Sprint 1

**Recursos:**para lograr la creación de un repositorio en Github, deben ingresar a los enlaces y acceder a los recursos indicados que se presentan a continuación, y a los que se les compartieron en la semana 1 en el material de estudio y el material de apoyo.

• Enlace para el manejo de repositorios públicos:

<https://github.com/>

• Enlace para la herramienta Git del equipo local:

<https://git-scm.com/download/win>

• Enlace para comandos básicos de Git:

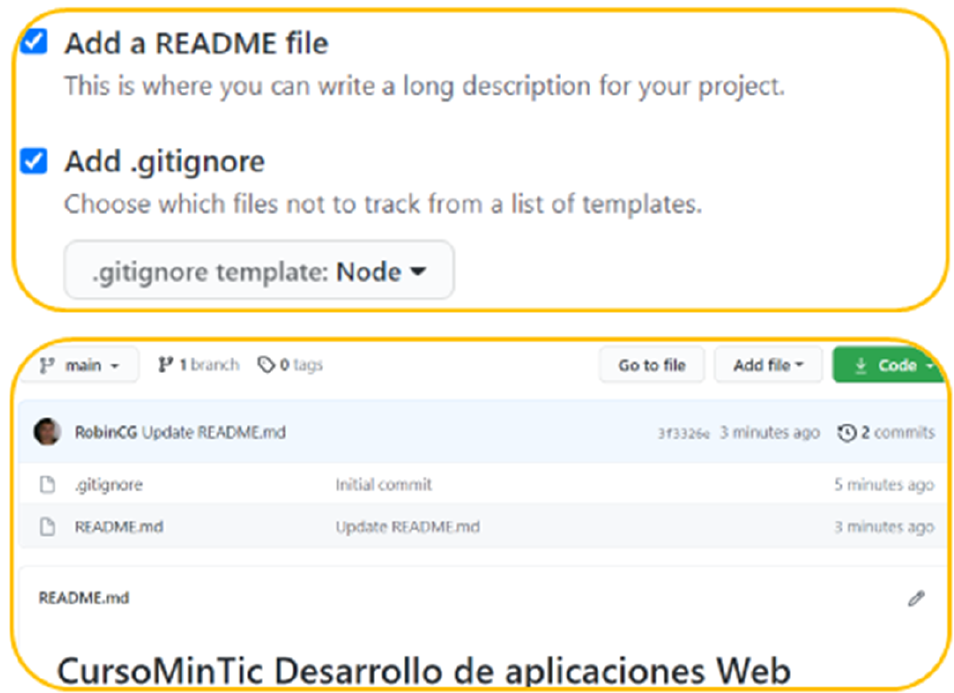
<https://www.freecodecamp.org/espanol/news/10-comandos-de-git-que-todo-desarrollador-deberia-saber/>

