ÁREA DE TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

PROGRAMA ACADÉMICO DE ENTORNOS VIRTUALES Y NEGOCIOS DIGITALES

SITIO WEB FARMACIA SANA SANA

PARA LA EMPRESA

Forma

Descripción generada automáticamente

NOMBRE DE LA EMPRESA

FARMACIA SANA SANA

PRESENTA

YULISA FERNANDA ARREOLA RRIVERA

JULIO CESAR RODRIGUEZ NAVA

BRANDON LOEZA BARRERA

ASESOR(A) ACADÉMICO(A): MA. EUGENIA HERNÁNDEZ SIERRA

ASESOR(A) ACADÉMICO(A): RODOLFO MARTÍNES PUENTE

ASESOR(A) ORGANIZACIONAL: MARISOL MORA MUÑOZ

GENERACIÓN: SEPTIEMBRE DICIEMBRE 2022

ACÁMBARO, GUANAJUATO. DICIEMBRE 2022

RESUMEN

El objetivo de esta sección, es sintetizar la información del proyecto documentado en el informe, para así invitar al lector a leer tu investigación completa.

El número de palabras de esta sección no debe exceder de 150, con un interlineado sencillo, en letra Arial, tamaño 12. Se recomienda estructurar de la siguiente manera:

* Antecedentes 1 o 2 oraciones
* Métodos 2 o 3 oraciones
* Resultados Menos de 10 oraciones
* Conclusión Una oración

Se redacta en tiempo presente para la oración de la pregunta de investigacióny de las conclusiones. Por otro lado, se redacta en tiempo pasado para los métodos y resultados. No incluye tablas, figuras ni referencias

**NOTA.** *De acuerdo a la Guía de Forma (Padilla, 2014) menciona:*

“*El docente que imparta la asesoría académica, determina si el resumen también se redacta en inglés. A este apartado se le llama* ***abstract****, y se coloca en la misma hoja del resumen con los lineamientos antes mencionados.*

*Se puede solicitar apoyo para dudas específicas del idioma inglés con el profesorado de la UGAC correspondiente, pero la responsabilidad recae en la asesoría académica. La UGAC de inglés publica sus horarios de asesoría desde la primera semana del cuatrimestre en los pizarrones de avisos de cada edificio*”

AGRADECIMIENTOS

Este apartado no debe ser mayor a una cuartilla, el título corresponde a primer orden sin estar numerado, por tal motivo es con las características correspondientes.. Los agradecimientos deben ser cortos y claros y redactarse en presente, evitando utilizar palabras innecesarias y frases rebuscadas. El desarrollo no es mayor a una cuartilla.

Se recomienda incluir la mención de las personas, instituciones, empresas y profesorado que colaboró para la realización del informe final de estadía. De igual manera, se mencionan instituciones que pudieron financiar la investigación, así como al titular de los derechos de autor de alguna figura o tabla usada (o mencionar al pie de la figura “*Reproducido con permiso de …*”).

¿Qué se agradece? Facilidades para entrar a la organización y hacer el estudio, soporte de material, financiamientos, soporte técnico, asistencia en redacción, realización de pruebas de laboratorio, etcétera.

**NOTA.** Se sugiere seguir el orden de aparición de agradecimientos siguiente: Agradecimientos referentes a la empresa, agradecimientos referentes a los asesores y agradecimientos referentes a la Universidad*.*

**ÍNDICE**

INTRODUCCIÓN

1. ANTECEDENTES 1
2. PROBLEMÁTICA 3
3. JUSTIFICACIÓN 4
4. OBJETIVOS 5
   1. OBJETIVO GENERAL 5
   2. OBJETIVOS METODOLÓGICOS 5
   3. ALCANCE 6
5. MARCO TEÓRICO 7
   1. SUBTEMA 1
   2. SUBTEMA 2
   3. SUBTEMA 3
6. METODOLOGÍA
   1. OBJETIVO METODOLÓGICO 1
   2. OBJETIVO METODOLÓGICO 2
   3. OBJETIVO METODOLÓGICO 3
7. RESULTADOS
   1. OBJETIVO METODOLÓGICO 1
   2. OBJETIVO METODOLÓGICO 2
   3. OBJETIVO METODOLÓGICO 3

CONCLUSIONES 20

REFERENCIAS 23

GLOSARIO

ANEXOS 25

**INTRODUCCIÓN**

En la actualidad existen muchas herramientas para la solución de diversas problemáticas que pueden presentar algunas empresas, como lo son la falta de tecnologías que les impiden actualizarse, varias de ellas utilizan la implementación de tic´s para poder llevar un control de sus ventas, un inventario, o darse a conocer por medio de una aplicación progresiva web.

Encontramos a la empresa Farmacia Sana Sana, la cual no tiene una aplicación web progresiva, por lo que se le hace más difícil llevar un control de dicha farmacia, habiendo identificado la problemática que presenta, se dio a la tarea de platicar con la dueña Marisol Mora, para poder entender y darle solución a su problemática, por lo que no le permite tener un control de cada uno de los productos con los que cuenta, de sus ofertas, precios, medios de contacto, esto debido a que no cuenta con una PWA, siendo estas de gran importancia ya que puede funcionar sin internet, y es más fácil estar viendo los productos a través de la app en el celular o computadora, brindando mayor comodidad, entre muchas más.

La redacción fue el primer paso, en donde el proyecto se define, los requerimientos tanto del sistema como de software, alcance de proyecto que es hasta donde pretende llegar, pero sobre todo y muy importante la planeación de este, e identificar las cartas de inicio del proyecto.

Así mismo para tener un mejor conocimiento y desenvoltura en el tema, la investigación realizada se basa en los conceptos básicos de la materia, como conocer y recordar que son las PWA y el manejo de distintos frameworks.

**INTRODUCCIÓN**

En la realización de los proyectos que se están desarrollando actualmente se están implementando algunos de los objetivos de sustentabilidad, los cuales sabemos que existen, teniendo el conocimiento de que se cuentan con 17 objetivos, la implementación de esto se lleva a cabo con la finalidad de proteger el planeta y reducir algunas problemáticas ambientales, sociales y económicas con las que cuenta nuestro país.

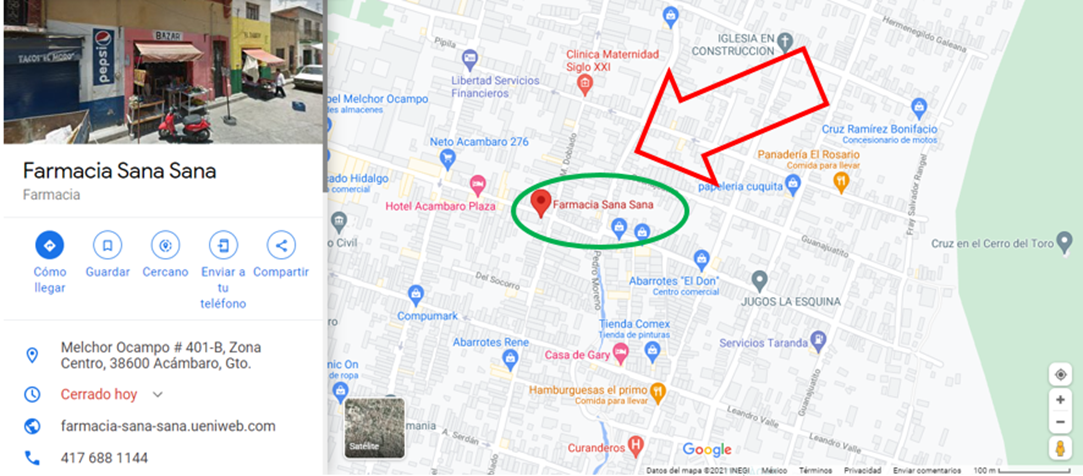
Por eso para la realización del proyecto de páginas web progresivas se implementará el objetivo de sustentabilidad de producción y consumos responsables, el cual evitará que se contamine más el medio ambiente, ya que no se imprimirán folletos, volantes, trípticos ni mucho menos, reducirá costos, y así mismo aumentar la eficiencia de recursos y promover estilos de vida sostenibles.

Este objetivo ayuda al proyecto en la implementación de tecnológica para avanzar hacia modalidades de consumo y producción más sostenibles, e incluso actualizándose evitando ocasionar el mayor desperdicio posible.

1. **ANTECEDENTES**

La Farmacia Sana Sana está ubicada en Melchor Ocampo # 401-B, Zona Centro, 38600 Acámbaro, Gto. Farmacia Sana Sana, unidad de negocio de Corporación GPF se fundó en el año 2000. Concebida como la cadena de farmacias propias más grande del país, está presente en las 24 provincias del Ecuador, contribuyendo al bienestar y la salud de cientos de comunidades y barrios, y de ahí se fue expandiendo en Guadalajara, México, Guanajuato, Querétaro. Ofrece a sus clientes excelencia en el servicio y un amplio portafolio de medicinas y productos para el cuidado de su salud a precios competitivos. Honrando su promesa de valor, está cerca de sus clientes, en más de 500 puntos de venta a nivel nacional, para llevarles salud y alivio, entregándoles su receta completa con los mejores precios.

Las tareas fundamentales que realizan en la Farmacia Sana Sana, son la atención de clientes y venta de medicamentos, pero no había tenido la oportunidad de adquirir un sitio web para su beneficio y de sus clientes. Por tal motivo se propuso implementar un sitio web con diferentes páginas, como lo son inicio, contactos, categorías de productos y ofertas. La Farmacia Sana Sana de Acámbaro Gto, es una empresa de sector privado, de tamaño pequeña con giro comercial.



**Figura 1. Ubicación. Fuente: www.google.com.mx/maps.**

1. problemática

Se identificó que la Farmacia Sana Sana de Acámbaro Gto, necesita una aplicación web progresiva, con las páginas de inicio, contactos, categorías de productos, y ofertas, debido a que tienen problemas con este tipo de gestión, impidiéndoles a sus clientes poder tener acceso a los productos de la Farmacia Sana Sana. Esta necesidad ha tratado de ser resuelta anteriormente, pero se han presentado diversos inconvenientes debido a implementaciones antiguas. Con esto se busca llevar a cabo la solución al problema, para así poder promocionar sus productos, y tener más tiempo para la realización de otras actividades dentro de la empresa. El sitio web tendrá un apartado que solo podrá utilizar el administrador, donde podrá colocar los productos con los que cuenta la farmacia, así como sus posibles ofertas, esto brindará un área de oportunidad al tener más control con el contenido del sitio web y una captura de productos rápida.

Las imágenes que contendrá el sitio serán imágenes de autoría propia, por lo que las imágenes colocadas en el sitio web serán imágenes tomadas en la misma farmacia, esto con la finalidad de no incumplir con los derechos de autor.

1. justificación

El propósito del proyecto es crear un sitio web progresiva para la Farmacia Sana Sana de Acámbaro Gto, diferente y único en cuanto a los diseños y colores que tienen los otros tomando en cuenta la psicología de los colores, para poder satisfacer a los clientes, con la finalidad de que también la empresa Farmacia Sana Sana dé a conocer sus productos. Con ello, obtendrá distintos beneficios, entre ellos:

* Consulta de sus productos y ofertas.
* Disminuir tiempo de captura de productos y medicamentos con los que cuenta la Farmacia Sana Sana.
* Los productos se llenarán con un formulario que estará definido con campos delimitados.
* Se llevará un control de todos los productos a través de un módulo especializado.

De no llevar a cabo el sitio web en la Farmacia Sana Sana le afectaría en la captura y promoción de sus productos, para esto se utilizarán los Frameworks de BackEnd (Administrador) y FrontEnd (Clientes), los cuales servirían para optimizar procesos, pero sobre todo tiempo, beneficiando principalmente a la dueña de la Farmacia Sana Sana quien es Marisol Mora Muñoz. Es importante implementar el uso de las tecnologías de la información, por lo que es una parte fundamental para llevar a cabo el proyecto, nos brinda las herramientas necesarias para poder comunicarnos y tener un fácil acceso a la información, por lo que algunos de los dispositivos que comúnmente utilizamos son los teléfonos, computadoras, radios, televisores, tablets, pero principalmente el internet.

1. objetivos
   1. OBJETIVO GENERAL

Desarrollar un sitio web para la Farmacia Sana Sana de la sucursal de Acámbaro Gto, que permita gestionar sus productos y ofertas.

* 1. OBJETIVOS METODOLÓGICOS

Planear la estructura de actividades mediante un diagrama de Gantt.

Aplicar por medio de técnicas de recolección de información.

Realizar un análisis de la información recabada.

Diseñar los diagramas UML.

Realizar un boceto del sitio web a mano alzada.

Diseñar los diagramas de las bases de datos.

Codificar las páginas y base de datos del sitio web.

Realizar pruebas para poder identificar las fallas.

Realizar los cambios correspondientes y necesarios para su buen funcionamiento.

Implementar el sitio web en un hosting de paga.

* 1. **ALCANCE**

A continuación, se muestran los siguientes alcances que contiene el sitio web:

* + Determinar los requerimientos funcionales y no funcionales para cumplir el objetivo general del sitio web.
  + Desarrollar bocetos a mano alzada.
  + Desarrollar el sitio web con acceso a base de datos.
  + Desarrollar la codificación del sitio web.
  + Administrar la base de datos.
  + Se desarrolla en el área administrativa para la gestión de productos de manera digital, de la Farmacia Sana Sana de Acámbaro Gto.
  + El sitio web contará con un solo administrador el cual será el único que podrá gestionar los productos.
  + Contendrá los formularios de inicio, productos, contactos, y categorías.
  + Contará con un login para el administrador, el cual le permitirá acceder a los módulos de contactos, productos, y categorías, teniendo la capacidad de editar, eliminar y crear.
  + El sitio web será responsivo.
  + Se entregará el código del proyecto y script de la base de datos, el sitio web se subirá a un hosting de paga.
  + Al terminar el sitio web, no se dará mantenimiento, al menos que la empresa Farmacia Sana Sana lo solicite de manera externa al proyecto.
  + Únicamente se le impartirá capacitación del manejo del sitio web a la dueña de la Farmacia Sana Sana.

1. **MARCO TEÓRICO**

**Investigación para inicio de proyecto**

**PWA.** Investigación sobre aplicaciones:

**Definiciones de PWA**

**Definición 1:** Progressive web apps (PWA) o aplicación web progresiva es una solución basada en la web tradicional que todos conocemos, aunque incorpora algunas particularidades que la hacen parecerse a una app nativa para teléfonos móviles y tabletas. (Vidal, 2019)

**Definición 2:** Son aplicaciones web que utilizan APIs y funciones emergentes del navegador web junto a una estrategia tradicional de mejora progresiva para ofrecer una aplicación nativa como la experiencia del usuario para aplicaciones web multiplataforma. Las aplicaciones web progresivas son un patrón de diseño útil, aunque no son un estándar formalizado. Se puede pensar que PWA es similar a AJAX u otros patrones similares que abarcan un conjunto de atributos de aplicación, incluido el uso de tecnologías y técnicas webs específicas. (*Aplicaciones Web Progresivas | MDN*, n.d.)

**Definición 3:** Las aplicaciones web progresivas están a medio camino entre las dos anteriores: son básicamente páginas web, pero mediante el uso de Service Workers y otras tecnologías se comportan más como aplicaciones normales que como aplicaciones web.

Mediante los Service Workers y otras tecnologías las aplicaciones web progresivas pueden seguir ejecutándose en segundo plano sin tener que vivir dentro del navegador. (Iván Ramírez, 2018)

**Definición 4:** Una aplicación web progresiva o PWA es un software de aplicación que se basa en tecnologías como HTML, CSS y JavaScript. Es decir, una PWA es una página web que se ve y se comporta como una aplicación móvil normal. (*¿Qué Es PWA? | Conceptos de Comercio Electrónico*, n.d.)

**Definición de cada framework y características**

1. **Angular:** Es un excelente marco para trabajar, es completamente modular en su naturaleza, con cada elemento Angular totalmente compatible con proyectos creados con otros marcos. Todo el proceso de desarrollo con Angular también se hace mucho más fácil, ya que está diseñado para evitarle la molestia de lidiar con tareas monótonas y repetitivas. Las aplicaciones web progresivas que se crean con Angular también se benefician enormemente de su método de compilación único, Ahead of Time (AOT), que permite una representación más rápida de la aplicación al convertir el código HTML y TypeScript de Angular en código JavaScript eficiente antes de que el navegador descargue y ejecute la aplicación.



**Figura *2*. Angular. Fuente: https://worldvectorlogo.com/es/logo/angular-3.**

**Características**

1. Metodología bien diseñada.
2. Gran y activa comunidad de desarrolladores.
3. Infraestructura compleja adecuada para proyectos de gran escala.
4. Gran archivo de extensiones de terceros para elegir.
5. Optimizado para la velocidad y la seguridad, con un concepto único y brillante: "compilador adelantado".
6. Inyección de dependencia para aumentar la eficiencia y modularidad del código.
7. Elementos completamente modulares.
8. Fácil implementación ya que la metodología utilizada está claramente definida
9. La adición de CLI ha acortado la curva de aprendizaje para el marco
10. Incluye IntelliSense y Typescript.
11. Excelente soporte, ya que Google lo respalda.
12. **Vue.js:** Presenta una de las bibliotecas de más rápido crecimiento entre todos los demás marcos.

Esto se debe a que el marco ha logrado descifrar el código al ofrecer dos cosas esenciales: renderizado de alta velocidad y codificación simple.



**Figura *3*. Vue.js. Fuente: https://www.vectorlogo.zone/logos/vuejs/index.html.**

**Características**

1. Grandes jugadores como Laravel y Alibaba apoyan el marco
2. Dado que el código es simple; es fácil para los nuevos desarrolladores aprenderlo
3. El marco es bastante simple y tiene conceptos similares como Angular y React.
4. Accesible.
5. Versátil: Su núcleo es bastante pequeño y se escala a través de plugins, con lo cual escucharás mucho que Vue es una librería muy parecida a React, una librería que cumple un propósito.
6. Escalable por el mismo tema de la versatilidad.
7. Reactivo.
8. Optimizado: Su core ocupa 74KB, como ves es bastante liviano.
9. Comunidad: Va creciendo a un ritmo importante con más 66500 estrellas en GitHub y 130 personas contribuyendo al core cada día.
10. Licencia MIT: se publicó bajo el amparo de esta licencia.
11. **React:** Con el respaldo de Facebook, React es uno de los marcos PWA más populares y altamente funcionales. Es famoso entre los desarrolladores debido a su extensa biblioteca de JavaScript.

