



UNIVERSIDAD DE
GUADALAJARA
Red Universitaria de Jalisco

Universidad de Guadalajara

Proyecto II

Actividad integradora. Sprint, el corazón de SCRUM

Asesora:

Elizabeth Cristina Hernández Hernández

Equipo:

Javier Hernández Hernández

Juan Pablo Hernández Rangel

Teresita de Jesús Gutiérrez Ávila

Índice

Introducción	pág. 3
Objetivo	pág. 3
Instrucciones	pág. 3
Product Backlog Visualización de la App	pág. 4
Sprint Backlog	pág. 6
Calendario	pág. 9
Creación de Código Github.....	12
Referencias	pág. 19

Introducción

En el universo ágil de SCRUM, los sprints son bloques de tiempo corto y enfocado donde los equipos trabajan intensivamente para lograr metas específicas. Esta práctica no solo mejora la productividad, sino que también permite una adaptación continua a medida que evolucionan los requisitos del proyecto. En este ejercicio, te sumergirás en el mundo dinámico de SCRUM, planificando y ejecutando un Sprint para un proyecto de diseño web. A medida que avances, no solo desarrollarás competencias en gestión ágil, sino que también experimentarás la satisfacción de entregar resultados tangibles en un periodo de tiempo definido.

Objetivo

Desarrollarás habilidades en la identificación y gestión de tareas, la colaboración efectiva en equipo, y la adaptación ágil a cambios en los requisitos del proyecto.

InSTRUCCIONES

1. Con base en el Backlog que llevaste a cabo en la actividad 2, realiza lo siguiente:

- Desarrolla y aplica el Scrum
- Desarrolla el código y documentación de sprint planning
- Desarróllalo en React
- Aloja el código en GitHub

2. En un documento integra:

- El acceso de GitHub
- Product Backlog
- Fichas de Backlog
- Calendarización de los Sprint

3. Envía tu producto al buzón y espera los comentarios por parte de tu asesor.

Product Backlog Visualización de la App Ventanilla Única

Identificador (ID) de la Historia	Enunciado de la Historia	Alias	Estado	*Dimensión/ Esfuerzo	Interacción (Sprint)	Prioridad	Comentarios
US01	Yo como usuario quiero ver la pantalla inicial con el título "Ventanilla Única" para identificar la aplicación.	Pantalla inicio	Por hacer	3 pts	Sprint 1	Alta	Asignado a Javier
US02	Yo como usuario quiero que el sistema muestre un campo de entrada de datos en la pantalla inicial para poder interactuar con la aplicación.	Input inicial	Por hacer	5 pts	Sprint 1	Alta	Asignado a Teresita
US03	Yo como usuario quiero que la pantalla inicial tenga un estilo atractivo (fondo, tipografía, colores) para una mejor experiencia visual.	Estilos inicio	Por hacer	5 pts	Sprint 1	Media	Asignado a Juan Pablo
US04	Yo como desarrollador quiero inicializar el proyecto en React para poder implementar la aplicación web.	Crear proyecto	Por hacer	8 pts	Sprint 1	Alta	Asignado a Javier

US05	Yo como desarrollador quiero instalar dependencias necesarias (React DOM, React Scripts, etc.) para ejecutar correctamente la app.	Dependencias	Por hacer	3 pts	Sprint 1	Alta	Asignado a Juan Pablo
US06	Yo como desarrollador quiero configurar los primeros componentes (App.js, index.js) para estructurar la aplicación.	Config inicial	Por hacer	5 pts	Sprint 1	Alta	Asignado a Teresita
US07	Yo como usuario quiero que el input inicial tenga borde redondeado y colores llamativos para facilitar la interacción.	Input diseño	Por hacer	3 pts	Sprint 1	Media	Asignado a Juan Pablo
US08	Yo como desarrollador quiero verificar que la aplicación se despliegue correctamente en el navegador para garantizar su funcionamiento.	Verificación	Por hacer	2 pts	Sprint 1	Alta	Asignado a Javier
US09	Yo como equipo quiero documentar el paso a paso de la creación de la pantalla inicial para futuras referencias.	Documentación	Por hacer	2 pts	Sprint 1	Media	Asignado a Teresita

