de Tareas” de la clase pasada.

Proyecto

Ejercicio: Conversor de unidades

Crea una página web que permita a los usuarios convertir unidades de medida. La página deberá tener dos campos de entrada: uno para ingresar la cantidad a convertir y otro para seleccionar la unidad de medida inicial. Además, habrá un tercer campo para seleccionar la unidad de medida final y un botón "Convertir".

Una vez que el usuario ingrese la cantidad y seleccione las unidades de medida, al hacer clic en el botón "Convertir", la página deberá mostrar el resultado de la conversión en una sección dedicada para ello.

Para este ejercicio, se deben incluir las siguientes conversiones:

- Convertir grados Celsius a Fahrenheit
- Convertir metros a pies
- Convertir kilómetros a millas
- Convertir libras a kilogramos

Consejos:

- Antes de comenzar a programar, piensa en cómo vas a estructurar tu código y planifica los pasos que debes seguir para realizar la conversión.
- Para la conversión de Celsius a Fahrenheit, puedes utilizar la siguiente fórmula: $f = c * 9/5 + 32$, donde c es la temperatura en grados Celsius y f es la temperatura en grados Fahrenheit.
- Para la conversión de metros a pies, puedes utilizar la siguiente fórmula: $ft = m * 3.2808$, donde m es la longitud en metros y ft es la longitud en pies.
- Para la conversión de kilómetros a millas, puedes utilizar la siguiente fórmula: $mi = km * 0.62137$, donde km es la longitud en kilómetros y mi es la longitud en millas.
- Para la conversión de libras a kilogramos, puedes utilizar la siguiente fórmula: $kg = lb / 2.2046$, donde lb es el peso en libras y kg es el peso en kilogramos.

Ejercicio: Generador de contraseñas

Crea una página web que permita a los usuarios generar contraseñas aleatorias y seguras. La página deberá tener un campo de entrada para especificar la longitud de la contraseña y un botón "Generar" para crear la contraseña aleatoria.

Al hacer clic en el botón "Generar", la página deberá mostrar la contraseña generada en una sección dedicada para ello. La contraseña debe ser aleatoria y debe cumplir con los siguientes criterios de seguridad:

- Tener al menos una letra mayúscula y una letra minúscula
- Tener al menos un número
- Tener al menos un carácter especial, como !, @, #, \$, %, &, *, (,), -, _, +, =, [,], {, }, |, \, /, ?, <, > y ..

Consejos:

- Antes de comenzar a programar, piensa en cómo vas a generar la contraseña aleatoria y planifica los pasos que debes seguir para cumplir con los criterios de seguridad.
- Para generar letras mayúsculas y minúsculas aleatorias, puedes utilizar la función `Math.random()` para generar un número entre 0 y 1 y la función `String.fromCharCode()` para convertir el número en un carácter ASCII correspondiente a una letra mayúscula o minúscula.

Otros ejercicios

En caso que no lo hayas hecho, podrás realizar:

- El ejercicio de la calculadora.
- El ejercicio de la agenda de tareas.

Clase 8

- Fecha: 26/04/2023
- Temario: API, Fetch, promises y async en Javascript. POO en Javascript.

Recursos

- **Obligatorio:** Si no lo has hecho todavía, mira el video [Curso Completo de JAVASCRIPT para Principiantes \(2021\)](#) 🚀 hasta el minuto 05:03:40.
- **Opcional:** Teoría de la programación orientada a objetos. [¿Qué es la programación orientada a objetos?](#) . Recomendado para entender el trasfondo.
- **Obligatorio:** POO en Javascript [JavaScript Orientado a Objetos \(POO\)](#) 💪 Curso de JAVASCRIPT 2018 💪 .
NOTA: en el video menciona que las clases todavía no están implementadas en algunos navegadores (ya que el video es del 2018). Al día de hoy, todos los navegadores lo implementaron.
- **Opcional:** [Aprende POO realizando un Proyecto con Javascript](#) ⚡ podrás ver POO en Javascript con un ejemplo más completo, aunque es similar al video anterior.
- **Opcional:** "Soy Dalto" explicando las bases de POO: [PROGRAMACION ORIENTADA A OBJETOS EXPLICADA EN 10 MINUTOS](#) .
Nota: le faltan ciertos conceptos que el primer video lo tiene.

Ejercicios autoevaluación

API: muestra la información de tu usuario de Codewars

Crea un HTML, en donde puedas ingresar a un input tu usuario de codewars, y un botón "Mostrar honor". Al apretar dicho botón, deberá consultar a la API de codewars, y mostrar el honor que posee.