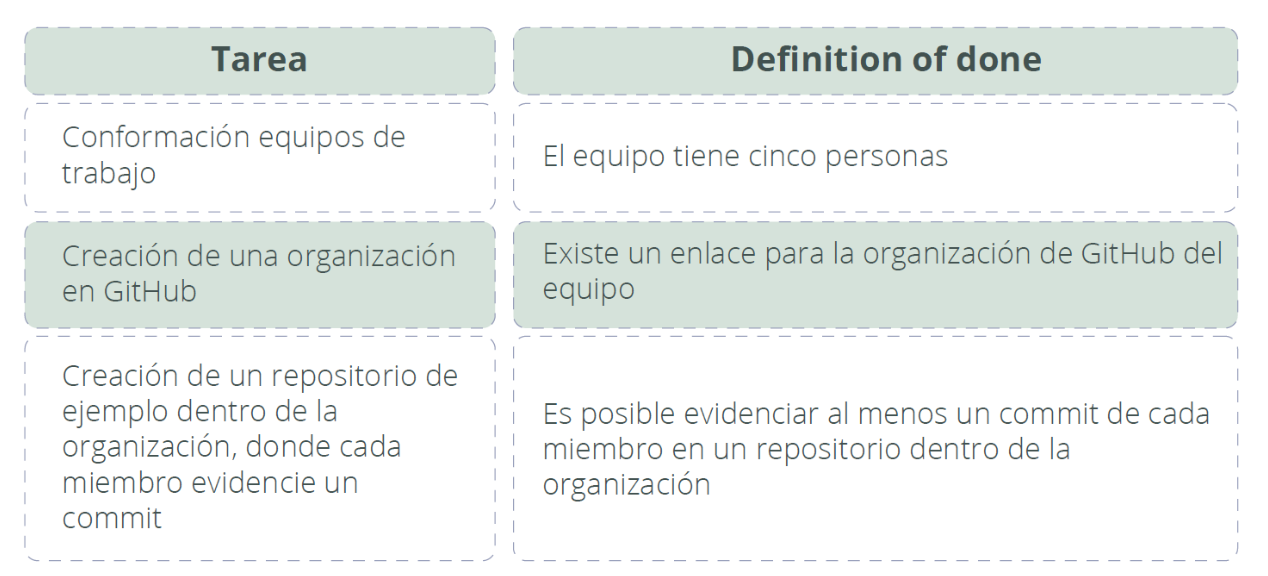
**Indicaciones:**

• Conformen un equipo de cinco integrantes y asígnenle un nombre.

• Analicen las historias de usuario propuestas y definan la tienda o emprendimiento a la que desean aplicarlo.

Se requiere construir un nuevo repositorio sobre la herramienta GitHub y definir sobre este repositorio las carpetas en las que se alojarán los artefactos que se generen a lo largo del ciclo de vida del proyecto de desarrollo de software, adicionando en la raíz del proyecto archivos iniciales como README y .gitignore para evitar versionar elementos que no requieren rastreo seleccionando la plantilla (template) del lenguaje de programación con el que haremos nuestros códigos, que debe ser Node.





Sprint 2

**Recursos:**para el desarrollo de esta unidad son necesarias las siguientes herramientas:

• IDE de desarrollo de su preferencia (se sugiere Visual Studio Code).

• GitHub.

• Para lograr la creación de las interfases gráficas de usuario se pueden basar en los siguientes mockups, y pueden adicionar los recursos que consideren necesarios.

Vista administrador - ventas:

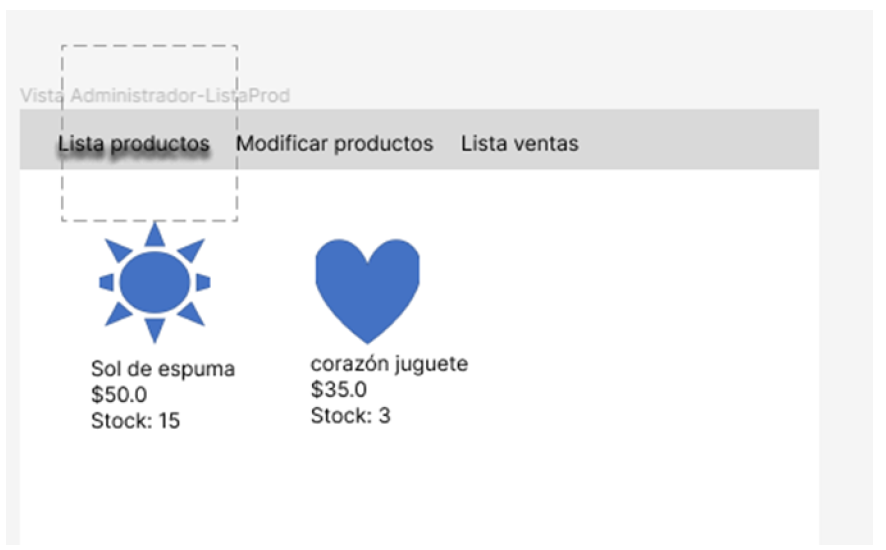


Vista administrador - modificar:



**NOTA:** el wireframe para modificar productos se puede tomar como base para registrar productos.

Vista administrador - lista de productos:



Vista cliente - carrito:



El enlace a Figma con estos wireframes se encuentra [aquí.](https://www.figma.com/file/kJXAdqycWH4lQfqB1lnZUm/Tienda%2FEmprendimiento)

Es posible utilizar plantillas de estilo gratuitas; deben ingresar a los enlaces y acceder a los recursos indicados que se presentan a continuación y a los que se les compartieron en la semana 3 en el material de estudio y el material de apoyo. Algunas plantillas CSS gratuitas se pueden encontrar en <https://nicepage.com/es/plantillas-css>

**Indicaciones:**

• Creen una SPA (single-page application con React), para lo cual se debe crear la aplicación con el comando npm create-react-app <nombreAplicación>; esto generará todos los paquetes necesarios para dicha creación.