Sprint Backlog

Elemento de trabajo pendiente	Puntos de historia	Responsable	Estado	Estimado original	Día 1	Día 2	Día 3	Día 4	Día 5	Revisión del Sprint
US01 – Pantalla inicio	3	Javier	Finalizado	6h	3h	2h	1h	0h	0h	Completo
Definir título y estructura inicial	1	Javier	Finalizado	2h	1h	1h	0h	0h	0h	Completo
Integrar texto en componente App.js	1	Javier	Finalizado	2h	2h	0h	0h	0h	0h	Completo
Ajustar estilos de tipografía	1	Javier	Finalizado	2h	0h	1h	1h	0h	0h	Completo
US02 – Input inicial	5	Teresita	Finalizado	10h	3h	3h	2h	2h	0h	Completo
Crear componente input básico	2	Teresita	Finalizado	4h	2h	2h	0h	0h	0h	Completo
Validar interacción de entrada	2	Teresita	Finalizado	3h	1h	1h	1h	0h	0h	Completo
Ajustar bordes y diseño visual	1	Teresita	Finalizado	3h	0h	0h	1h	2h	0h	Completo
US03 – Estilos inicio	5	Juan Pablo	Finalizado	10h	2h	3h	2h	2h	1h	Completo
Definir colores principales	0.5	Juan Pablo	Finalizado	2h	1h	1h	0h	0h	0h	Completo

Configurar fondo y contraste	2	Juan Pablo	Finalizado	3h	0h	2h	1h	0h	0h	Completo
Ajustar fuentes globales	0.5	Juan Pablo	Finalizado	2h	0h	0h	1h	1h	0h	Completo
Aplicar diseño responsivo	2	Juan Pablo	Finalizado	3h	1h	0h	0h	1h	1h	Completo
US04 – Crear proyecto React	8	Javier	Finalizado	12h	4h	4h	2h	2h	0h	Completo
Inicializar proyecto con create-react-app	2	Javier	Finalizado	3h	3h	0h	0h	0h	0h	Completo
Configurar estructura carpetas	2	Javier	Finalizado	3h	0h	2h	1h	0h	0h	Completo
Verificar ejecución inicial	2	Javier	Finalizado	2h	1h	0h	1h	0h	0h	Completo
Subir proyecto a repositorio	2	Javier	Finalizado	4h	0h	2h	0h	2h	0h	Completo
US05 – Instalar dependencias	3	Juan Pablo	Finalizado	6h	2h	2h	2h	0h	0h	Completo
Instalar dependencias base	1.5	Juan Pablo	Finalizado	3h	2h	1h	0h	0h	0h	Completo
Validar versiones	1.5	Juan Pablo	Finalizado	3h	0h	1h	2h	0h	0h	Completo
US06 – Config inicial	5	Teresita	Finalizado	10h	3h	2h	2h	2h	1h	Completo

Configurar App.js e index.js	3	Teresita	Finalizado	5h	2h	1h	1h	1h	0h	Completo
Crear componentes iniciales	1	Teresita	Finalizado	3h	1h	1h	0h	1h	0h	Completo
Validar renderizado	1	Teresita	Finalizado	2h	0h	0h	1h	0h	1h	Completo
US07 – Input diseño	3	Juan Pablo	Finalizado	6h	2h	2h	1h	1h	0h	Completo
Estilos de bordes y colores	1.5	Juan Pablo	Finalizado	3h	1h	1h	1h	0h	0h	Completo
Aplicar animaciones	1.5	Juan Pablo	Finalizado	3h	1h	1h	0h	1h	0h	Completo
US08 – Verificación app	2	Javier	Finalizado	4h	2h	1h	1h	0h	0h	Completo
Ejecutar en navegador	1	Javier	Finalizado	2h	2h	0h	0h	0h	0h	Completo
Validar consola (errores)	1	Javier	Finalizado	2h	0h	1h	1h	0h	0h	Completo
US09 – Documentación	2	Teresita	Finalizado	4h	2h	1h	1h	0h	0h	Completo
Documentar pasos iniciales	1	Teresita	Finalizado	2h	1h	1h	0h	0h	0h	Completo
Redactar guía de instalación	1	Teresita	Finalizado	2h	1h	0h	1h	0h	0h	Completo
Total	36			136h	46h	40h	28h	18h	4h	