El marco utiliza JSX para vincularse con estructuras HTML. Uno puede usar varios paquetes disponibles para ampliar aún más su aplicación y proyecto.



**Figura *4*. React. Fuente: https://nightdeveloper.net/react-provider-patron-diseno/react\_logo\_wordmark/.**

**Características**

1. Un ecosistema omnipresente
2. Facebook mantiene el código, lo que garantiza operaciones y soporte sin problemas.
3. El marco ayuda a crear aplicaciones escalables y flexibles
4. El código usado en React para aplicaciones web también se puede usar para aplicaciones nativas
5. El proceso de renderizado es muy rápido, gracias a Virtual-DOM
6. **Ionic:** Se introdujo en 2013. Es un marco de código abierto que se basa en el marco Angular y en Apache Cordova. A partir de ahora, el marco se ha utilizado para crear más de 5 millones de aplicaciones híbridas.

Tiene una biblioteca de componentes tanto para iOS como para Android. Ionic permite a los desarrolladores crear páginas web que se ejecutan dentro del navegador de un dispositivo usando WebView, que esencialmente representa las páginas web para que se parezcan a una aplicación nativa.



**Figura *5*. Ionic. Fuente: https://www.itop.es/soluciones-tecnologicas/apps/ionic.html.**

**Características**

1. Dado que Ionic es de código abierto, se puede adquirir de forma gratuita, lo que reduce el costo de desarrollo.
2. Es más fácil para los desarrolladores aprender cómo funciona.
3. Las herramientas integradas permiten un fácil mantenimiento.
4. Cuenta con una gran comunidad de usuarios.
5. Tiene una biblioteca muy extensa. Puede acceder a las API sin codificación

(Mehul Rajput, 2001)

* + - **Tabla comparativa** entre los frameworks investigados.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Comparativa / Framework | Angular | Vue.js | React | Ionic |
| Logo | Angular | Vue | React | Ionic |
| Costo | Gratuito | Gratuito | Gratuito | Versión gratuita, prueba gratis, paga. |
| Requerimientos | IDE.  Git.  Node.  Npm.  CLI Angular 8. | HTML  CSS  CLI  JavaScript. | Node. js. (LTS,). npm  npx | NodeJS  Npm  Apache Cordova |
| S.O | Multiplataforma | Multiplataforma | sistemas operativos libres | Android,  IOS, Windows. |
| Lenguaje | JavaScript de código abierto escrito en TypeScript. | JavaScript | JavaScript | HTML5, CSS. |
| Aplicación laboral | GitHub Google, Forbes, Indiegogo, BMW, Nike, Xbox, Udacity, YoutubeTV, Firebase, AT&T y Adobe. | Nintendo, Passionate People, FindlayWebTech, Adobe, BridgeU, Eva, Gitlab, Coursedog, Extradom. | Facebook, Walmart, Instagram, Artsy, Coinbase Pro, Airbnb, Tesla, Discord. | Burger King, Huawei, Microsoft, BMW, NASA.  EA Sports. |
| Licencia | MIT | MIT | GNU GPL, LGPL, y Licencia BSD | MIT |

* + **Framework más utilizado.**

**Express.js**

Este es el framework más conocido y utilizado por los desarrolladores de Node.js. Minimalista y con una curva de aprendizaje bastante sencilla, cuenta con un ecosistema muy completo para ejecutar lo que necesites, con las funcionalidades separadas en diferentes módulos.

Se emplea, entre otros usos, para:

Desarrollar REST APIs que sirven para comunicarse con las aplicaciones clientes.

Crear aplicaciones tradicionales MVC.

(mytaskpanel, n.d.)

Este Framework fue creado por TJ Holowaychuk, uno de los integrantes del equipo central que Desarrollo Node JS. Express JS es respaldado por una gran comunidad de Desarrolladores, entre las ventajas que tiene este Framework, es la continua actualización de sus características principales. Express JS es un Framework minimalista y te permite crear aplicaciones Web y APIs.

Su API robusta permite a los usuarios configurar rutas para enviar y recibir solicitudes entre la vista Front y la Base de Datos actuando como un servidor HTTP, asimismo es compatible con muchos paquetes NPM y otros motores de plantillas como Pug, Moustache, EJS, entre otros. (nubecolectiva, 2020)

Express es un framework web rápido y minimalista. Si has trabajado con la tecnología Node.js seguro que has oído hablar de él o lo has usado. Este framework está muy bien mantenido y lo usan montones de compañías en su entorno de producción, así que está bien comprobado.

Puedes empezar con Express instalándolo en tu proyecto usando npm.

(javfon1, 2019)

Express.js es un framework web rápido, minimalista para Node.js. Es simplemente una tecnología construida en Node.js que se comporta como un middleware para ayudar a administrar nuestros servidores y rutas. Considerando la naturaleza asincrónica de Node, la capacidad de crear una aplicación liviana que pueda procesar más de una sola solicitud sin problemas realmente depende de la capacidad de servicio de las tecnologías como Express.

(GodoFredo, 2018)

Express.js es definitivamente el rey de los frameworks node.js, alcanzará la increíble marca de 2 millones de descargas diarias a finales de 2019.

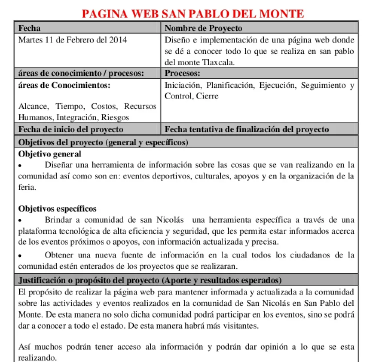
A pesar de ser un framework de trabajo tan antiguo, Express.js es mantenido activamente por la comunidad y es utilizado por grandes empresas como User, Mulesoft, IBM, etc. (softwareontheroad, 2019)

**Mapa mental.**

****

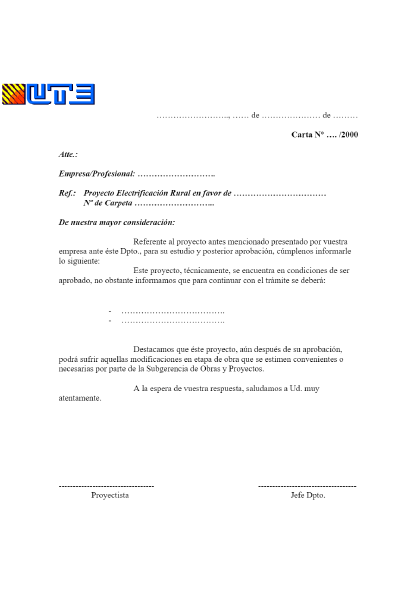
**Figura 6. Mapa mental. Fuente: Propia.**

En este ejemplo de acta de inicio de trabajo utilizamos la parte de arriba para su desarrollo. (Fany Peregrina, 2014)



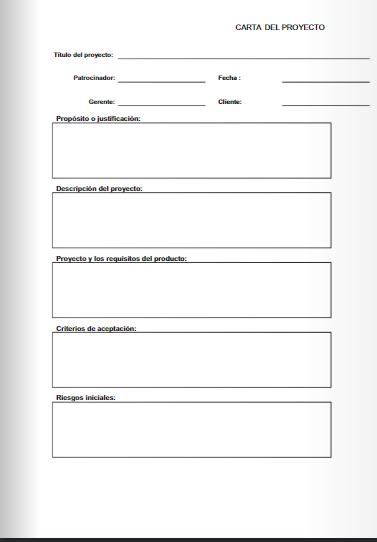
**Figura 7. Ejemplo acta de inicio. Fuente: https://es.slideshare.net/fanyperegrina/ejemplo-de-carta-de-proyectos.**

En este ejemplo de acta de inicio de trabajo utilizamos la parte de abajo donde esta la despedida y la firma para su desarrollo. (studylib, n.d.)



**Figura 8. Ejemplo acta de inicio. Fuente: https://studylib.es/doc/5344314/carta-de-aprobaci%C3%B3n-del-proyecto.**

En este ejemplo de acta de inicio de trabajo utilizamos la parte central para su desarrollo. (gelmer restrepo hernandez, n.d.)



**Figura 9. Ejemplo acta de inicio. Fuente: https://es.calameo.com/books/0004149205b649394a20c.**

* **Metodologías de desarrollo ágil.**

**1. Extreme Programming XP**

Esta herramienta es muy útil sobre todo para startups o empresas que están en proceso de consolidación, puesto que su principal objetivo es ayudar en las relaciones entre los empleados y clientes.

* Planificación del proyecto con el cliente
* Diseño del proyecto
* Codificación, donde los programadores trabajan en pareja para obtener resultados más eficientes y de calidad
* Pruebas para comprobar que funcionan los códigos que se van implementando

**2. SCRUM**

Esta metodología permite abordar proyectos complejos que exigen una flexibilidad y una rapidez esencial a la hora de ejecutar los resultados. La estrategia irá orientada a gestionar y normalizar los errores que se puedan producir en desarrollos demasiado largos, a través de, reuniones frecuentes para asegurar el cumplimiento de los objetivos establecidos.

* Se caracteriza por ser la “metodología del caos” que se basa en una estructura de desarrollo incremental,
* Posee análisis, desarrollo y testing.
* En la etapa de desarrollo encontramos lo que se conoce como interacciones del proceso o Sprint,
* Se realizan entregas regulares y parciales del producto final.

**3. Kanban**

La estrategia Kanban conocida como ‘Tarjeta Visual» muy útil para los responsables de proyectos. Esta consiste en la elaboración de un cuadro o diagrama en el que se reflejan tres columnas de tareas; pendientes, en proceso o terminadas.

* Planificación de tareas
* Mejora en el rendimiento de trabajo del equipo
* Métricas visuales
* Los plazos de entregas son continuos.

**4. Agile Inception**

Está orientada a la definición de los objetivos generales de las empresas. Su meta es clarificar cuestiones como el tipo de cliente objetivo, las propuestas de valor añadido, las formas de venta.

* Suele girar en torno al método de “elevator pitch”.
* Consiste en pequeñas reuniones.
* Su margen de falla es bajo.
* Es un método usado por muchas empresas.
* Su aplicación es posible en cualquier proyecto

(Sandra Garrido Sotomayor, 2022)

**Metodología elegida.**

La metodología que se escogió fue la de scrum, ya que lo que se necesita mucho el mejorar el rendimiento del proyecto y esta metodología acopla eso a nuestro proyecto, además de que también es una metodología que ya se ha implementado en otros proyectos realizados, y se conoce bien su alcance, así como su capacidad de uso.

**Tecnologías de desarrollo**

**Django:**

Django es un framework web diseñado para realizar aplicaciones de cualquier complejidad en unos tiempos muy razonables. Está escrito en Python y tiene una comunidad muy amplia, que está en continuo crecimiento. (Pablo Camino Bueno, 2018)

**Ventajas:**

* Cuenta con un sistema de autentificación de usuarios.
* Manejo de versiones que permite una distribución simple de actualizaciones.
* Ofrece un gran rendimiento y flexibilidad, pudiendo escalar proyectos de forma sencilla.
* Trabajar bajo un patrón MVC (Modelo Vista Controlador), lo que permite un desarrollo ágil y reutilizable.
* Incorpora una amplia variedad de paquetes de librerías (más de 4000).
* Dispone de una inmensa comunidad de usuarios en internet.
* Incluye opciones de protección para las aplicaciones, por ejemplo, contra ataques de SQL injection o ataques XSS (cross site scripting).
* Proporciona una estructura de código autogenerado.
* Cuenta con panel de administración para bases de datos.

(DEVOPS, 2020)

**Características:**

* **Seguridad:** Django tiene activados mecanismos incluidos para proteger tu base de datos, formularios y JavaScript.
* **Escalabilidad:** puedes utilizar el framework para un desarrollo sencillo, hasta uno mucho más complejo, ambos casos funcionarán de manera estable y con rapidez.
* **Interfaz:** Su interfaz para acceso a la base de datos y hacer consultas es sumamente buena.
* **Portable:** Al estar escrito en Python, se puede ejecutar en muchas plataformas como Windows, OS X, entre otras, dándole muchísima libertad al programador al momento de ejecutar las aplicaciones.

(Staff BEDU, 2020)

**Laravel:**

Laravel es el framework de aplicaciones web ideal para artesanos de la web por su elegante sintaxis. Es un código abierto y ofrece gratis un framework PHP. Laravel fue creado para aplicaciones web complejas que utilizan el patrón arquitectónico MVC o modelo-vista-controlador. Algunas de sus características incluyen OOP (programación orientada a objetos) o bibliotecas orientadas a objetos, un sistema de migración de bases de datos, motor de enrutamiento e interfaz de línea de comandos llamada "Artisan". (STARTECHUP, 2021)

**Ventajas:**

* Cuenta con un buen rendimiento.
* Alta seguridad.
* Autenticación poderosa.
* Comunidad de soporte dedicada y de código abierto.
* Plantillas de hoja.
* Migración de bases de datos.
* Bibliotecas orientadas a objetos.
* Arquitectura MVC.

(Pooja Sharma, 2019)

**Características:**

* Su motor de plantilla, llamado Blade, da numerosas posibilidades para hacer unas páginas visualmente muy potentes y eficaces, capaz de utilizar sus propias variables y reutilizarlas.
* Su motor de plantilla, llamado Blade, da numerosas posibilidades para hacer unas páginas visualmente muy potentes y eficaces, capaz de utilizar sus propias variables y reutilizarlas.
* Eloquent ORM, es muy intuitivo para escribir consultas en PHP sobre objetos.
* En seguridad, ofrece un nivel bastante fuerte con mecanismos de hash y salt para encriptar por medio de librerías como BCrypt, que también lo usa por ejemplo Zend Framework.
* Artisan, su sistema de comandos otorga al framework gran poder y a los programadores grandes facilidades y posibilidades, para crear controladores, entidades o actualizar la base de datos por ejemplo entre muchísimas cosas.
* Librerías y modularidad. Laravel aparte de sus propias librerías cuenta con ayuda de Symfony en otras muchas, otro MVC de los más usados en los últimos tiempos y con una gran comunidad detrás que hace que su avance y evolución sea muy significativo. También condiciona que la evolución de Laravel en parte dependa de Symfony en estos aspectos.
* Base de datos y migraciones. Permite actualizar y migrar la base de datos una vez que el desarrollo ya está comenzamos y hay cambios en el código conforme se requiera sin necesidad de borrarla y volverla a crear, gracias a esto el riesgo de perder datos sean del valor que sean es mínimo. Además, gracias a su Schema Builder hace que no requiera usar el SQL, cuenta con un sistema intuitivo en PHP para hacerlo más fácil.

(Rafael Altube Vera, 2021)

**Ruby on Rails:**

Este framework de backend web está escrito y basado en el lenguaje de programación Ruby. Utiliza los patrones de diseño DRY o Don't Repeat Yourself y MVC (modelo-vista-controlador). Es bastante similar a Django en cuanto a sus principios. Ruby on Rails aplica el principio de "convención sobre configuración" (o codificación por convención). Es decir, los programas se desarrollan en base a las convenciones típicas de programación y no a las definidas por el programador. (STARTECHUP, 2021)

**Ventajas:**

* Económico.
* Construido sobre la arquitectura Model-View-Controller (MVC).
* Facilita gestionar los cambios.
* Seguridad.
* Rendimiento.
* Productividad.
* Un gran repositorio de complementos gratuitos.
* Comunidad grande.

(Reclu IT, 2022)

**Características:**

* Es un lenguaje de propósito general, es decir, con Ruby se pueden desarrollar todo tipo de aplicaciones diferentes: aplicaciones de servicio web, clientes de correo electrónico, procesamiento de datos en Backend, aplicaciones de red, etc. Cualquier cosa que se te ocurra puedes hacerlo con Ruby.
* Es un lenguaje interpretado, es decir, no es compilado, se necesita que el intérprete de Ruby evalué el código y lo traduzca en lenguaje de máquina entendible por un ordenador, pero no existe un proceso previo de compilación como en C o Java.
* Es dinámico y flexible. Con estrategias como la meta programación (escribir programas que escriben o manipulan otros programas). Incluso se puede utilizar el propio lenguaje Ruby para alterar el código en tiempo de ejecución.
* Es de alto nivel, significa que está lejos del lenguaje de máquina y es muy cercano al inglés, a veces parecerá que simplemente se está escribiendo en inglés.
* Es de software libre (Open Source) y multiplataforma, se puede descargar totalmente gratis de la página oficial y ejecutarlo en diferentes sistemas operativos.

(Julio García, 2017)

**Spring Boot:**

Este framework de backend web está basado en Java y es de código abierto. Es ideal para aplicaciones independientes basadas en Spring y listas para la producción. Además, está pensado para aplicaciones a gran escala que utilizan la computación en la nube. Spring Boot viene con diferentes módulos para varios servicios. (STARTECHUP, 2021)

**Ventajas:**

* SpringBoot es gratis.
* SpringBoot funciona sobre la JVM.
* SpringBoot permite hacer Webs y Microservicios.
* SpringBoot te permite empezar a desarrollar prácticamente de inmediato y desplegar tu aplicación sin complicaciones.
* SpringBoot está en continua evolución.
* SpringBoot tiene un gran número de programadores.

(arteco-consulting, 2020)

**Características:**

* Contenedores Java embebidos: Tomcat o Jetty
* Soporte para la automatización con Maven y Gradle
* Configuración sugerida para iniciar rápidamente con un proyecto (Starters)
* Configura automáticamente Spring, cuando sea posible
* Características listas para producción: métricas, seguridad, verificación del estatus, externalización de configuración, etc.
* No genera código y no requiere configuración XML.