Por ejemplo, supongamos que ingresó "agustinruatta" en el input y presiono el botón. Eso hará una consulta a <https://www.codewars.com/api/v1/users/agustinruatta>, el cuál devolverá el siguiente JSON (se muestra una parte):

```
{
  "username": "agustinruatta",
  "name": null,
  "honor": 320,
  "clan": null,
  "leaderboardPosition": 139717,
  "skills": null,
```

Como podrás ver, existe un atributo "honor", el cual contiene dicho honor. Mostrar el dato insertándolo en un HTML.

Objetos: Receta de cocina

Cree un objeto para guardar la información de tu receta favorita. Debe tener como propiedades:

- titulo: string
- porciones: number
- ingredientes: array de strings

En líneas separadas (o usando un console.log para cada uno), mostrar la información de la receta para que muestre algo como:

- Pizza
- 8 porciones
- Ingredientes: harina, queso, jamón.

[Solución del ejercicio.](#)

Objetos: Lista de lectura

¡Lleva un registro de los libros que leíste y los que quieres leer!

Crea un array de objetos, donde cada objeto describa un libro y tenga de propiedades un título (string), autor (string), yaLeído (boolean, que indica si ya lo has leído).

Itera a través del array de los libros. Por cada libro, muestra (usando console.log o poniéndolo en HTML) el título del libro y el autor al estilo "The Hobbit, escrito por J.R.R. Tolkien"

Ahora usa una instrucción if/else para mostrar una salida diferente dependiendo si ya lo has leído o no. Si ya lo has leído, deberías mostrar algo como “Ya has leído ‘The Hobbit’, escrito por J.R.R.”. Si no lo has leído, deberías mostrar “Todavía no has leído ‘The Hobbit’, escrito por J.R.R.”

[Solución del ejercicio.](#)

Objetos: más ejercicios

Si deseas hacer más ejercicios, visita la siguientes páginas:

- https://www.learn-js.org/en/Object_Oriented_JavaScript
- <https://lizthedeveloper.github.io/javascript-for-python-developers/exercises/objects.html>

Proyecto

¡Vamos a interactuar con una API en nuestro marketplace! Recuerda subir todo a tu repositorio de GitHub.

Obligatorio

En la pantalla principal, donde se encuentra toda la lista de productos, agregarás un nuevo campo arriba en donde se indicará la cantidad de notebooks que actualmente están en ventas. ¿Cómo obtendrás ese dato?

Si accedes a

https://my-json-server.typicode.com/agustinruatta/fake_json_server_db/statistics podrás ver que retorna un JSON, el cual tiene un campo amount_of_products, que indica la cantidad de productos en oferta. Es por eso que, cuando cargues la página, deberás hacer una consulta GET a dicha URL, obtener la cantidad total de productos, y mostrarla en tu HTML.

Opcional 1

En la pantalla del producto estamos mostrando la información del mismo, aunque es un HTML fijo. ¡Usemos una API!

Si ingresas a

https://my-json-server.typicode.com/agustinruatta/fake_json_server_db/products/1, notarás que devuelve un JSON con toda la información del producto. Modifica la pantalla del producto para que cada vez que se cargue, llame a dicha URL y use la información devuelta para mostrar los datos.

Opcional 2

En la pantalla principal estás mostrando actualmente una lista de todos los productos que están a la venta. Pero dichos productos están puestos en el HTML de forma “hardcodeada” (es decir, escrito a mano en el HTML). Vamos a reemplazarlo con los que nos devuelve la API.

Si accedes a

https://my-json-server.typicode.com/agustinruatta/fake_json_server_db/products observarás que devuelve un JSON que contiene un array de productos. Cada vez que cargues la

página HTML, haz una llamada HTTP GET a dicha URL, e imprime dentro del HTML los productos que devuelve.

Clase 9

- Fecha: 10/05/2023
- Temario: Vue.js

Recursos

Obligatorio

Terminar de ver los materiales de la clase 8.

Opcional

Si ya estás seguro con Javascript, podés adelantar el material que veremos más adelante.

Son 3 videos cortos de Vue.js:

- [Lección 1 - Introducción.](#)
- [Lección 2 - Creando nuestra app Vue.](#)
- [Lección 3 - Attribute Binding.](#)

Ejercicios autoevaluación

Obligatorio

Realizar los ejercicios planteados en la clase 8.

Opcional

Al finalizar cada video de Vue.js tendrás ejercicios de autoevaluación. **Recomiendo que los hagan.**

Proyecto

Obligatorio

Realizar los puntos dados para la clase anterior.

Opcional

En caso que hayas implementado todos los puntos anteriores, deberás realizar una página para mostrar información acerca de los planetas. Consultando la API

<https://handlers.education.launchcode.org/static/planets.json>, muestra los diferentes datos, como la distancia, imagen, etc.

Clase 10

- Fecha: 17/05/2023
- Temario: Vue.js

Recursos

Esta semana arrancaremos con Vue.js. Son 3 videos muy cortos.