Calendario

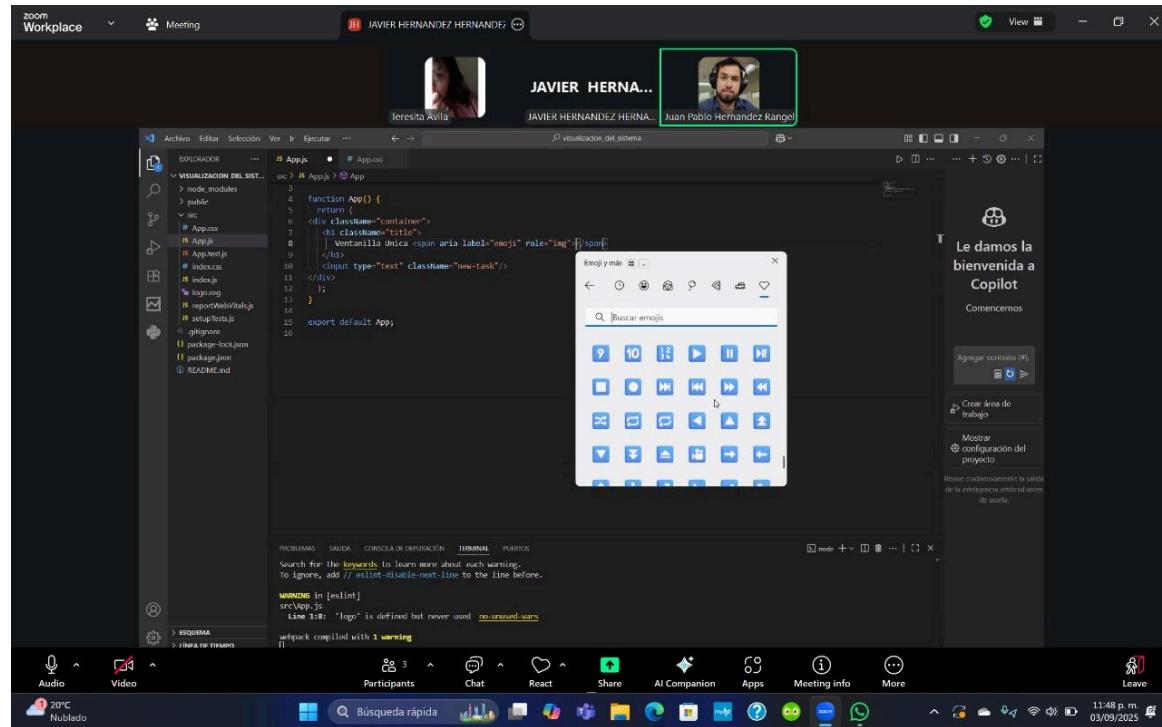
Día	Tarea / Historia	Responsable	Horas estimadas	Revisión del Sprint
Día 1	US01 – Pantalla inicio / Definir título y estructura	Javier	3h	Completo
Día 1	US02 – Input inicial / Crear componente input básico	Teresita	2h	Completo
Día 1	US02 – Validar interacción de entrada	Teresita	1h	Completo
Día 1	US03 – Estilos inicio / Definir colores principales	Juan Pablo	1h	Completo
Día 1	US04 – Crear proyecto React / Inicializar proyecto	Javier	4h	Completo
Día 2	US01 – Integrar texto en App.js	Javier	2h	Completo
Día 2	US02 – Crear componente input básico	Teresita	2h	Completo