(Jorge Aguilar, 2017)

* **Arquitectura PWA**
  + Investiga los elementos de la arquitectura para PWA

**MANIFEST**

Es un archivo JSON que define el nombre de la aplicación, la ubicación de los íconos que serán utilizados en diferentes resoluciones, la URL inicial, el color de fondo por defecto, entre otras configuraciones. Estas opciones le permitirán a la plataforma o dispositivo, saber cómo se llamará y comportará nuestra aplicación si fuera instalada. (Dragon Nomada, 2021)

**Ejemplo**

El manifest es un simple archivo JSON que incorpora la *PWA*, con unos campos tal que estos:

****

**Figura 10. MANIFEST. Fuente: https://medium.com/react-adventure/arquitectura-de-una-pwa-26e4c4a58da2.**



**Figura 11. MANIFEST. Fuente: https://medium.com/react-adventure/arquitectura-de-una-pwa-26e4c4a58da2.**

**SERVICES WORKERS**

Son archivos JS que se ejecutan de forma separada con respecto a tu página web y gestionan eventos lanzados bien por el navegador, bien por tu propia web. No necesitan que tu web esté abierta y no tienen ningún template (o DOM) que mostrar. Los service workers se sitúan entre el cliente y la red, actuando como un proxy en el lado del cliente, permiten interceptar las peticiones web de tu página, con lo que se pueden cachear las llamadas localmente. (Enrique Oriol, 2016)

**Ejemplo**

Si el navegador soporta service workers, lo único que tienes que hacer es llamar a su método register con el path al archivo javascript donde está definido. Se hace así:

****

**Figura 12. SERVICES WORKERS. Fuente: http://blog.enriqueoriol.com/2016/12/progressive-web-apps.html#:~:text=Arquitectura%20de%20una%20PWA&text=App%20Shell%3A%20Los%20archivos%20HTML,cat%C3%A1logo)%20o%20no%20es%20imprescindible.**

**.**

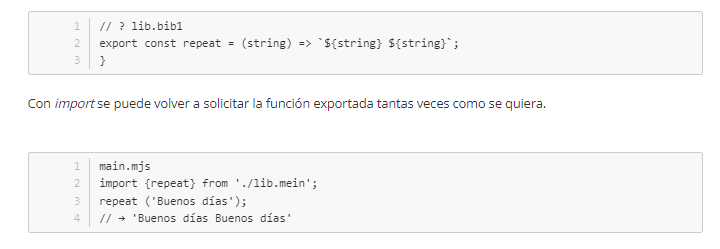
****

**Figura 13. SERVICES WORKERS. Fuente: http://blog.enriqueoriol.com/2016/12/progressive-web-apps.html#:~:text=Arquitectura%20de%20una%20PWA&text=App%20Shell%3A%20Los%20archivos%20HTML,cat%C3%A1logo)%20o%20no%20es%20imprescindible.**

**WEB COMPONENTS**

Los componentes web son una adición relativamente nueva a la plataforma web. Se basan en tres especificaciones diferentes: La API de elementos personalizados le permite crear nuevos elementos HTML. Puede interactuar con ellos configurando atributos DOM o propiedades de JavaScript, y los componentes pueden comunicarse con el mundo exterior mediante el envío de eventos (personalizados). Si está utilizando Angular, puede conocer este principio de Entradas y salidas. Shadow DOM es una parte encapsulada del DOM que per se no se ve afectada por las definiciones de estilo externo (aunque hay algunas excepciones, como el color del texto o las propiedades personalizadas de CSS), lo que le permite escribir componentes aislados. (Christian Liebel, 2021)

**Ejemplo**

****

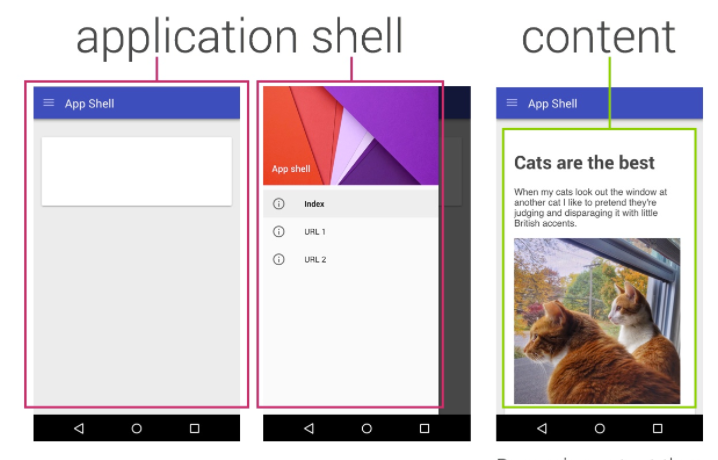
**Figura 14. WEB COMPONENTS. Fuente: https://www.ionos.mx/digitalguide/paginas-web/desarrollo-web/web-components/.**

**APP SHELLS**

La app shell no es otra cosa que el esqueleto o estructura básica de la aplicación, aquella que permite activar la interfaz de usuario y en la que se cargará el contenido.

Una vez abrimos una PWA, este paquete básico se almacenará en la memoria caché del navegador y utilizará un service worker para que esta funcione. A partir de entonces, como ya tendremos los componentes necesarios para poner en marcha la app, podremos volver a abrir la progressive web app de forma rápida y sin necesidad de tener una conexión a internet en las siguientes visitas, asegurando un mayor rendimiento y fiabilidad. (Enrique Oriol, 2016)

**Ejemplo**

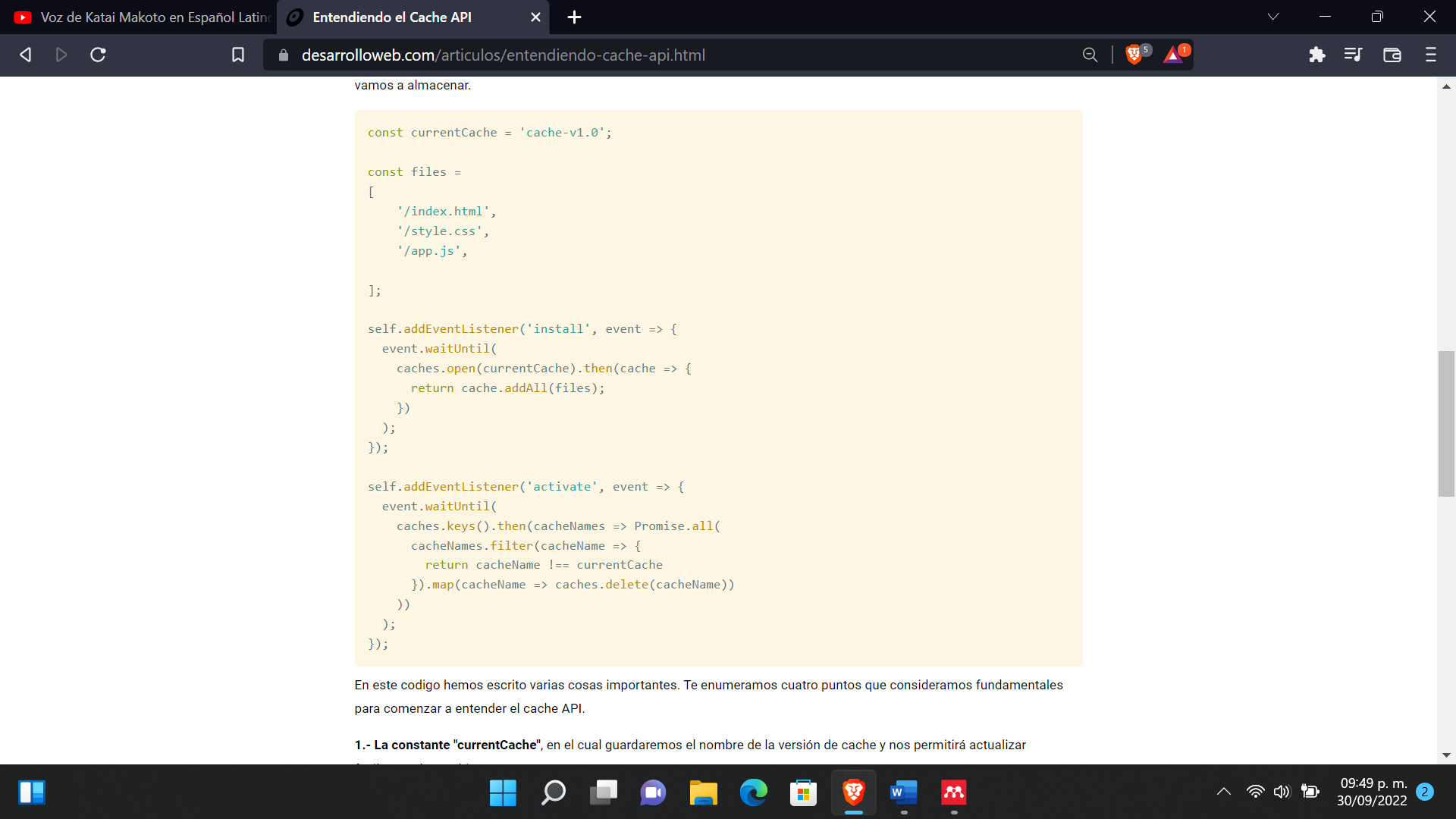
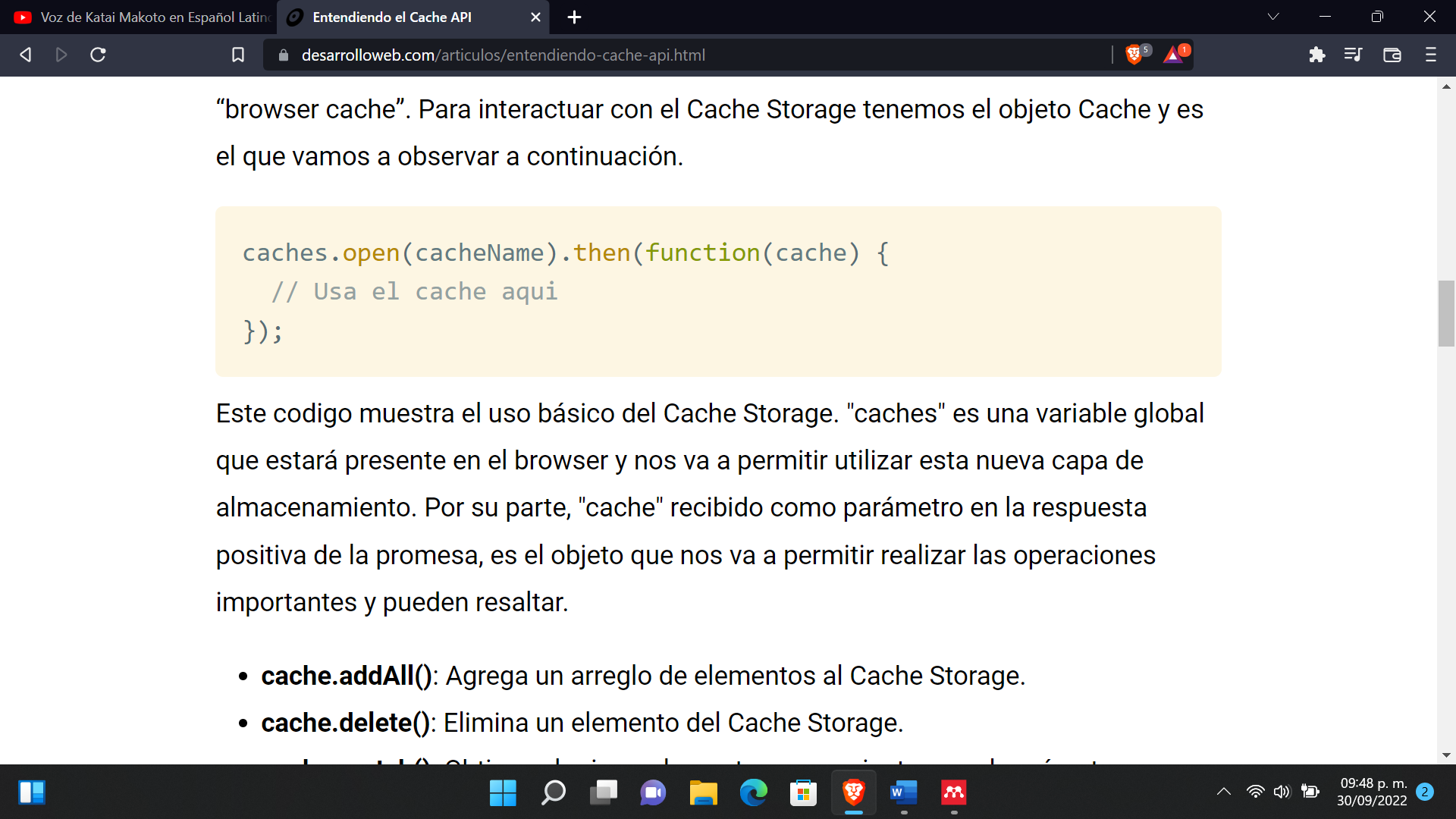
****

**Figura 15. APP SHELLS. Fuente: http://blog.enriqueoriol.com/2016/12/progressive-web-apps.html#:~:text=Arquitectura%20de%20una%20PWA&text=App%20Shell%3A%20Los%20archivos%20HTML,cat%C3%A1logo)%20o%20no%20es%20imprescindible.**

**Cache API**

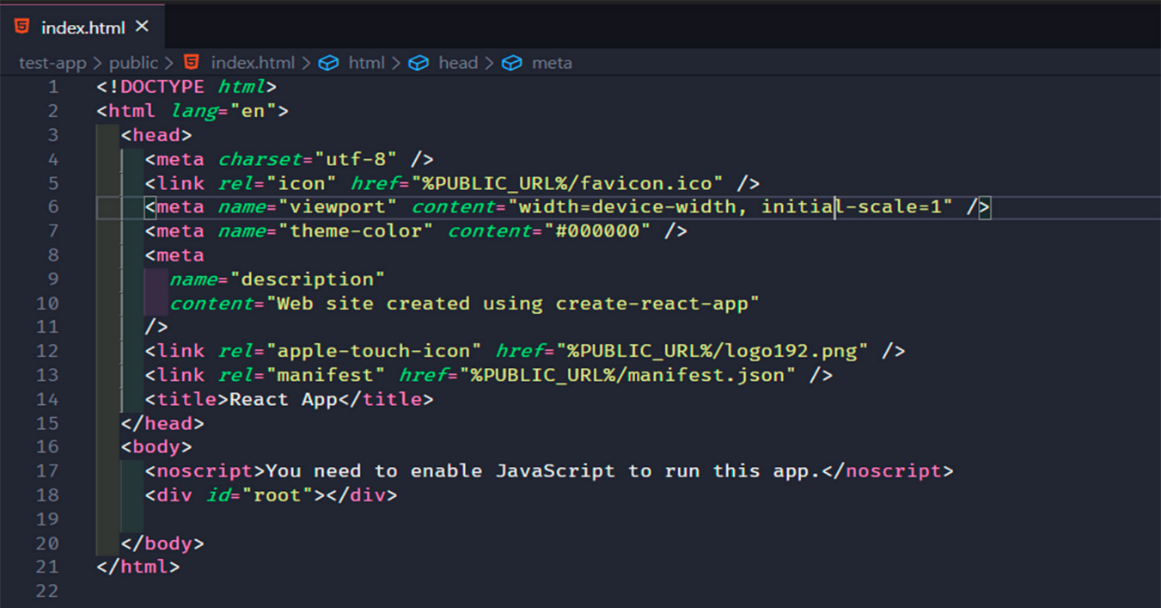
Cache API y el CacheStorage API, que nos permite almacenar los archivos importantes de manera local y responder con el “App Shell” en todas las ocasiones, logrando que nuestros usuarios no estén esperando a que se cargue web cuando tengan una conexión inestable.

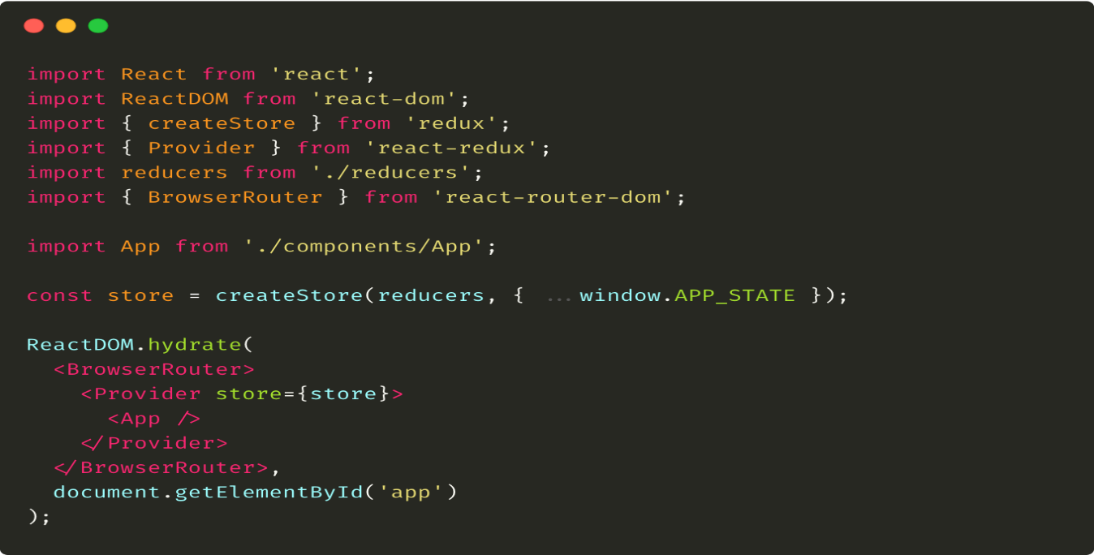
El Cache Storage es una nueva capa de almacenamiento y no se debe confundir con el “browser cache”. Para interactuar con el Cache Storage tenemos el objeto Cache y es el que vamos a observar a continuación.(Carlos Rojas, 2017)