- [Lección 1 - Introducción.](#)
- [Lección 2 - Creando nuestra app Vue.](#)
- [Lección 3 - Attribute Binding.](#)
- [Lección 4 - Renderizado condicional.](#)

Código en CodePen: <https://codepen.io/VueMastery/project/editor/AYQBOW#0>

Repositorio GitHub: <https://github.com/Code-Pop/Intro-to-Vue-3>

IMPORTANTE: Si buscan recursos por su cuenta (videos, texto, etc), **es importante que verifiquen que es para Vue 3**. En internet hay mucho material, pero para Vue 2 (una versión anterior). De hecho, si buscan en Google la documentación de Vue, uno de los primeros links que sale es el de Vue 2. Hay algunas diferencias entre Vue 2 y 3, y les va a pasar que les falla algo sin saber por qué.

Ejercicios autoevaluación

Al finalizar cada video tendrás ejercicios de autoevaluación. **Recomiendo que los hagan.**

Proyecto

Proyecto 1

Modifica el marketplace que has hecho para usar Vue.js. Recuerda modificar todo lo que en un futuro pueda ser cambiado por los datos dependiendo del contexto. Por ejemplo, el título de un producto es un claro candidato.

Proyecto 2

Haz tu CV usando Vue.js.

Clase 11

- Fecha: 24/05/2023
- Temario: Vue.js

Recursos

- [Lección 5 - Renderizado de listas.](#)
- [Lección 6 - Manejo de eventos.](#)
- [Lección 7 - Class y Style Binding.](#)
- [Lección 8 - Computed Properties.](#)

Ejercicios autoevaluación

Al finalizar cada video tendrás ejercicios de autoevaluación. **Recomiendo que los hagan.**

Proyecto

- **Obligatorio:** Agrega un botón en la página del producto, donde diga “Comprar”. Eso lo añadirá a tu carro de compras. En la primera iteración, sólo mostrar la cantidad de compras hechas. No importa si se pierde lo comprado al refrescar la página (ya veremos cómo guardarlo más adelante).
- **Obligatorio:** La misma notebook puede tener diferentes fotos. Colocar al lado de la imagen principal, las miniaturas de las diferentes tomas de las computadoras. Al posicionarse sobre esa miniatura, deberá cambiar la imagen principal.
- **Obligatorio:** usa computed properties en todos los lugares donde sea conveniente. Queda a criterio del estudiante elegir esto.
- **Opcional:** cada computadora puede tener diferente cantidad de RAM. Por ejemplo, una puede tener 8 GB de RAM, mientras que la otra tiene 16. Obviamente los precios son diferentes. Mostrar una lista de las diversas opciones de RAM en cada

computadora, y al presionar sobre cada opción, cambiar el precio final que se muestra.



- Opcional: como mencionamos antes, cada variación tiene un costo diferente. Por ende, vamos a mejorar el carro que tenemos, y además indicaremos el precio de cada una. Por ejemplo, si compro (es decir, presiono el botón de “Comprar”) una computadora de 8 GB (cuyo costo es \$100.000), y otra de 16 GB (precio: \$125.000), mostrar en el carro ambas computadoras acompañadas de su precio junto con el total. En este caso sería:
 - Computadora X: \$100.000
 - Computadora Y: \$125.000
 - Total: \$225.000

Si logras hacer los 5 puntos, ¡eres un maestro/maestra en Vue.js!”

Clase 12

- Fecha: 31/05/2023
- Temario: Vue.js

Recursos

- [Lección 9 - Componentes](#).
- [Lección 10 - Comunicando eventos](#).
- [Lección 11 - Formularios](#).
-   Instalar Node.js, NPM y YARN (en Windows10) : en caso que no tengas instalado ni NPM ni Yarn, hazlo. No es obligatorio que instales “Yarn” , aunque lo recomiendo.
- [Import y export en JavaScript](#): debido a que en Vue se usa mucho los import y export, es importante que conozcas cómo funcionan. Ten en cuenta que aunque en el video muestra la exportación e importación de funciones, también se puede hacer de clases, constantes, etc.

Ejercicios autoevaluación

Al finalizar cada video tendrás ejercicios de autoevaluación. **Recomiendo que los hagan.**

Proyecto



Vamos a seguir trabajando sobre el marketplace:

- Obligatorio: pasa a componentes las diferentes partes de tu página. Por ejemplo, el sector con información del producto, cada imagen mostrada en la página principal, etc.
- Obligatorio: usa la comunicación de eventos para poder conectar los componentes hijos con los padres, y notificar de los cambios
- Obligatorio: usar los formularios de de Vue.js para hacer funcional el formulario del feedback.