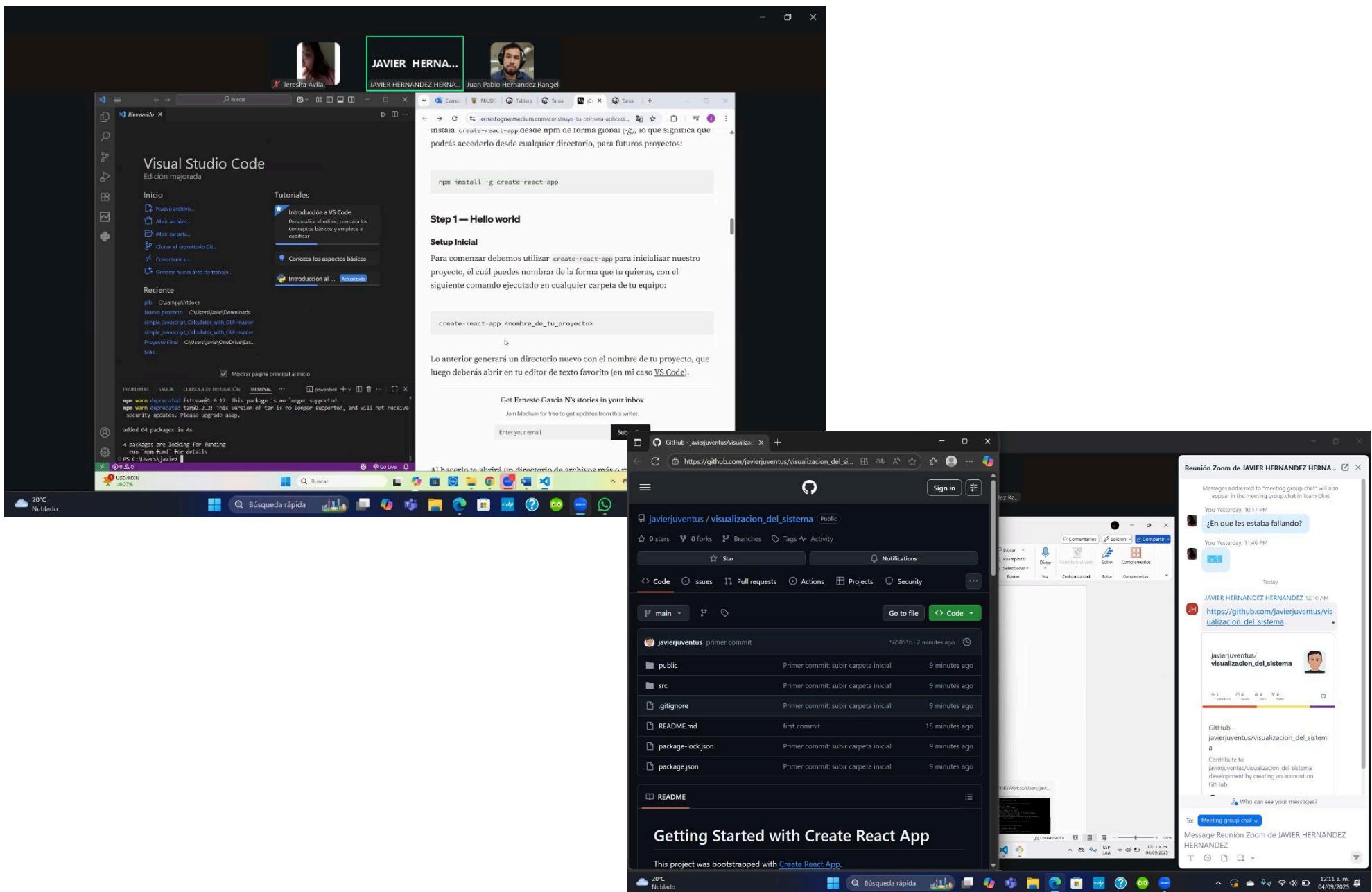
Día 2	US02 – Validar interacción de entrada	Teresita	1h	Completo
Día 2	US03 – Configurar fondo y contraste	Juan Pablo	2h	Completo
Día 2	US04 – Configurar estructura carpetas	Javier	4h	Completo
Día 3	US01 – Ajustar estilos de tipografía	Javier	1h	Completo
Día 3	US02 – Ajustar bordes y diseño visual	Teresita	1h	Completo
Día 3	US03 – Ajustar fuentes globales	Juan Pablo	1h	Completo
Día 3	US04 – Verificar ejecución inicial	Javier	2h	Completo
Día 4	US02 – Ajustar bordes y diseño visual	Teresita	2h	Completo
Día 4	US03 – Aplicar diseño responsivo	Juan Pablo	1h	Completo

Día 4	US04 – Subir proyecto a repositorio	Javier	2h	Completo
Día 5	US06 – Config inicial / Configurar App.js e index.js	Teresita	1h	No iniciado
Día 5	US06 – Crear componentes iniciales	Teresita	0h	No iniciado
Día 5	US06 – Validar renderizado	Teresita	1h	No iniciado
Día 5	US07 – Input diseño / Estilos de bordes y colores	Juan Pablo	0h	No iniciado
Día 5	US07 – Aplicar animaciones	Juan Pablo	0h	No iniciado

Creación de código - GitHub

Link: http://github.com/javierjuventus/visualizacion_del_sistema





Explicación del código de la aplicación React

1. Introducción general

Este proyecto es una aplicación web desarrollada con React utilizando Create React App. Su propósito es visualizar un sistema específico, aunque el repositorio no proporciona detalles adicionales sobre el sistema en cuestión.

2. Estructura del proyecto

El repositorio contiene la siguiente estructura de directorios y archivos:

public/: Contiene archivos estáticos como el index.html.

src/: Contiene los archivos fuente de la aplicación React.

.gitignore: Especifica qué archivos o directorios deben ser ignorados por Git.

README.md: Proporciona información sobre el proyecto.

package.json: Define las dependencias y scripts del proyecto.

package-lock.json: Registra las versiones exactas de las dependencias instaladas.