**Ejemplo**

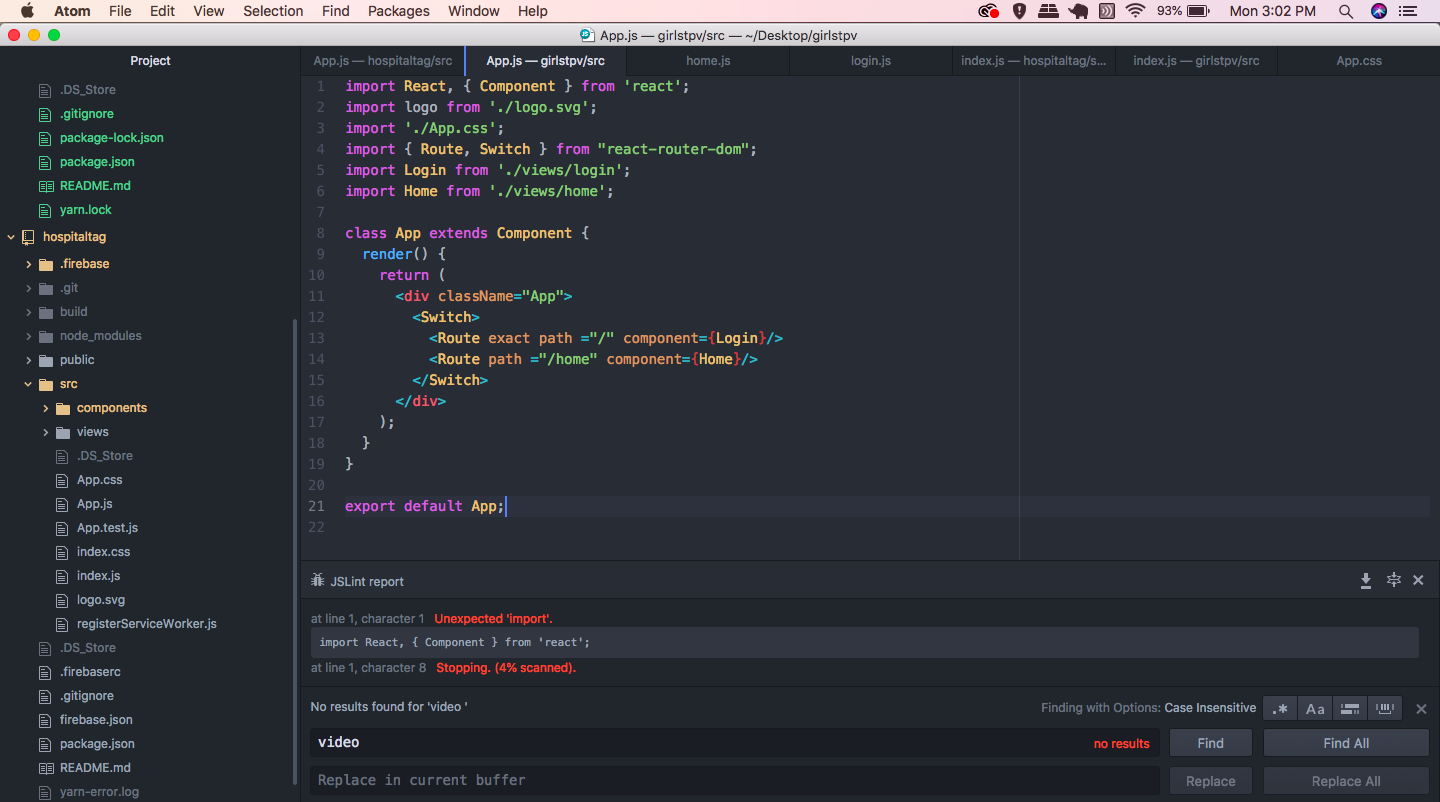
**Figura 16. Ejemplo Appi. Fuente: https://desarrolloweb.com/articulos/entendiendo-cache-api.html.**

* + **Investiga arquitecturas para PWA (3)**
    - **Del lado del servidor SSR**

La representación del lado del servidor (SSR) significa que un sitio web se representa en el servidor, por lo que ofrece una primera carga más rápida, pero navegar entre páginas requiere descargar contenido HTML nuevo. (developer mozilla, n.d.-a)

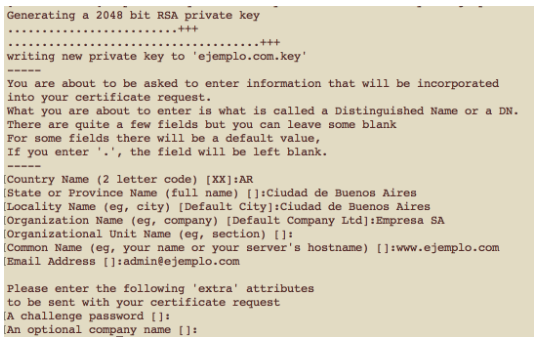
**Figura 17. Ejemplo de SSR. Fuente:https://pwaexperts.io/blog/server-side-rendering-pwa**

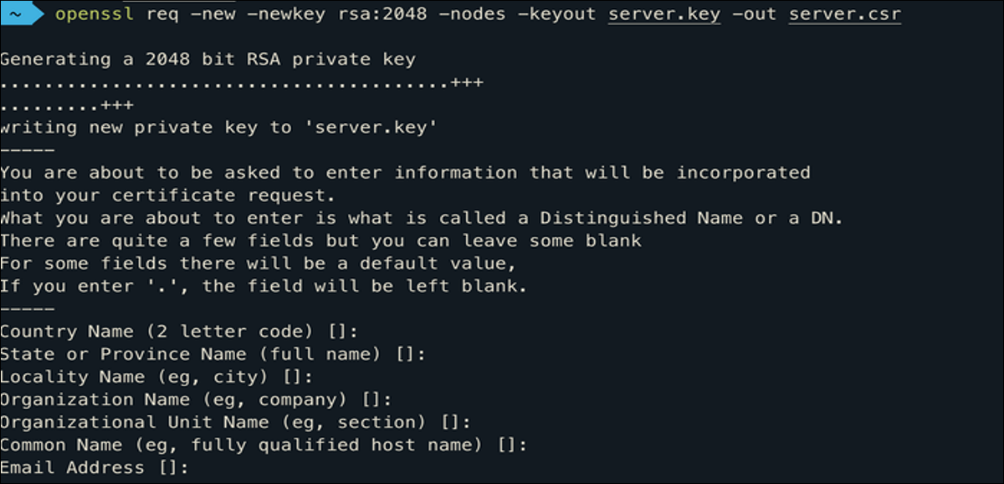
**Figura 18. Ejemplo de SSR. Fuente: https://javascript.plainenglish.io/server-side-rendering-in-react-redux-ab0af31c9c4b?gi=e1a5c64f16bb**



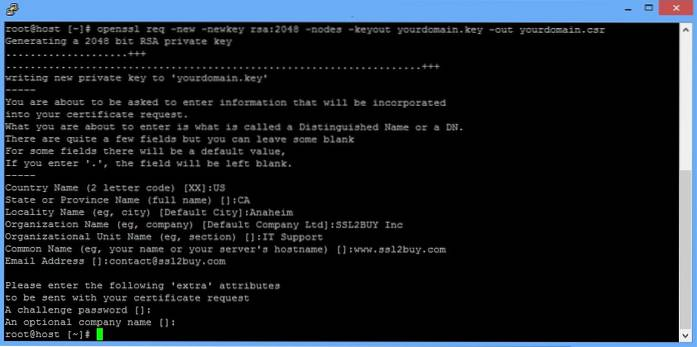
**Figura 19. Ejemplo de SSR. Fuente: https://shop53002.lemaprod.com/category?name=react%20router%20default%20route**

* + - **Del lado del cliente CSR**

La representación de lado del cliente (CSR) permite que el sitio web se actualice en el navegador casi instantáneamente cuando se navega a diferentes páginas, pero requiere más de un golpe de descarga inicial y una representación adicional en el cliente al principio. El sitio web es más lento en una visita inicial, pero puede ser más rápido para navegar.

**Figura 20. Ejemplo de CSR. Fuente: https://www.intermediasp.com/soporte/articulo/apache-server-openssl/**

**Figura 21. Ejemplo de CSR. Fuente: https://systempeaker.com/linux/que-son-los-archivos-csr-y-como-se-crean/**



**Figura 22. Ejemplo de CSR. Fuente: https://www.linuxteaching.com/article/openssl\_generate\_csr**

**JAVASCRIPT**

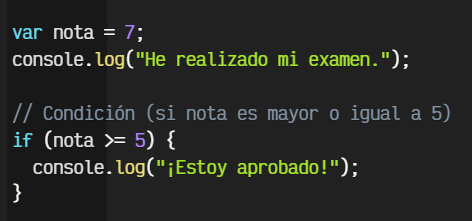
**ESTRUCTURAS DE CONTROL**

Una estructura de control es una herramienta que tienen los lenguajes de programación y que se utiliza para controlar el flujo de las operaciones durante el programa. (salamarkesa, n.d.)

**Ejemplo**

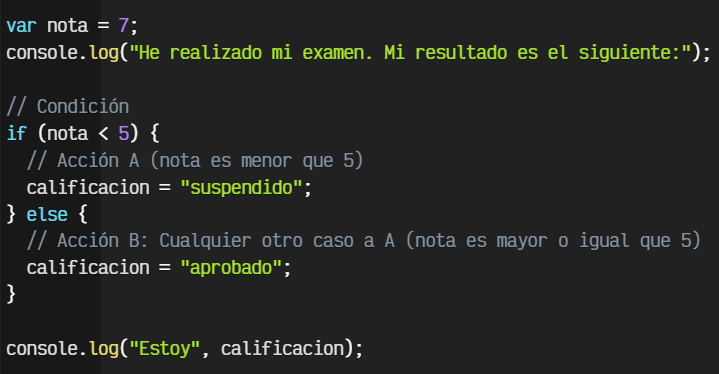
|  |  |
| --- | --- |
| Estructura de control | Descripción |
| If | **Condición simple**: Si ocurre algo, haz lo siguiente... |
| If/else | **Condición con alternativa**: Si ocurre algo, haz esto, si no, haz lo esto otro... |
| ?: | **Operador ternario**: Equivalente a **If/else**, método abreviado. |
| Switch | **Estructura para casos específicos**: Similar a varios **If/else** anidados. |

**Condicional IF**

****

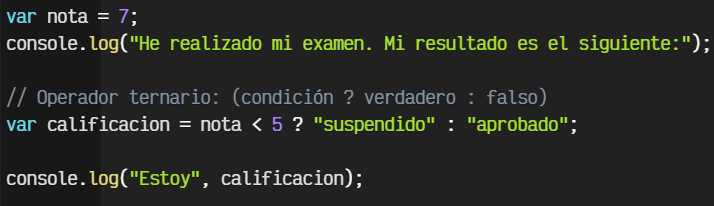
**Figura 23. JavaScript Condición IF. Fuente: https://www.salamarkesa.com/estructuras-control-javascript-explicacion-ejemplos/.**

**Condicional If else**

****

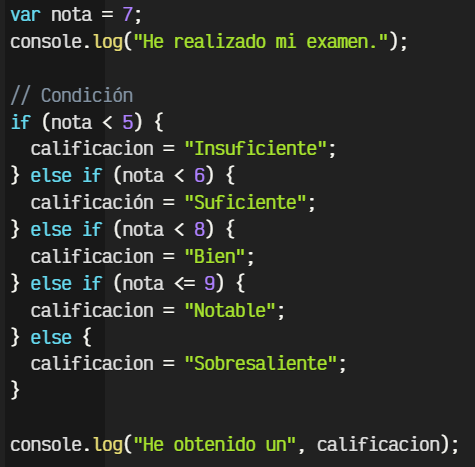
**Figura 24. JavaScript Condicional If else. Fuente: https://www.salamarkesa.com/estructuras-control-javascript-explicacion-ejemplos/.**

**Operador ternario**

****

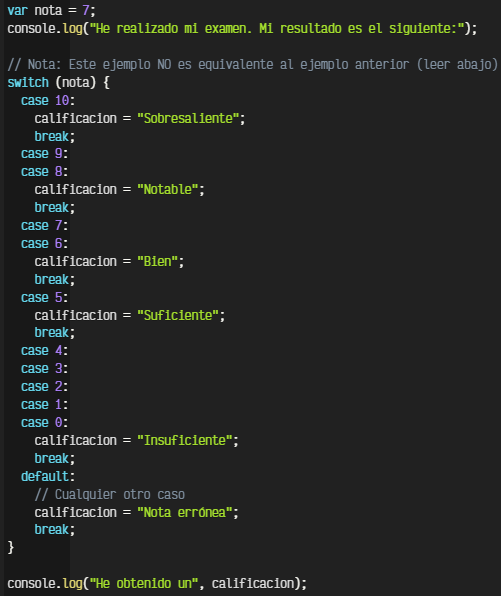
**Figura 25. JavaScript Operador ternario. Fuente: https://www.salamarkesa.com/estructuras-control-javascript-explicacion-ejemplos/.**

**Condicional If múltiple**

****

**Figura 26. JavaScript** **Condicional If múltiple. Fuente: https://www.salamarkesa.com/estructuras-control-javascript-explicacion-ejemplos/.**

**Condicional switch**

****

**Figura 27. JavaScript Condicional switch. Fuente: https://www.salamarkesa.com/estructuras-control-javascript-explicacion-ejemplos/.**

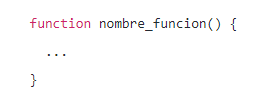
**FUNCIONES**

Las funciones son uno de los bloques de construcción fundamentales en JavaScript. Una función en JavaScript es similar a un procedimiento un conjunto de instrucciones que realiza una tarea o calcula un valor, pero para que un procedimiento califique como función, debe tomar alguna entrada y devolver una salida donde hay alguna relación obvia entre la entrada y la salida. (developer mozilla, n.d.-b)

**Ejemplo**

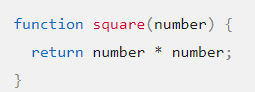
(uniwebsidad, n.d.)

**Las funciones en JavaScript se definen mediante la palabra reservada function, seguida del nombre de la función. Su definición formal es la siguiente:**

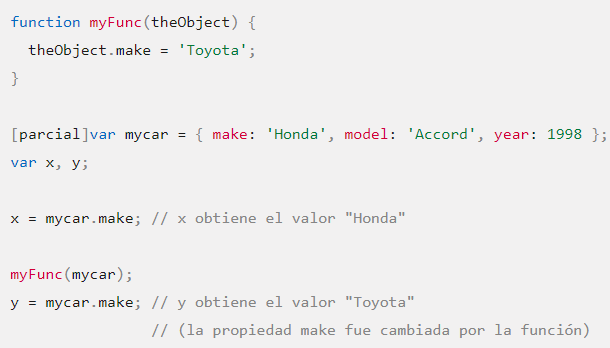
****

**Figura 28.** **JavaScript ejemplo funciones. Fuente: https://uniwebsidad.com/libros/javascript/capitulo-4/funciones.**

**El siguiente código define una función simple llamada square ("cuadrado"):**

****

**Figura 29. JavaScript ejemplo funciones. Fuente: https://uniwebsidad.com/libros/javascript/capitulo-4/funciones.**

****

**Figura 30. JavaScript ejemplo funciones. Fuente: https://uniwebsidad.com/libros/javascript/capitulo-4/funciones.**

**EVENTOS**

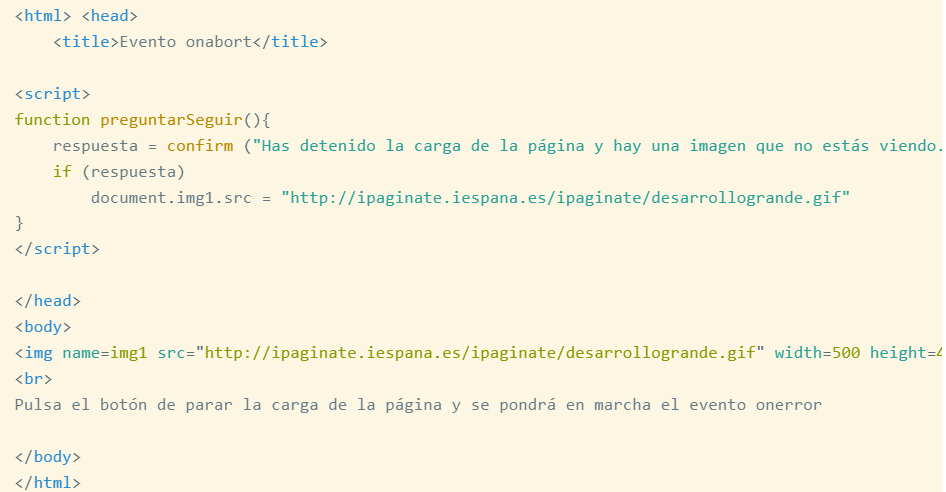
JavaScript permite realizar scripts con ambos métodos de programación: secuencial y basada en eventos. Los eventos de JavaScript permiten la interacción entre las aplicaciones JavaScript y los usuarios. Cada vez que se pulsa un botón, se produce un evento. Cada vez que se pulsa una tecla, también se produce un evento. (arkaitzgarro, n.d.)

|  |  |  |
| --- | --- | --- |
| Evento | Descripción | Elementos para los que está definido |
| onblur | Un elemento pierde el foco | <button>, <input>, <label>, <select>, <textarea>,  <body> |
| onchange | Un elemento ha sido modificado | <input>, <select>, <textarea> |
| onclick | Pulsar y soltar el ratón | Todos los elementos |
| ondblclick | Pulsar dos veces seguidas con el ratón | Todos los elementos |
| onfocus | Un elemento obtiene el foco | <button>, <input>, <label>, <select>, <textarea>,  <body> |
| onkeydown | Pulsar una tecla y no soltarla | Elementos de formulario y <body> |
| onkeypress | Pulsar una tecla | Elementos de formulario y <body> |
| onkeyup | Soltar una tecla pulsada | Elementos de formulario y <body> |
| onload | Página cargada completamente | <body> |
| onmousedown | Pulsar un botón del ratón y no soltarlo | Todos los elementos |
| onmousemove | Mover el ratón | Todos los elementos |
| onmouseout | El ratón "sale" del elemento | Todos los elementos |
| onmouseover | El ratón "entra" en el elemento | Todos los elementos |
| onmouseup | Soltar el botón del ratón | Todos los elementos |
| onreset | Inicializar el formulario | <form> |
| onresize | Modificar el tamaño de la ventana | <body> |
| onselect | Seleccionar un texto | <input>, <textarea> |
| onsubmit | Enviar el formulario | <form> |
| onunload | Se abandona la página, por ejemplo, al cerrar el navegador | <body> |

(Miguel Angel Alvarez, 2021)

**Ejemplo**

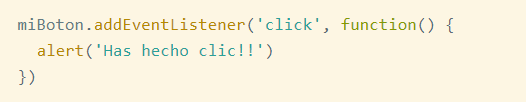
**Evento onabort**

****

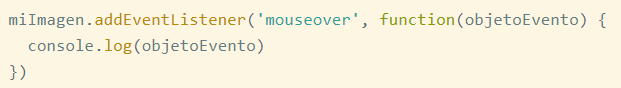
**Figura 31. Evento onabort. Fuente: https://desarrolloweb.com/articulos/1246.php.**

(Miguel Angel Alvarez, 2003)

**Dependiendo del evento que se utilizara es el que se mandara a llamar, como se muestra a continuación**

****

**Figura 32. Evento onabort. Fuente: https://desarrolloweb.com/articulos/1246.php.**

****

**Figura 33. Evento onabort. Fuente: https://desarrolloweb.com/articulos/1235.php.**

(Miguel Angel Alvarez, 2021)