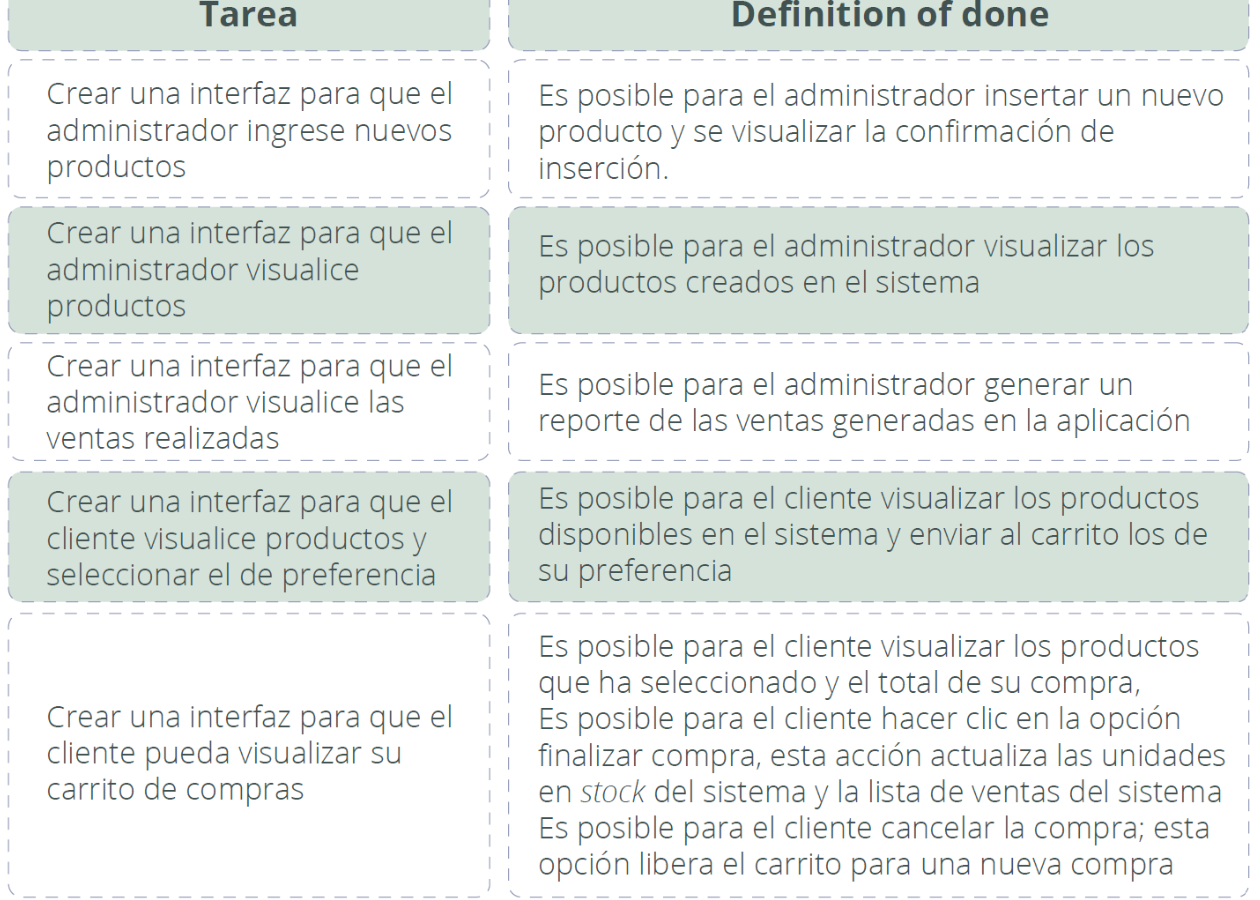
• Ahora es necesario crear los componentes del proyecto, que básicamente son aquellos que permitirán separar las interfaces de usuario en piezas independientes (pueden ser reutilizados cuando se requiera).