Clase 13

- Fecha: 07/06/2023
- Temario: Vue.js

Recursos

- En caso que no lo hayas hecho para la clase anterior ->   Instalar Node.js, NPM y YARN (en Windows10) : en caso que no tengas instalado ni NPM ni Yarn, hazlo. No es obligatorio que instales “Yarn” , aunque lo recomiendo.
- En caso que no lo hayas visto para la clase anterior -> [Import y export en JavaScript](#): debido a que en Vue se usa mucho los import y export, es importante que conozcas cómo funcionan. Ten en cuenta que aunque en el video muestra la exportación e importación de funciones, también se puede hacer de clases, constantes, etc.
- [Repositorio GitHub usado en las siguientes lecciones.](#)
- [Lección 1 - Introducción.](#)
- [Lección 2 - Creando nuestro proyecto.](#)
- [Lección 3 - Single File Components.](#)

Ejercicios autoevaluación

Haremos la misma agenda de tareas que has hecho para la clase 5, pero en Vue. Crea el proyecto con Vue CLI, y agrega todas las características pedidas.

Proyecto

Crear un proyecto con Vue CLI, y mover todo lo hecho en el marketplace al mismo.

Clase 14

- Fecha: 14/06/2022
- Temario: Vue.js

Recursos

- [Lección 4 - Fundamentos de Vue Router.](#)
- [Lección 5 - Llamada a APIs con Axios.](#)

Ejercicios autoevaluación

Buscador de películas

Date de alta en el sitio <https://www.omdbapi.com/>, y [obtén un token](#) para consultar su API.

Cree una aplicación en Vue que tenga un input en donde se ingresará el nombre de alguna película a buscar. Usando la API de OMDb, a través del método http://www.omdbapi.com/?apikey=REEMPLAZAR_CON_TOKEN&t=Some , obtendrás la información necesaria para mostrar los detalles de la película (por ejemplo, el año de lanzamiento, director, etc).

Proyecto

Vamos a agregar el router e integración con Axios

Obligatorio

A las dos vistas que ya existen (pantalla principal y la del producto), usar Vue Router para navegarlas. Por ejemplo, una tendrá la ruta “/”, y la otra será “/product”.

Obligatorio

En la pantalla principal, donde se encuentra toda la lista de productos, agregarás un nuevo campo arriba en donde se indicará la cantidad de notebooks que actualmente están en ventas. ¿Cómo obtendrás ese dato?

Si accedes a

https://my-json-server.typicode.com/agustinruatta/fake_json_server_db/statistics podrás ver que retorna un JSON, el cual tiene un campo `amount_of_products`, que indica la cantidad de productos en oferta. Es por eso que, cuando cargues la página, deberás hacer una consulta GET a dicha URL, obtener la cantidad total de productos, y mostrarla en tu HTML.

Obligatorio

En la pantalla del producto estamos mostrando la información del mismo, aunque es un HTML fijo. ¡Usemos una API!

Si ingresas a

https://my-json-server.typicode.com/agustinruatta/fake_json_server_db/products/1, notarás que devuelve un JSON con toda la información del producto. Modifica la pantalla del producto para que cada vez que se cargue, llame a dicha URL y use la información devuelta para mostrar los datos.

Obligatorio

En la pantalla principal estás mostrando actualmente una lista de todos los productos que están a la venta. Pero dichos productos están puestos en el HTML de forma “hardcodeada” (es decir, escrito a mano en el HTML). Vamos a reemplazarlo con los que nos devuelve la API.



Si accedes a

https://my-json-server.typicode.com/agustinruatta/fake_json_server_db/products observarás que devuelve un JSON que contiene un array de productos. Cada vez que cargues la página HTML, haz una llamada HTTP GET a dicha URL, e imprime dentro del HTML los productos que devuelve.

Clase 15

- Fecha: 21/06/2022
- Temario: Vue.js

Recursos

- **Obligatorio:** [Lección 6 - Enrutamiento dinámico](#).
- **Obligatorio:**  Introducción a Vuex
- **Obligatorio:**  #06 | Vuex | Tienda Store | Manejo de estados | Curso Vue JS int...
- **Opcional:** [Cómo Construir Aplicaciones Complejas y a Gran Escala Vue.js con Vuex](#)
-> Este tutorial está enfocado a una versión anterior de Vuex (por eso notarás que la instalación es diferente), pero es útil para reforzar. Leer desde la sección “Propiedades de Estado” hasta “Vuex Asignación de Ayudantes” inclusive.
- **Opcional:** Si pueden leer en inglés, les recomiendo la documentación oficial de Vuex. Si leen la misma, no les será necesario leer los materiales anteriores.

Ejercicios autoevaluación

Integra Vuex al buscador de películas que estaba como tarea en la clase 14. Guarda en el store los datos de las películas, y todo lo que te parezca que debería almacenarse.

Proyecto

Implementar ruteo dinámico al marketplace para que cuando se presione sobre un producto en la pantalla principal, lo lleve a la url `"/product/:id"` (reemplazar el id con el del producto seleccionado) y muestre la información de esa notebook en particular.