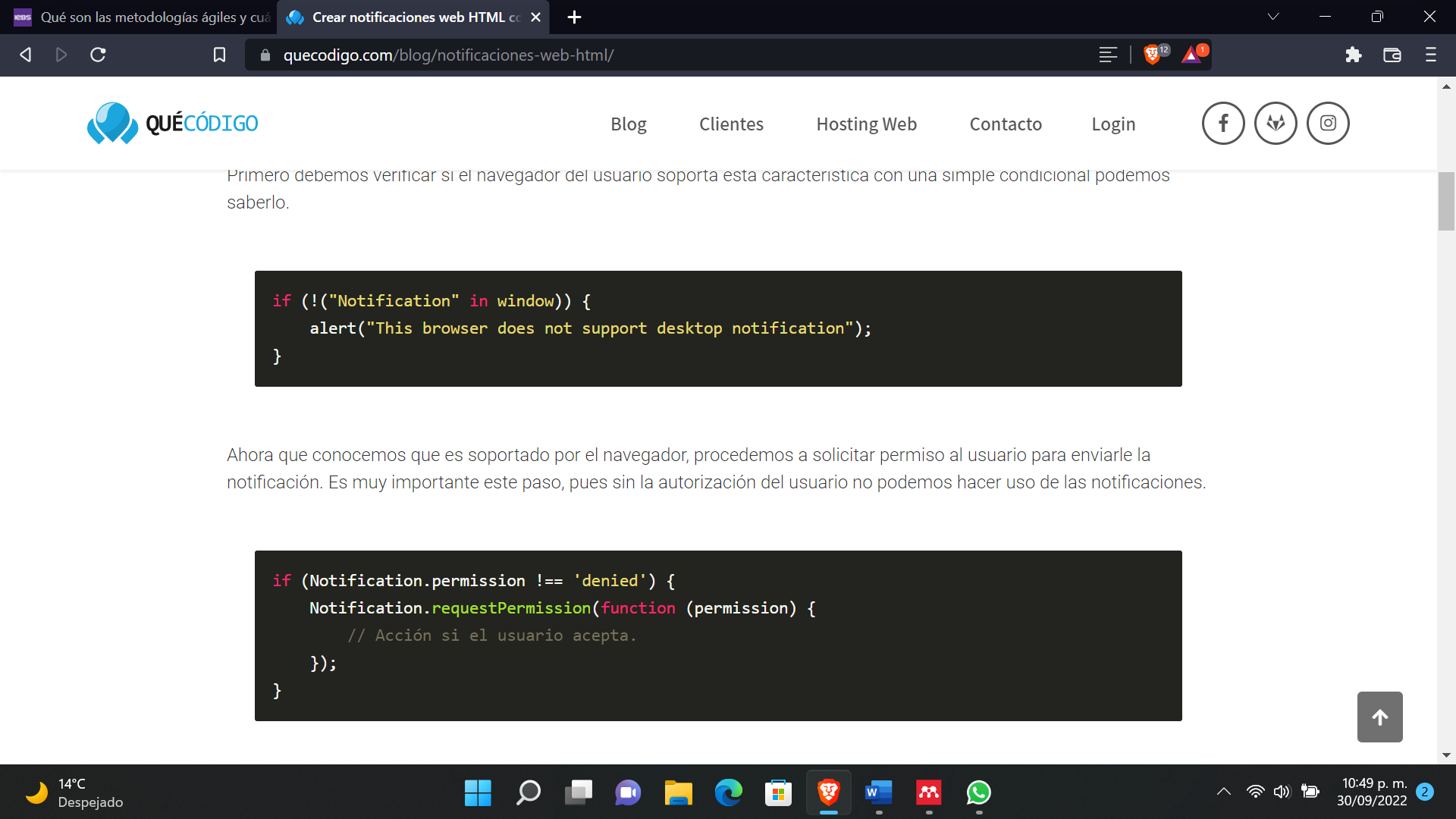
**NOTIFICACIONES**

Notificaciones de escritorio, últimamente ya toda aplicación tiene estas notificaciones de escritorio, más que todo las vemos en la aplicación de WhatsApp Web.

Para implementar esta API en nuestro sitio web, primero tenemos que conocer cuales navegadores soportan esta característica y en cuales versiones del navegador podemos usarlo. En este sentido contamos con Caniuse.

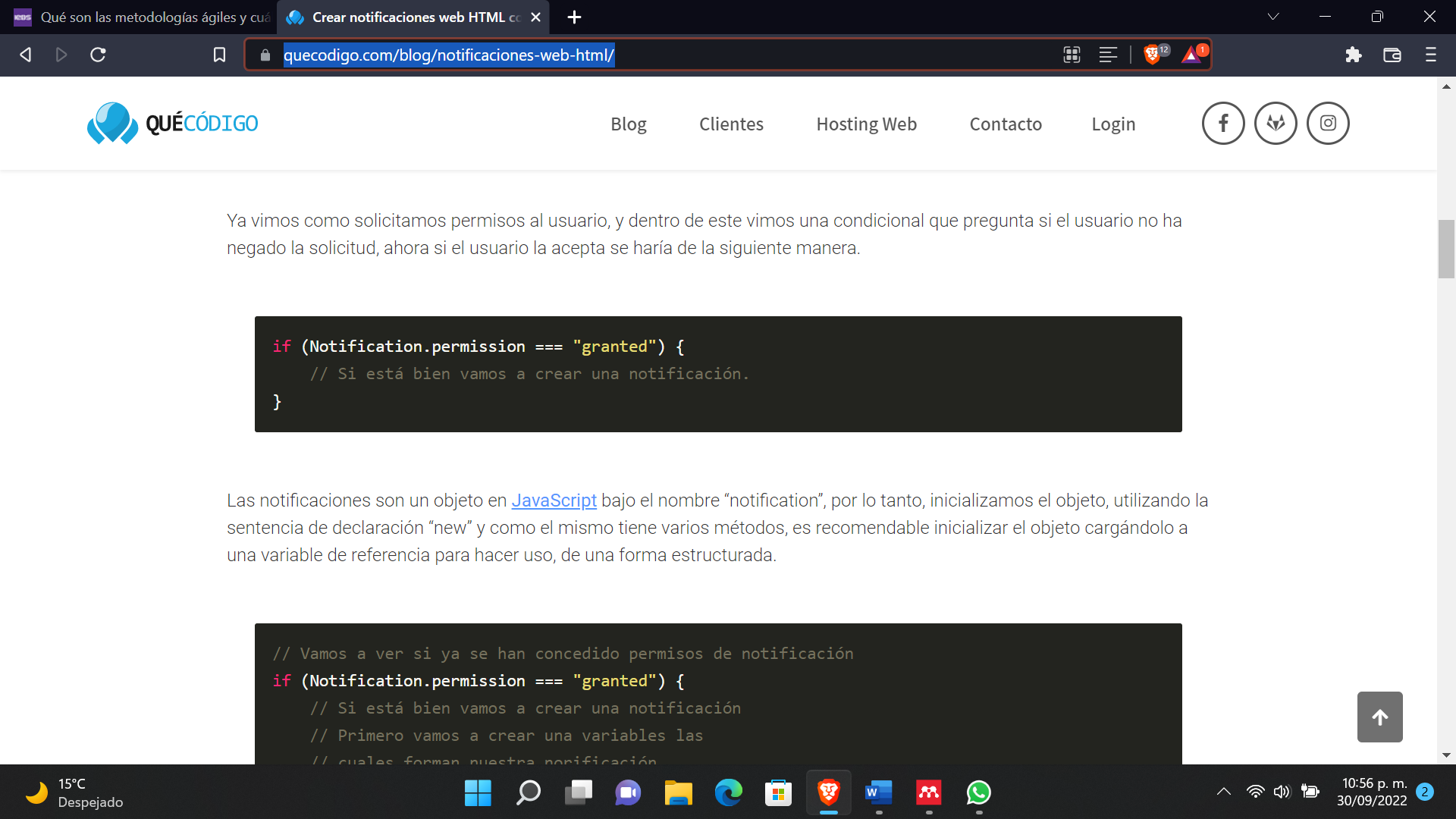
Al ser esta característica propia de JavaScript no necesitamos incluir ninguna librería. Ahora vamos a conocer cómo implementarlo con pocas líneas de código.

(Edinson Tique, 2020)

**Ejemplo**

Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamente**Figura 34. Notificaciones Java. Fuente: https://www.quecodigo.com/blog/notificaciones-web-html/**

**Figura 35. Notificaciones Java. Fuente: https://www.quecodigo.com/blog/notificaciones-web-html/**

**Figura 36. Notificaciones Java. Fuente: https://www.quecodigo.com/blog/notificaciones-web-html/**

(Edinson Tique, 2020)

**Ciclo de vida de una PWA**

* + Investiga los siguientes conceptos de Ciclo de vida de una PWA, haciendo énfasis en:

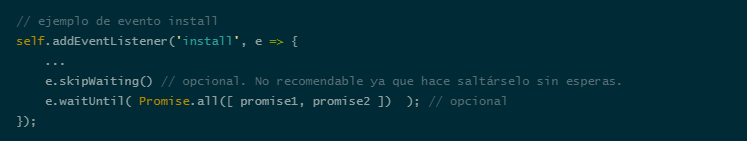
1. **Install**

El primer evento que recibe un service worker es install. se va a disparar cada la primera vez, y cada vez que haya cambios en ese listener. Pero aún no se activa.

Es el lugar adecuado para almacenar en caché todo lo que necesitas para poder controlar los clientes. La promesa que pasas a event.waitUntil() permite que el navegador sepa que la instalación se completó correctamente.

Si se rechaza la promesa, significa que no se completó la instalación y el navegador elimina el service worker.

**Ejemplo**



**Figura 37.Ciclo de vida de pwa.**

<https://www.jolugama.com/blog/2021/05/27/PWA-aplicaciones-web-progresivas/>

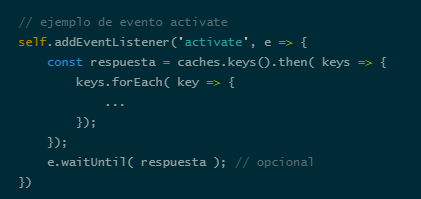
1. **Activate**

Cuando el Service Worker se instala y entra en el estado activado, se activará el evento de activación. Puede hacer un preprocesamiento escuchando el evento de activación, como actualizar la versión anterior y borrar el caché inútil.

Una vez que tu service worker esté listo para controlar clientes y administrar eventos funcionales como push y sync, recibirás un evento activate. Sin embargo, eso no significa que se controlará la página desde la que se realizó la llamada a .register().

La primera vez que cargas la versión, no se procesa la solicitud. La configuración predeterminada es consistencia: si tu página se carga sin un service worker, tampoco lo harán los subrecursos. Si cargas la versión otra vez (en otras palabras, si actualizas la página), se controlará la solicitud. Tanto la página como la imagen atravesarán eventos fetch.

**Ejemplo**

****

**Figura 38. Ciclo de vida de pwa 2.**

<https://programmerclick.com/article/8606135720/>

1. **Sync**

La sincronización de fondo emite el evento de sincronización. La sincronización en segundo plano es una API lanzada por Google en cooperación con SW, y se utiliza para proporcionar un método para que los trabajadores de servicios implementen la sincronización de registro y monitoreo.

El evento Sync permite retrasar las tareas de la red hasta que el usuario se conecte a la red. La función que implementa a menudo se denomina sincronización en segundo plano. Esto es muy útil en el modo fuera de línea para garantizar que cualquier tarea dependiente de la red iniciada por el usuario eventualmente logre el propósito deseado cuando la red vuelva a estar disponible.

<https://programmerclick.com/article/8606135720/>

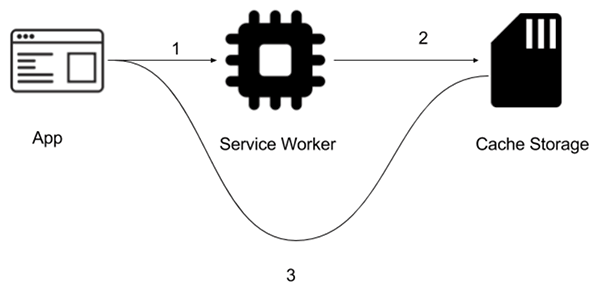
1. **Push**

El evento push está preparado para push. A través de la API PUSH, después de suscribirse al servicio push, puede usar el método push para despertar al Service Worker en respuesta a los mensajes del servicio de mensajería del sistema, incluso si el usuario ha cerrado la página.

<https://programmerclick.com/article/8606135720/>

* **Estrategias Cache y Modo Offline**
  + Investiga las estrategias Cache y Modo Offline, haciendo énfasis en:
    - **Cache Only**

Esta estrategia depende directamente de la caché. Si hay archivos en caché, se entregarán, este patrón responde a todas las peticiones con archivos de Cache Storage. Si no lo encuentra las solicitudes fallarán.



**Figura 39. Ejemplo de cache Only.**

Su implementación sería así:

self.addEventListener('fetch', function(event) {

event.respondWith(caches.match(event.request));

});

<https://desarrolloweb.com/articulos/estrategias-offline-pwa.html>

* + - **Cache con fallback**

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteWorkbox utilizando el fallback mantiene cachés independientes para el almacenamiento en caché previo y los cachés en tiempo de ejecución, y puede haber situaciones en las que desee almacenar algo en caché con anticipación sin depender del almacenamiento en caché previo, ya que las actualizaciones del manifiesto de caché previo requerirán que implemente un trabajador de servicio actualizado.

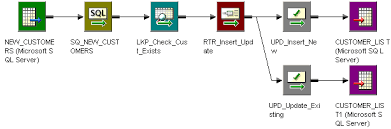
**Figura 40. Ejemplo de fallback.**

(Chrome, 2021)

* + - **Cache dinámica**

El almacenamiento en caché dinámico almacena toda la salida HTML de tu aplicación web. Por ejemplo, una página generada por WordPress consta de una parte HTML generada dinámicamente que se almacena en caché mediante la caché dinámica y una serie de recursos estáticos: JS, CSS, imágenes y otros medios.

Sin almacenamiento en caché, el servicio PHP proporciona una visita a tu página, que procesa todo el código de WordPress, temas y plugins, realiza una serie de consultas al servidor de base de datos MySQL y luego produce el resultado final. Aunque estos servicios son increíblemente rápidos, se necesita tiempo para obtener todos los datos y procesarlos. La caché dinámica almacena el resultado final de todo este trabajo y lo entrega directamente al siguiente cliente. Esto reduce en gran medida el TTFB (tiempo hasta el primer byte) de tu web y puede reducir hasta más de 10 veces el tiempo de carga de tus páginas web.

****

**Figura 41. Ejemplo de cache dinamica.**

(*Siteground*, 2022)

1. **METODOLOGÍA**

Definición propia: Una PWA la podemos definir como una evolución natural de las páginas web normales, las cuales no necesitamos una cuenta de desarrollador para publicar la pwa, por lo que no son aplicaciones nativas, y son seguras si cuentan con certificado SSL comenzando su url con https://

Podemos acceder a ella sin necesidad de estar conectados a internet por lo que en la primera sesión esta descarga el contenido que requiere y a la siguiente vez se puede acceder sin problemas, aunque no este conectada a internet. En las pwa podemos acceder a través de un enlace o incluso con un código QR.

**Indica cuál framework para front-end utilizarás.**

Se utilizará el framework de react porque es unos de los mas populares en el desarrollo de aplicaciones en PWA lo que hace que tengamos más información que podamos utilizar el desarrollo de sitios web progresivos y también altamente funcionales porque su biblioteca extensa es de JavaScript y JSX que es solo para react con el que podremos vincularnos con las estructuras HTML.

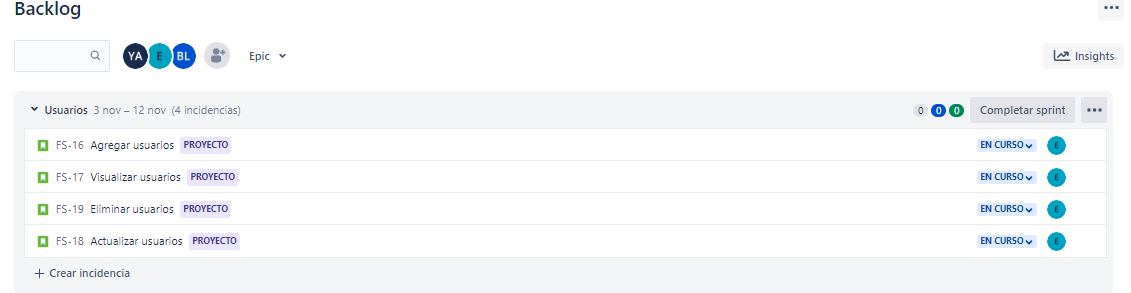
**Indica cuál framework para back-end utilizarás.**

Se utilizará el framework de laravel porque está basado en PHP aparte de que sus migraciones a la base de datos son fáciles de hacer por medio de un comando, laravel también cuenta con una comunidad grande que nos proporciona más información sobre esta.