• Para ello se debe crear una carpeta denominada “components”, en la que se deben alojar todos los componentes que creen (estos deben estar alojados por separado).

• Generen la carpeta “styles”, en la cual deberán alojar los estilos que deseen usar en todo el proyecto.

• Modifiquen sus componentes e importen los estilos que crearon en sus archivos de componentes. Verifiquen que se están generando los cambios lanzando el proyecto.

• Se debe simular el consumo del API (backend) a través de un archivo JSON que va a tener los datos de prueba.



Sprint 3

**Recursos:**para el desarrollo de esta unidad son necesarias las siguientes herramientas:

IDE de desarrollo de su preferencia (se sugiere Visual Studio Code)

GitHub

**Indicaciones:**

Creen los servicios para el usuario administrador. Inicialmente pueden guardarse los valores en variables y mostrarlos por consola:

• Creen un nuevo producto con la siguiente información mínima: imagen (pendiente para cuando se desarrolle el frontend), nombre, descripción, valor, cantidad en stock.

• Creen el proceso para la modificación del producto.

• Creen un proceso que permita agregar unidades a un producto.

• Creen un proceso que permita listar los productos disponibles.

Creen los servicios para el usuario cliente. Inicialmente pueden guardarse los valores en variables y mostrarlos por consola:

• Creen un proceso para listar los productos disponibles para la venta (entiéndase por productos disponibles todos aquellos que tienen una unidad o más en stock).

• Creen el proceso para seleccionar productos y guardarlos en un arreglo.

• Creen un proceso que al seleccionar finalizar compra muestre el listado de los productos seleccionados y el valor a pagar, y que en el momento en que el cliente seleccione pagar se actualice el stock de los productos haciendo el respectivo descuento y sume en una variable acumuladora el valor de la venta.

Tabla, Calendario

Descripción generada automáticamente

Sprint 4

**Recursos:**para lograr el diseño e implementación de la base de datos en MongoDB es necesario contar con la instalación de la aplicación de MongoDB o usar la aplicación en la nube en su versión gratuita MongoDB Atlas.

Enlace de descarga de MongoDB y tutorial oficial de instalación:

<https://www.mongodb.com/docs/manual/installation/>

Enlace para trabajar virtualmente con MongoDB Atlas:

<https://www.mongodb.com/cloud/atlas/register>

IDE de desarrollo de su preferencia (se sugiere Visual Studio Code).

GitHub

[Docker](https://lms.misiontic2022udea.com/mod/resource/view.php?id=47111) desktop

DockerHub

**Indicaciones**

Teniendo en cuenta el siguiente **modelo de datos**:

Clientes

Productos

Ventas

Creen la base de datos y las respectivas colecciones usando el Shell de Mongo o, en su defecto, con interfaz gráfica de Mongo Atlas. Pueden revisar un ejemplo en la documentación de MongoDB en el enlace siguiente: <https://www.mongodb.com/es/basics/create-database#:~:text=En%20MongoDB%20Compass%2C%20cree%20una,de%20datos%E2%80%9D%20(Create%20Database)>

De acuerdo con el modelo, creen las colecciones necesarias e identifiquen los comandos sobre MongoDB para poblar datos en las estructuras del proyecto. Pueden hacerlo por el Shell de MongoDB con la sentencia createCollection(“ubicar datos”). Pueden apoyarse en el siguiente tutorial, que detalla la creación de colecciones por consola: <https://lineadecodigo.com/mongodb/crear-una-coleccion-en-mongodb/>

Es necesario que todo el equipo identifique los comandos para traer la información almacenada en las estructuras, dado que estas se van a utilizar posteriormente desde el backend. Pueden guiarse con algunos comandos en el siguiente enlace: <https://geekflare.com/es/mongodb-queries-examples/>

Imagen que contiene Tabla

Descripción generada automáticamente