3. Dependencias principales

El proyecto utiliza las siguientes dependencias:

react: Biblioteca principal para construir la interfaz de usuario.

react-dom: Proporciona métodos específicos del DOM para React.

react-scripts: Herramientas de construcción y desarrollo para aplicaciones React.

Historia / Tarea	Explicación	Fragmento de código relacionado
US01 – Pantalla inicio	Definir título y estructura inicial	Se crea la estructura básica de la interfaz con un <code>div</code> contenedor y un título principal.
Integrar texto en componente App.js	El texto “Ventanilla Unica” se coloca dentro del <code><h1></code> .	<pre>jsx
<div className="container"><h1 className="title">Ventanilla Unica </h1></div>
</pre>
Ajustar estilos de tipografía	Se aplica color blanco y centrado al título.	<pre>css
.title { text-align: center; color: white; }
</pre>
US02 – Input inicial	Crear componente input básico	Se agrega el campo de entrada para futuras tareas.
Validar interacción de entrada	Aún pendiente de implementación, se tendría que manejar con <code>onChange</code> en React.	<pre>jsx
<input type="text" className="new-task" /></pre> <p>(No implementado todavía)</p>
Ajustar bordes y diseño visual	Se definen bordes redondeados, colores y tipografía del input.	<pre>css
.new-task { background-color: azure; border-radius: 10px; border-color: aquamarine;
</pre>

US03 – Estilos inicio	Se establece fondo negro y contraste blanco en el título.	<code>css
body { background-color: black; }</code>
Definir colores principales		
Configurar fondo y contraste	Se asegura contraste negro/blanco + azule.	<code>css
body { background-color: black; } .title { color: white; }</code>
Ajustar fuentes globales	Definición de fuentes generales en <code>index.css</code> .	<code>css
body { font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', ... }</code>
Aplicar diseño responsivo	Falta aplicar <code>media queries</code> o <code>flexbox/grid</code> adaptativos.	(Pendiente)
S04 – Crear proyecto React	<code>index.js</code> y <code>App.js</code> son generados por <code>create-react-app</code> .	<code>js
import ReactDOM from 'react-dom/client';
root .render(<App />);</code>
Inicializar proyecto con <code>create-react-app</code>		
Configurar estructura carpetas	Archivos <code>App.js</code> , <code>App.css</code> , <code>index.js</code> muestran estructura inicial.	Estructura del proyecto
Verificar ejecución inicial	Al ejecutar <code>npm start</code> , se renderiza correctamente en navegador.	<code>js
root.render(<App />);</code>
Subir proyecto a repositorio	No aparece en el código, es parte de la gestión.	(Fuera del código)
S06 – Config inicial	<code>App.js</code> define la interfaz y <code>index.js</code>	<code>js
const root =</code>

Configurar App.js e index.js

renderiza en #root.

```
ReactDOM.createRoot(document  
 .getElementById('root'));  
root.render(<App />);
```

Crear componentes iniciales

App es el primer componente creado.

```
js<br/>function App() {  
 return (...) }
```

Validar renderizado

Confirmado al cargar la aplicación sin errores.

Ejecución en navegador

US07 – Input diseño

Estilos de bordes y colores

Ya definidos en .new-task.

```
css<br/>.new-task {  
 border-radius: 10px;  
 border-color: aquamarine; }
```

Aplicar animaciones

No implementado todavía.

(Pendiente)

US08 – Verificación app

Ejecutar en navegador

Se logra con npm start al renderizar App.

```
js<br/>root.render(<App />);
```

Validar consola (errores)

Es parte de pruebas manuales.

(Fuera del código)

US09 – Documentación

Documentar pasos iniciales / guía de instalación

No aparece en el código, debe ir en archivos como README.md.

(Pendiente)

Referencias

- Luque, A. (2016, 15 de mayo). Ejercicio Product Backlog y Sprint Backlog https://youtu.be/ySJMHC1-8Uo?si=xpxppyf5hUP_kpMr
- CECyT Digital Harbor. (2018, 6 de abril). El Sprint Backlog <https://youtu.be/fztgMv5Gl8k?si=wN-i202QIKddTpZG>
- Join Academia. (2015, 27 de mayo). Scrum Ejemplo Práctico de la Planeación del Sprint
<https://youtu.be/BNCDcGqYNwk?si=9kROha0pl0V9cp4p>
- GitHub. GitHub. <https://github.com>
- García N., E. (2019, 18 de febrero). ¡Construye tu primera aplicación web con ReactJS! Medium.
<https://medium.com/@ernestognw/construye-tu-primera-aplicaci%C3%B3n-web-con-reactjs-parte-1-4c8bccca3b9db>