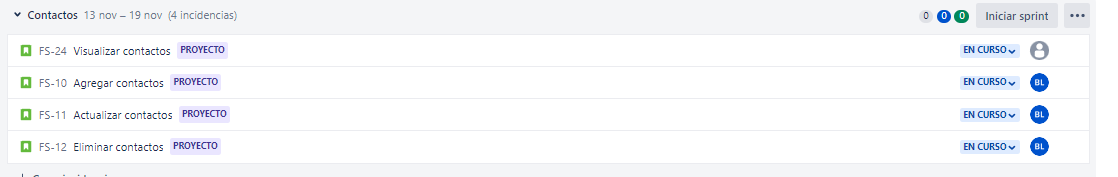
1. **RESULTADOS**



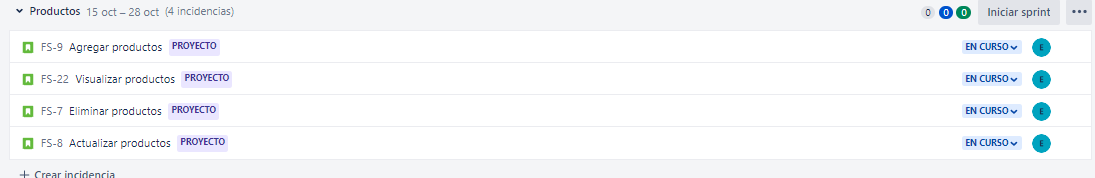
**Figura 42. Backlog Farmacia sana sana. Fuente: propia.**

****

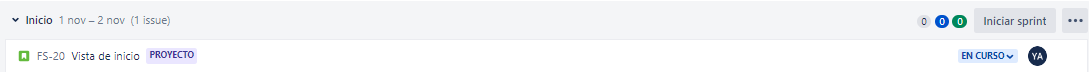
**Figura 43. Backlog usuarios. Fuente: propia.**

****

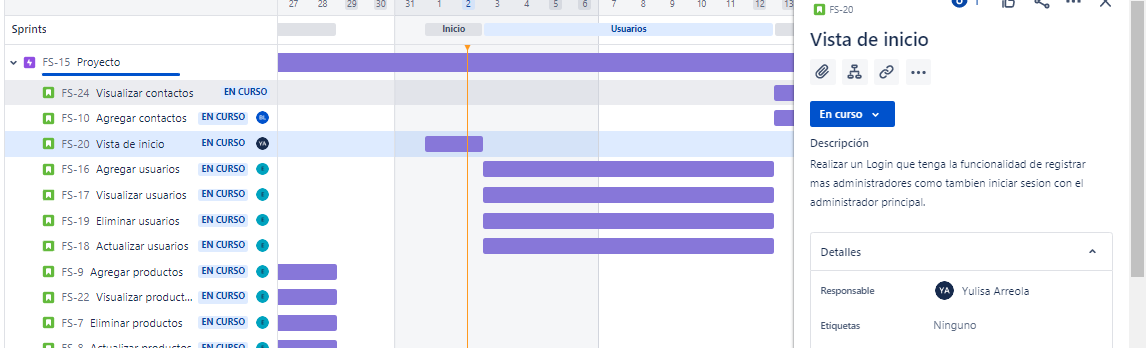
**Figura 44. Backlog contactos. Fuente: propia.**

****

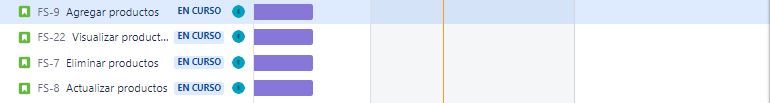
**Figura 45. Backlog productos. Fuente: propia.**

****

**Figura 46. Backlog inicio. Fuente: propia.**

****

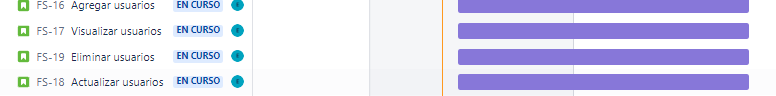
**Figura 47. Sprints inicio. Fuente: propia.**

****

**Figura 48. Sprints productos. Fuente: propia.**

****

**Figura 49. Sprints contactos. Fuente: propia.**

****

**Figura 50. Sprints usuarios. Fuente: propia.**



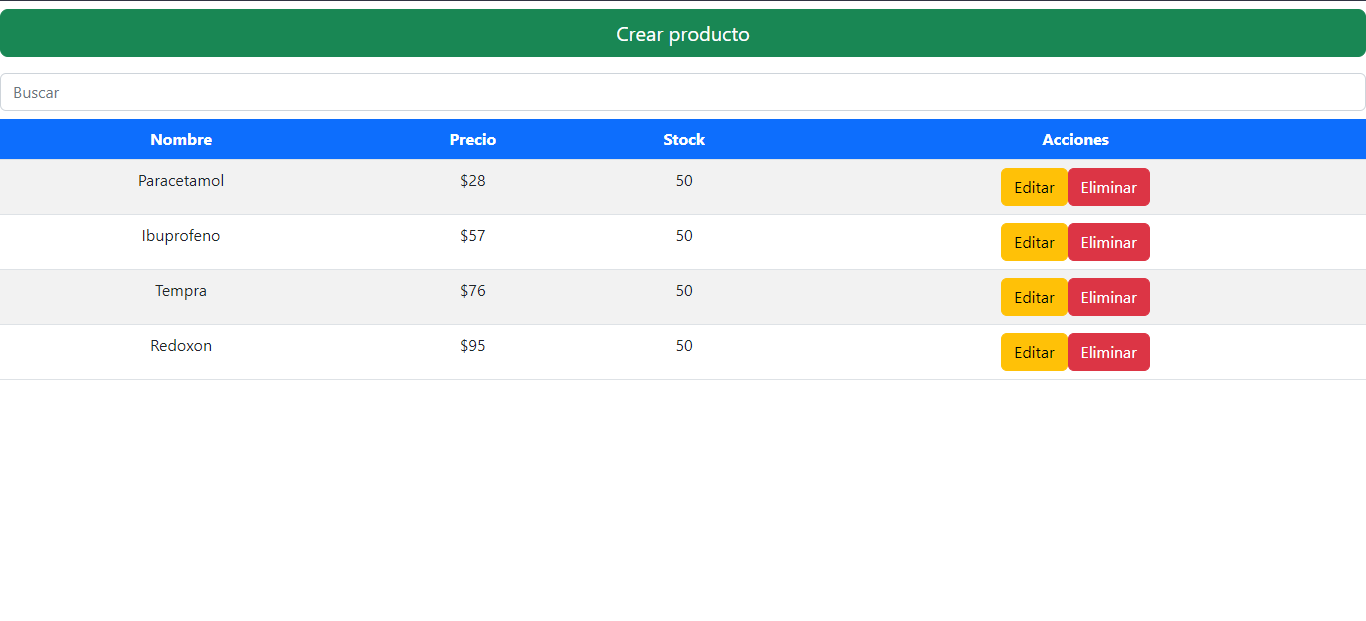
**Figura 51. Definición de costos. Fuente: propia.**

**Figura 52. Matriz de comunicaciones. Fuente: propia.**

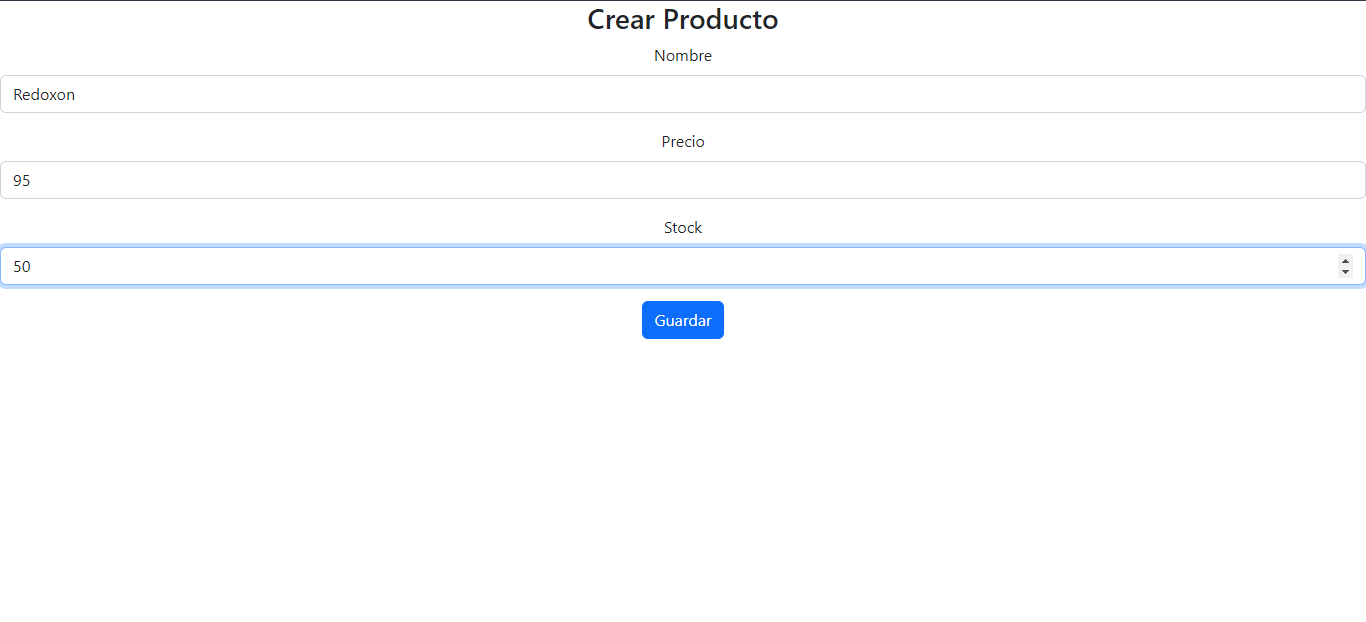


**Figura 53. Stakeholders. Fuente: propia.**

**VISTA DEL CRUD PRODUCTOS**

****

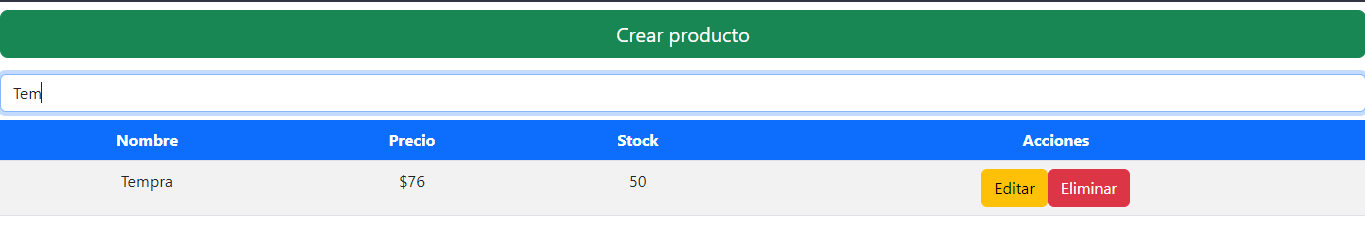
**Figura 54. VISTA DEL CRUD PRODUCTOS. Fuente: propia.**

****

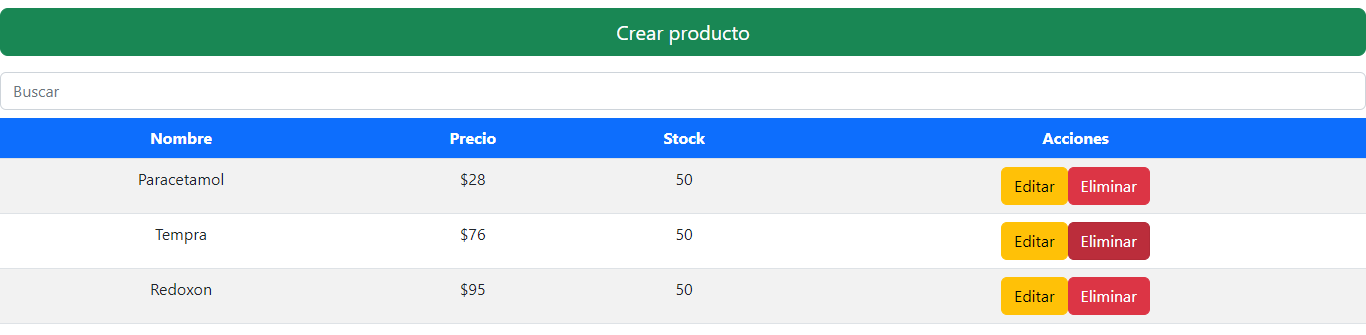
**Figura 55. AGREGAR PRODUCTO. Fuente: propia.**

****

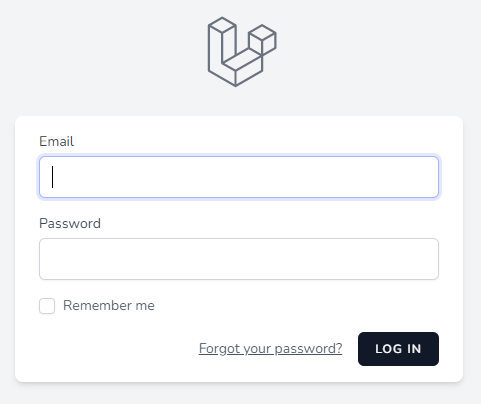
**Figura 56. EDITAR PRODUCTOS. Fuente: propia.**

****

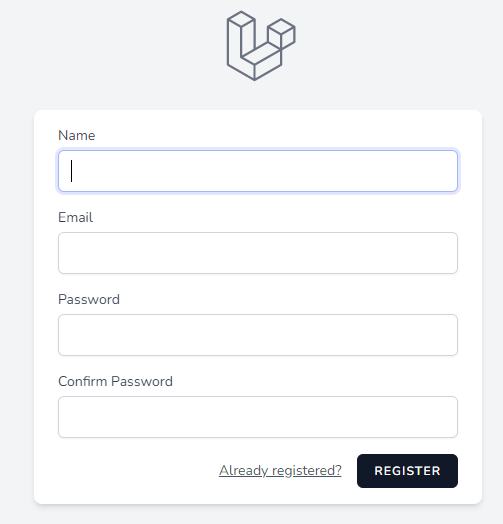
**Figura 57. BUSCADOR Y FILTRO PRODUCTOS. Fuente: propia.**

****

**Figura 58. ELIMINAR PRODUCTO (SE ELIMINÓ IBUPROFENO).** **Fuente: propia.**

****

**Figura 59. LOGIN. Fuente: propia.**

****

**Figura 60. REGISTRO. Fuente: propia.**

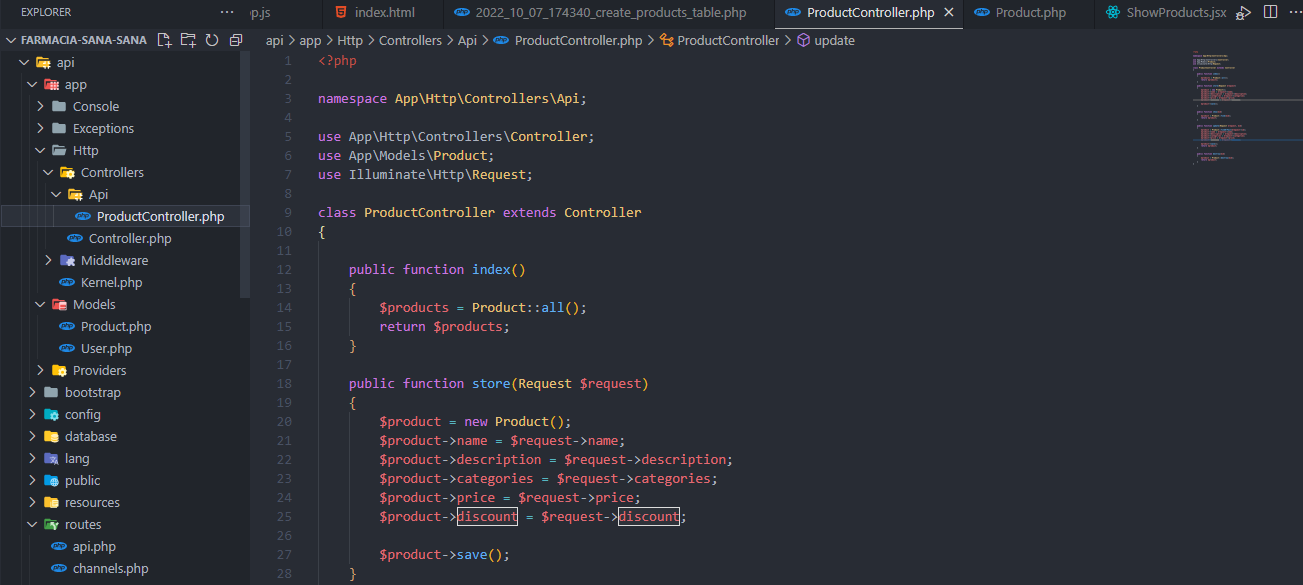


Figura 61. ProductController. Fuente: Propia.

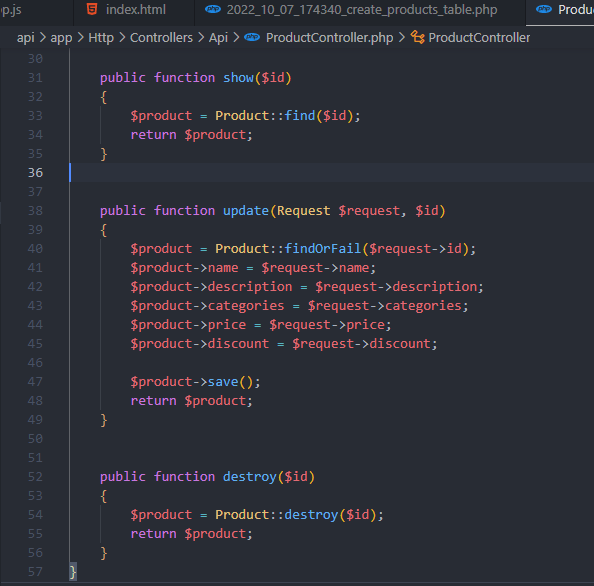


Figura 62. ProductController. Fuente: Propia.

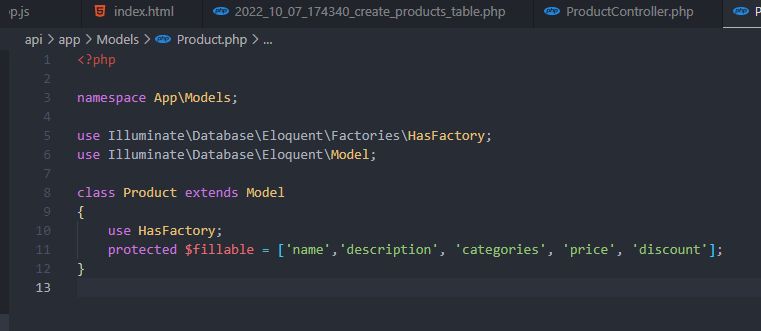


Figura 63. Model Product. Fuente: Propia.

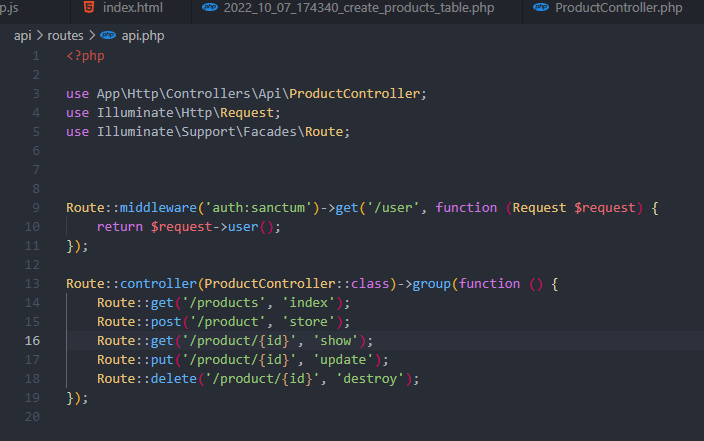


Figura 64. Api.php. Fuente: Propia.

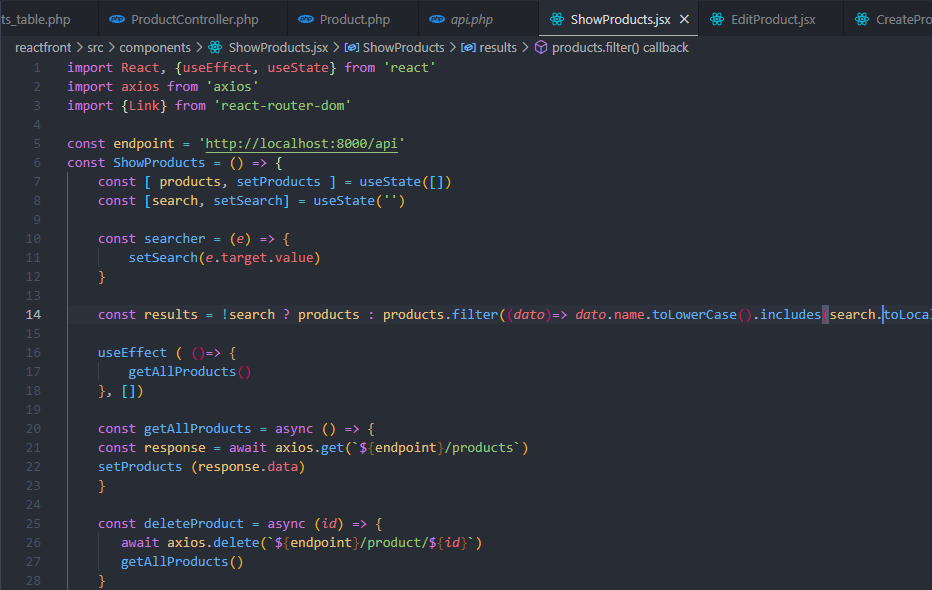


Figura 65. ShowProducts. Fuente: Propia.

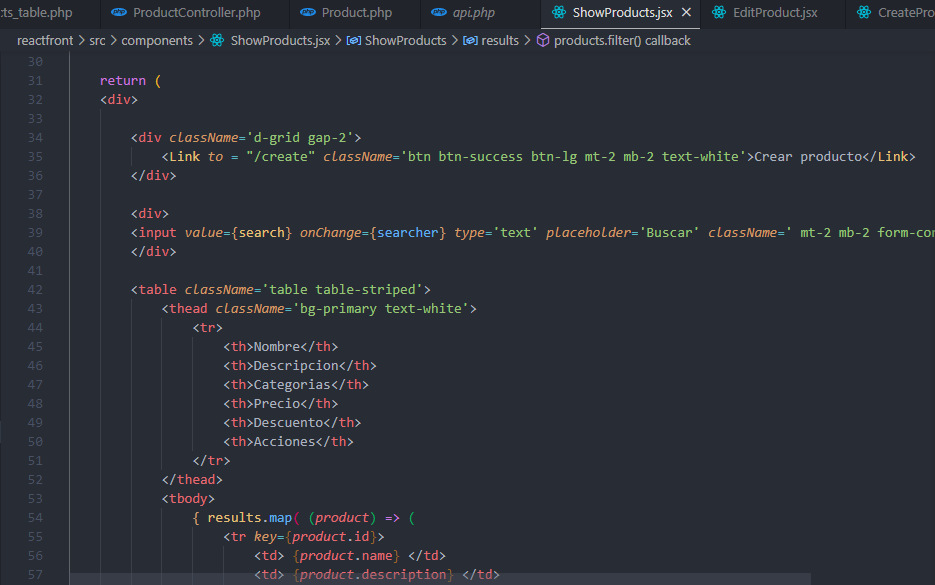


Figura 66. ShowProducts. Fuente: Propia.



Figura 67. ShowProducts. Fuente: Propia.



Figura 68. EditProducts. Fuente: Propia.

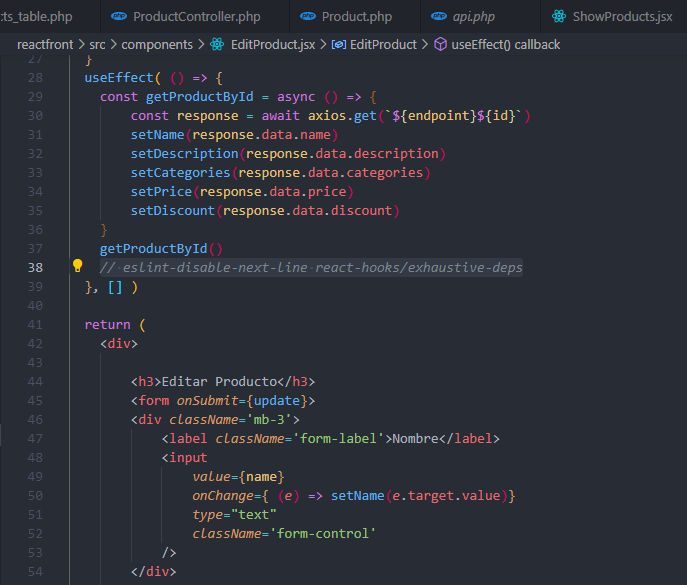


Figura 69. EditProducts. Fuente: Propia.

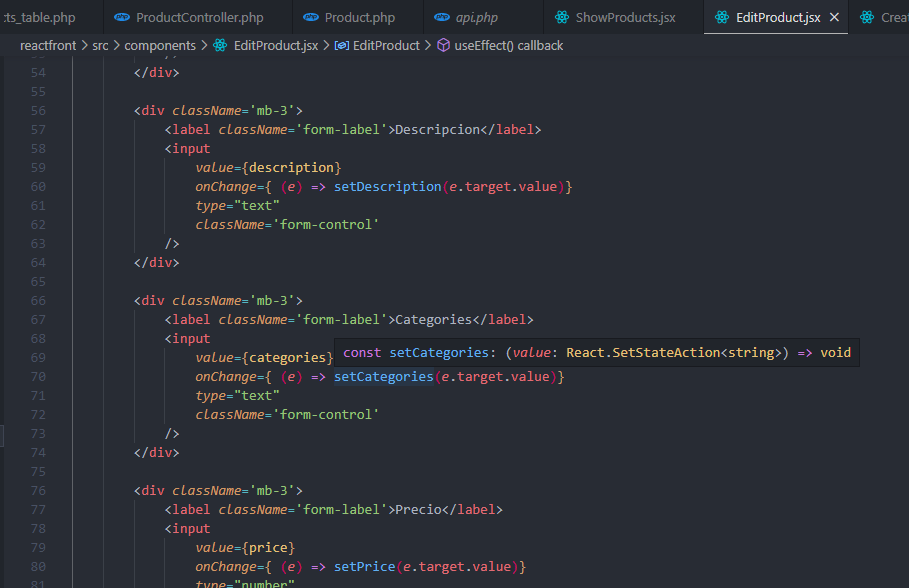


Figura 70. EditProducts. Fuente: Propia.

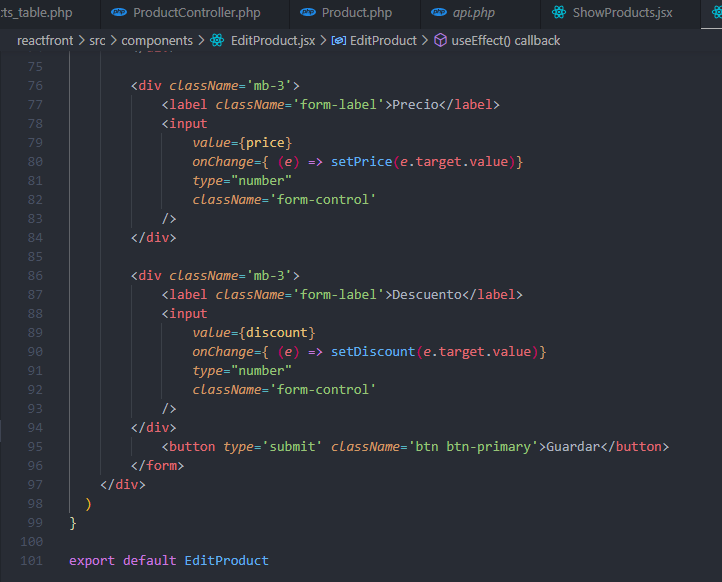


Figura 71. EditProducts. Fuente: Propia.

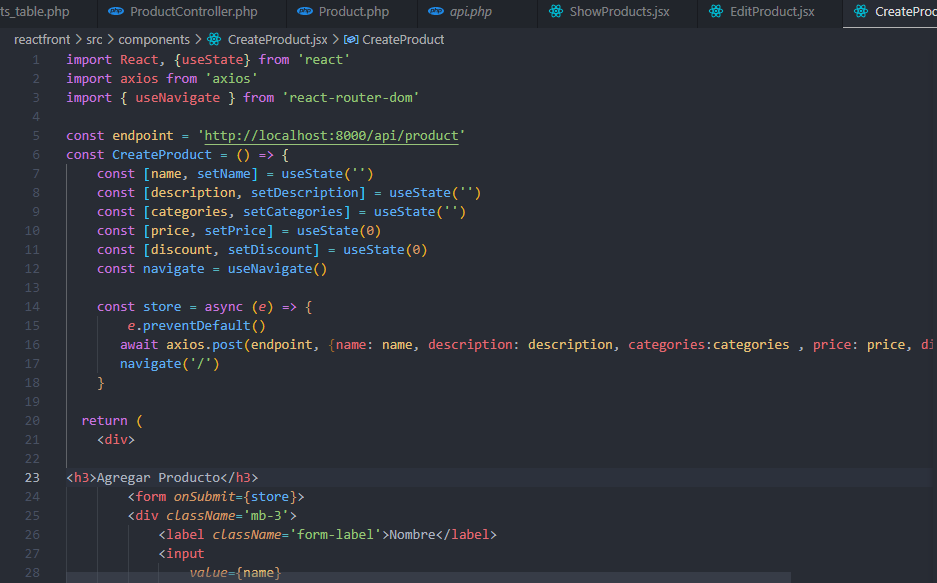


Figura 72. CreateProducts. Fuente: Propia.

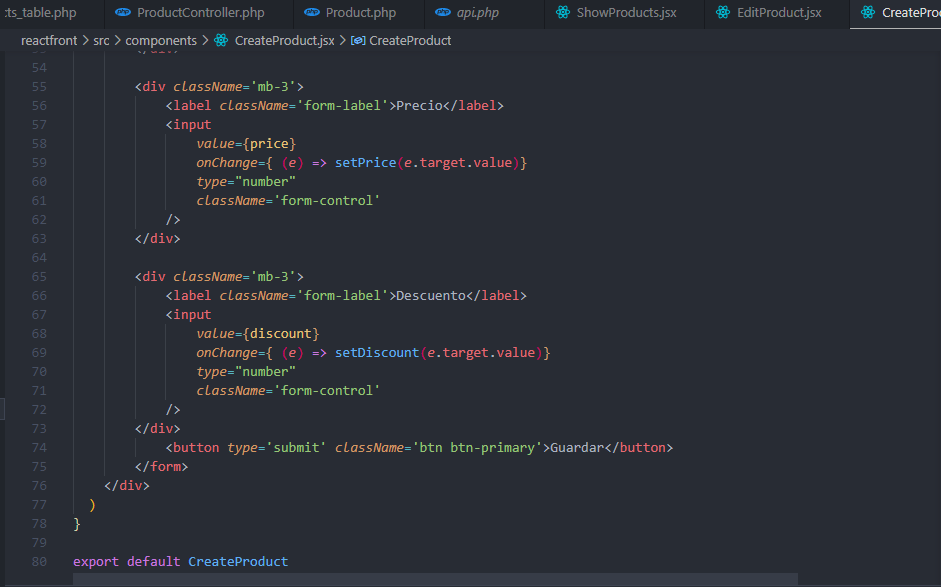


Figura 73. Create Products. Fuente: Propia.

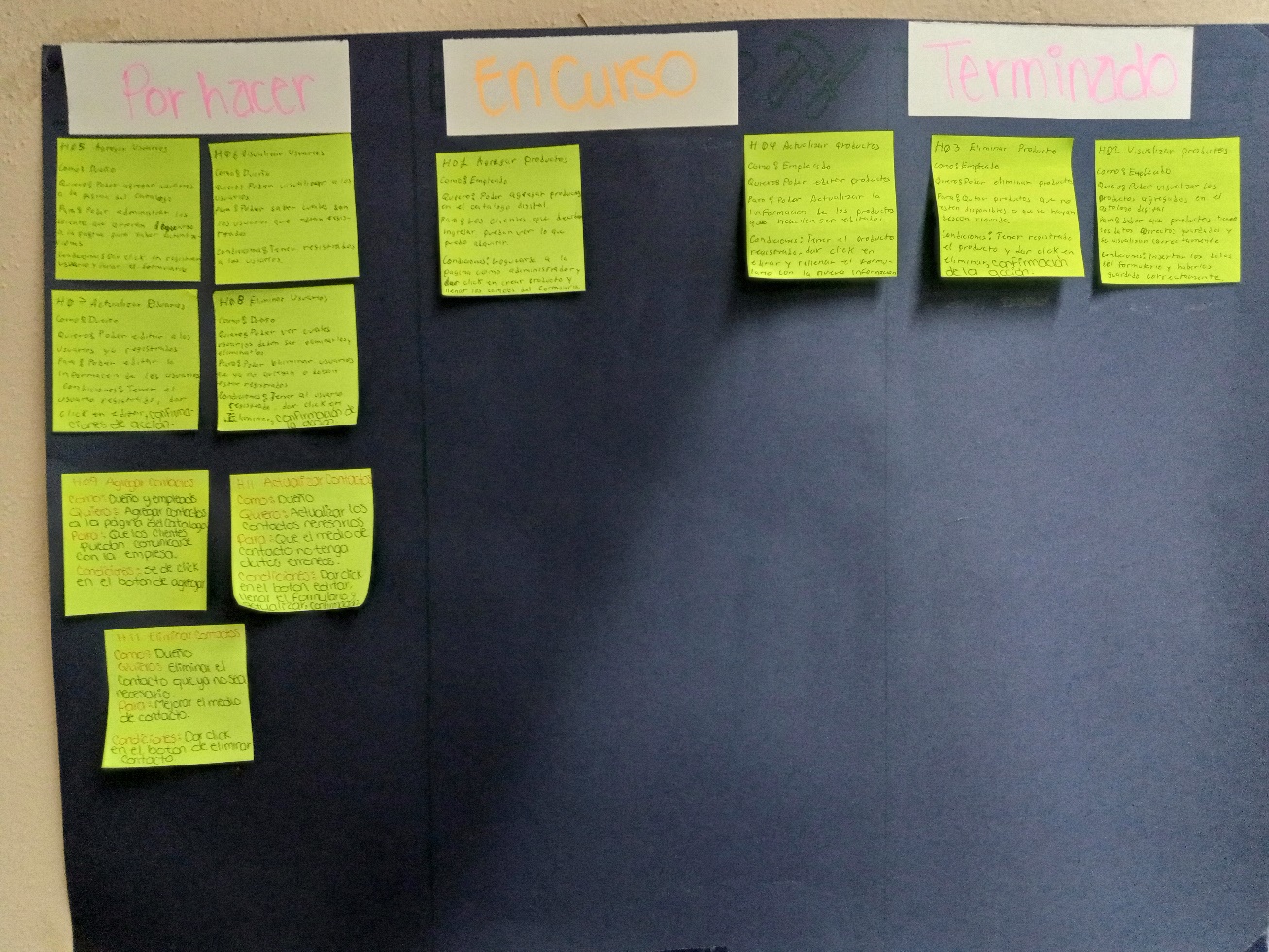


Figura 74 Avance de planeación.

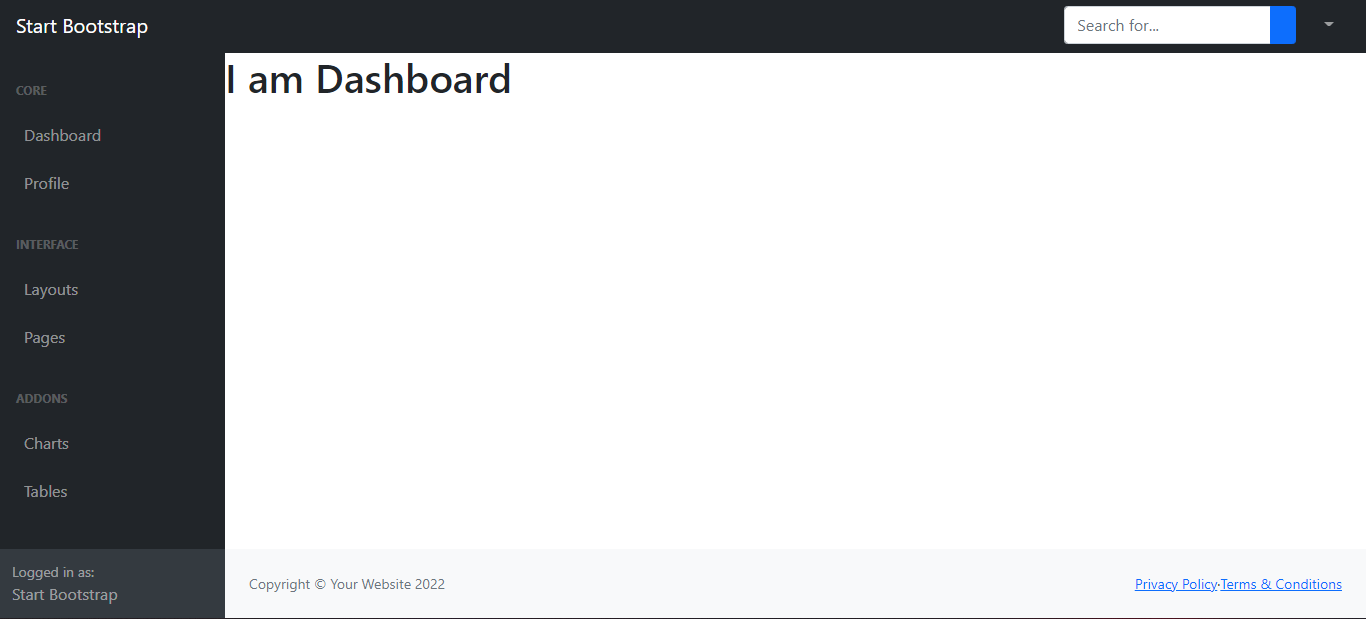


Figura 75. Vista\_Dashboard. Fuente: Propia

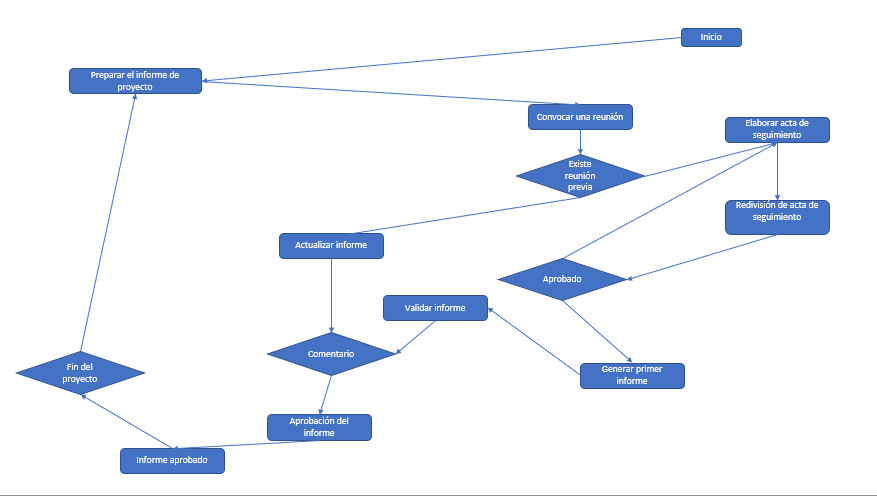


Figura 76. Diagramas de Procesos de Gestión de los Riesgos del Proyecto. Fuente: Propia



Figura 77. Diagramas de Procesos de Gestión de los Riesgos del Proyecto. Fuente: Propia

**CONCLUSIONES**

Llenar la siguiente tabla\* de auto y co-evaluación y conteste a las siguientes preguntas de manera individual:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Integrantes | 1 | 2 | 3 | Total |
| 1. Yulisa Fernanda Arreola Rivera | 9 | 9 | 9 | 27 |
| 2. Julio Cesar Rodríguez Nava | 9 | 9 | 9 | 27 |
| 3. Brandon Loeza Barrera | 9 | 9 | 9 | 27 |

**Yulisa Fernanda Arreola Rivera**

La realización de esta actividad se me hizo muy sencilla, por lo que se basa en la implementación de un proyecto en el cual se implementara una WPA, y para esto ya se tenía el conocimiento para el desarrollo del documento, en el cual se identificó la problemática, justificación, los requerimientos que se necesitaban, el alcance y la planeación de cada una de las actividades.

**¿Qué impacto tiene tu carrera en las empresas?**

La carrera de tecnologías de la información y comunicación es de gran impacto en las empresas, por lo que les permite realizar cambios e innovaciones dentro de sus empresas y así mismo poder ser más reconocidas ante la sociedad, tener un mejor acercamiento con sus clientes e incluso un mejor desarrollo en los procesos de sus productos.

**¿Cómo puedes, como ingeniero, apoyar a las empresas en este proceso postpandemia?** Se pueden apoyar en las asesorías para la implementación e innovación de tecnologías de la información y comunicación, o incluso en la educación por lo que varias escuelas en otros estados o países manejan hasta el pase de lista por medio del código QR, por lo que es más fácil y practico tanto para los estudiantes como para los maestros.

**¿Con base en lo investigado, cual consideras que sea el futuro de las aplicaciones web progresivas?**

Puede haber innovaciones implementados al desarrollo de las capacidades en el cuidado de la salud, en el ámbito escolar, e incluso laboral y mental, que le permita a la sociedad desenvolverse más como persona.

**Brandon Loeza Barrera**

**¿Qué impacto tiene tu carrera en las empresas?**

Se puede decir que mi carrera como tal solo tiene impacto en las empresas que tengan la necesidad de utilizar servicios informáticos o ya los utilicen de ante mano y los cuales necesiten ser mejorados o quieran implementarlos.

**¿Cómo puedes, como ingeniero, apoyar a las empresas en este proceso post pandemia?**

Se podría decir que el hecho de que actualmente ya varias empresasentendieran que el integrarse a los aspectos digitales como digitalizar sus servicios o tener una página propia es algo que los puede beneficiar as de lo que creen.

**¿Con base en lo investigado, cual crees que sea el futuro de las aplicaciones web progresivas?**

Como tal pienso que estará en el hecho de que se podrán actualizar en tiempo real sin la necesidad de que las personas tengan que refrescar la página o que tengan que descargar algo para poder acceder a esa actualización y serán más rápidas y no tendrán problemas de rendimiento.

**Julio Cesar Rodríguez Nava**

**¿Qué impacto tiene tu carrera en las empresas?**

En la actualidad en las empresas ya es muy indispensable la tecnología por la facilidad que nos da para trabajar y los diferentes conocimientos adquiridos en la carrera sabiendo generar estrategias para negocios, mejorar el seguimiento de procesos, optimizar los recursos y crear herramientas que pueden facilitar el trabajo.

**¿Cómo puedes, como ingeniero, apoyar a las empresas en este proceso post pandemia?**

Generar estrategias para facilitar la interacción con la empresa como generar sitios web o medidas de salud para la empresa.

**¿Con base en lo investigado, cual crees que sea el futuro de las aplicaciones web progresivas?**

Con las aplicaciones web progresivas las actualizaciones son rápidas porque se actualizan al momento de que se realizó la modificación y evitar recargar seguido la página, aparte de que llegan a ser rápidas en temas de rendimiento mejorando la interacción con los usuarios.

**CONCLUSIONES 2**

Llenar la siguiente tabla\* de auto y co-evaluación y conteste a las siguientes preguntas de manera individual:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Integrantes | 1 | 2 | 3 | Total |
| 1. Yulisa Fernanda Arreola Rivera | 9 | 9 | 9 | 27 |
| 2. Julio Cesar Rodríguez Nava | 9 | 9 | 9 | 27 |
| 3. Brandon Loeza Barrera | 9 | 9 | 9 | 27 |

**Yulisa Fernanda Arreola Rivera**

El desarrollo de esta actividad fue sencillo por lo que se pudieron conocer los conceptos básicos de la materia, en la cual se pueden comprender los conceptos y así mismo tener una idea de cómo implementarlo en las actividades futuras.

En cuanto al trabajo en equipo, considero que se pudo realizar como estaba planeado, siguiendo cada uno de los apartados a evaluar, por lo que de autoevaluación y coevaluación considero que podemos alcanzar un 10.

• ¿Por qué consideras que JavaScript ha tenido el éxito actual?

Considero que JavaScript ha tenido éxito debido a sus herramientas que ayudan a que se puedan desarrollar sitios web, animar imágenes, además de que su lenguaje no es complejo, permitiendo así aprender de manera rápida y comprender el funcionamiento de cada uno de sus herramientas y funcionamiento, como lo son sus eventos, notificaciones, estructuras de control, entre muchas más.

• ¿Por qué será importante generar una PWA del lado servidor?

La importancia que tiene el generar una PWA del lado del servidor es mucha por lo que permite que las cargas de la página webs progresivas sean más rápidas al momento de buscar o procesar la información que se desea y se descargue el contenido una vez que se conecta a internet, mostrándonos la página precargada, también nos permite navegar entre las páginas que se desean ir consultando e ingresando.

**Julio Cesar Rodríguez Nava**

¿Por qué consideras que javascript ha tenido el éxito actual?

Javascript tiene éxito porque es de los más utilizados para sitios web y algúna aplicación, siempre que hablamos sobre ese tema lo primero que más se escucha es JavaScript Gracias a JavaScript, la comunicación con el servidor solo se hace cuando el usuario necesita, o realiza alguna acción dinámica.

¿Por qué será importante generar una PWA del lado servidor?

Porque las PWA cargan los sitios web de forma más rápida también tomando en cuenta que las actualizaciones de dichos sitios web son en el momento en que se realiza algún cambio y puede llegar a ser más cómodo.

**Brandon Loeza Barrera**

¿Por qué consideras que JavaScript ha tenido el éxito actual?

Debido a la flexibilidad que este tiene a la hora de ser utilizado y más que nada también para poder realizar todos los elementos necesarios.

¿Por qué será importante generar una PWA del lado del servidor?

Esto principalmente es debido a que la estructura de esta permitirá que se implemente mejor al servidor además de que podrá ser editada de manera concreta y completa.

**CONCLUSIONES 3**

Llenar la siguiente tabla\* de auto y co-evaluación y conteste a las siguientes preguntas de manera individual:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Integrantes | 1 | 2 | 3 | Total |
| 1. Yulisa Fernanda Arreola Rivera | 9 | 9 | 9 | 27 |
| 2. Julio Cesar Rodríguez Nava | 9 | 9 | 9 | 27 |
| 3. Brandon Loeza Barrera | 9 | 9 | 9 | 27 |

**Yulisa Fernanda Arreola Rivera**

¿Consideras importante el diseño responsivo en un PWA, por qué?

Es de gran importancia que una PWA cuente con un diseño responsivo el cual le permite que tenga configuraciones específicas, como colores, configuración de distintos tipos de resolución de pantalla, permitiendo que este se pueda visualizar en distintos dispositivos en los que consideremos necesarios.

¿En qué te puede ayudar la gestión de riesgos?

La gestión de riesgos ayuda mucho en la realización de un proyecto, por lo que permite conocer cada uno de los riesgos que se tienen en la realización del proyecto, planteando que tanto puede afectar, o incluso como se puede evitar o solucionar para que este no sea tan fuerte, también identificando el nivel de riesgo con el que este puede contar.

En cuanto a la autoevaluación y coevaluación considero que podríamos alcanzar un 9, aunque si tuvimos algunos pequeños detalles faltantes.

**Julio Cesar Rodríguez Nava**

¿Consideras importante el diseño responsivo en un PWA, por qué?

¿En qué te puede ayudar la gestión de riesgos?

**Brandon Loeza Barrera**

¿Consideras importante el diseño responsivo en un PWA, por qué?

¿En qué te puede ayudar la gestión de riesgos?

**REFERENCIAS**

*Aplicaciones Web Progresivas | MDN*. (n.d.). Retrieved September 15, 2022, from https://developer.mozilla.org/es/docs/Web/Progressive\_web\_apps

arkaitzgarro. (n.d.). *Eventos | JavaScript*. Retrieved September 29, 2022, from https://www.arkaitzgarro.com/javascript/capitulo-15.html

arteco-consulting. (2020, September 1). *Por qué DEBES usar Spring Boot en tus proyectos JAVA - //Arteco*. https://www.arteco-consulting.com/post/por-que-debes-usar-spring-boot

Carlos Rojas. (2017, December 6). *Entendiendo el Cache API*. https://desarrolloweb.com/articulos/entendiendo-cache-api.html

Christian Liebel. (2021, April 7). *Overview, Web Components & Architecture - Real-World PWA: The Making Of Paint.Js.Org - Part 1 - Thinktecture AG*. https://www.thinktecture.com/en/pwa/making-of-paint-js-overview-web-components/

developer mozilla. (n.d.-a). *Estructura de una aplicación web progresiva - Aplicaciones Web Progresivas | MDN*. Retrieved September 29, 2022, from https://developer.mozilla.org/es/docs/Web/Progressive\_web\_apps/App\_structure

developer mozilla. (n.d.-b). *Funciones - JavaScript | MDN*. Retrieved September 29, 2022, from https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Functions

DEVOPS. (2020, July 30). *Comparativa de Flask vs Django*. https://www.ilimit.com/blog/flask-vs-django/

Dragon Nomada. (2021, May 24). *Arquitectura de una PWA. Componentes de las Aplicaciones Web… | by Dragon Nomada | React Adventure | Medium*. https://medium.com/react-adventure/arquitectura-de-una-pwa-26e4c4a58da2

Edinson Tique. (2020, October 24). *QUÉCODIGO*. https://www.quecodigo.com/blog/notificaciones-web-html/

Enrique Oriol. (2016, December 1). *Descubriendo las Progressive Web Apps, el futuro mobile*. http://blog.enriqueoriol.com/2016/12/progressive-web-apps.html

Fany Peregrina. (2014, February 11). *Ejemplo de carta de proyectos*. https://es.slideshare.net/fanyperegrina/ejemplo-de-carta-de-proyectos

gelmer restrepo hernandez. (n.d.). *Calaméo - Carta inicio de proyecto*. Retrieved September 29, 2022, from https://es.calameo.com/books/0004149205b649394a20c

GodoFredo. (2018, December 6). *10 Mejores Frameworks de Node.js para utilizar en 2019*. https://godofredo.ninja/10-mejores-frameworks-de-node-js-para-utilizar-en-2019/

Iván Ramírez. (2018, July 3). *¿Qué es una Aplicación Web Progresiva o PWA?* https://www.xataka.com/basics/que-es-una-aplicacion-web-progresiva-o-pwa

javfon1. (2019, December 20). *Los mejores frameworks de Node.js para aumentar la productividad – Developing in Spanish*. http://developinginspanish.com/2019/12/20/los-mejores-frameworks-de-node-js-para-aumentar-la-productividad/

Jorge Aguilar. (2017, August 14). *[:es]Desarrollo de aplicaciones Web con Spring Boot[:] - Software Evolutivo*. https://softwareevolutivo.com.ec/espanol-desarrollo-de-aplicaciones-web-con-spring-boot/

Julio García. (2017, October 20). *Qué es Ruby y sus características | OpenWebinars*. https://openwebinars.net/blog/que-es-ruby/

Mehul Rajput. (2001, August 23). *The Most Popular Progressive Web Apps (PWA) Frameworks in 2022*. https://www.mindinventory.com/blog/best-progressive-web-apps-frameworks/

Miguel Angel Alvarez. (2003, August 13). *Ejemplos de eventos en Javascript. Onabort*. https://desarrolloweb.com/articulos/1246.php

Miguel Angel Alvarez. (2021, December 28). *Los tipos de eventos en Javascript*. https://desarrolloweb.com/articulos/1236.php

mytaskpanel. (n.d.). *Los 4 frameworks más usados de Node.js: nuestras recomendaciones*. Retrieved September 15, 2022, from https://www.mytaskpanel.com/los-4-frameworks-mas-usados-de-node-js-nuestras-recomendaciones/

nubecolectiva. (2020, January 8). *5 Principales Frameworks para Node JS | Blog Nube Colectiva*. https://blog.nubecolectiva.com/5-principales-frameworks-para-node-js/

Pablo Camino Bueno. (2018, August 3). *Qué es Django y por qué usarlo | OpenWebinars*. https://openwebinars.net/blog/que-es-django-y-por-que-usarlo/

Pooja Sharma. (2019, December 13). *Las 10 principales ventajas de los servicios de desarrollo de Laravel para empresas*. https://cynoteck.com/es/blog-post/top-10-advantages-of-laravel-development-services-for-enterprises/

*¿Qué es PWA? | Conceptos de comercio electrónico*. (n.d.). Retrieved September 15, 2022, from https://www.sana-commerce.com/es/conceptos-de-comercio-electronico/que-es-pwa/

Rafael Altube Vera. (2021, March 31). *Qué es Laravel: Características y ventajas | OpenWebinars*. https://openwebinars.net/blog/que-es-laravel-caracteristicas-y-ventajas/

Reclu IT. (2022, February 17). *Reclu IT*. https://recluit.com/ventajas-al-desarrollar-con-ruby-on-rails/

salamarkesa. (n.d.). *Estructuras de control en Javascript | Explicación y ejemplos – Salamarkesa*. Retrieved September 29, 2022, from https://www.salamarkesa.com/estructuras-control-javascript-explicacion-ejemplos/

Sandra Garrido Sotomayor. (2022, September 30). *Qué son las metodologías ágiles y cuáles son sus ventajas empresariales*. https://www.iebschool.com/blog/que-son-metodologias-agiles-agile-scrum/

softwareontheroad. (2019, August 17). *Los 10 mejores Frameworks de node.js para el 2019*. https://softwareontheroad.com/es/nodejs-frameworks/

Staff BEDU. (2020, October 23). *3 razones para usar el framework de Django*. https://bedu.org/blog/tecnologia/3-razones-para-usar-el-framework-django

STARTECHUP. (2021, January 25). *Los 4 Mejores Frameworks de Backend Web*. https://www.startechup.com/es/blog/the-top-4-web-backend-frameworks/

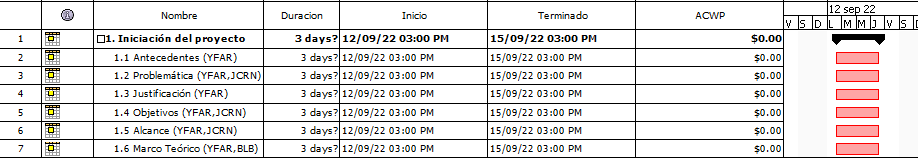
studylib. (n.d.). *Carta de aprobación del Proyecto*. Retrieved September 29, 2022, from https://studylib.es/doc/5344314/carta-de-aprobaci%C3%B3n-del-proyecto

uniwebsidad. (n.d.). *4.1. Funciones (Introducción a JavaScript)*. Retrieved September 29, 2022, from https://uniwebsidad.com/libros/javascript/capitulo-4/funciones

Vidal, M. (2019). ¿Qué son las Progressive Web Apps? ¿Por qué son tan importantes? *Thinking for Innovation*. https://www.iebschool.com/blog/progressive-web-apps-analitica-usabilidad/

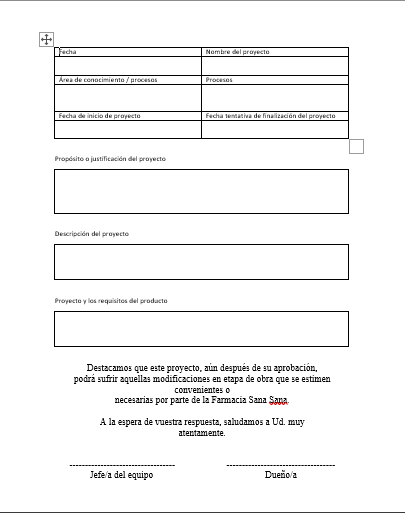
**ANEXOS**

**ANEXO 1. PLANEACIÓN.**

****

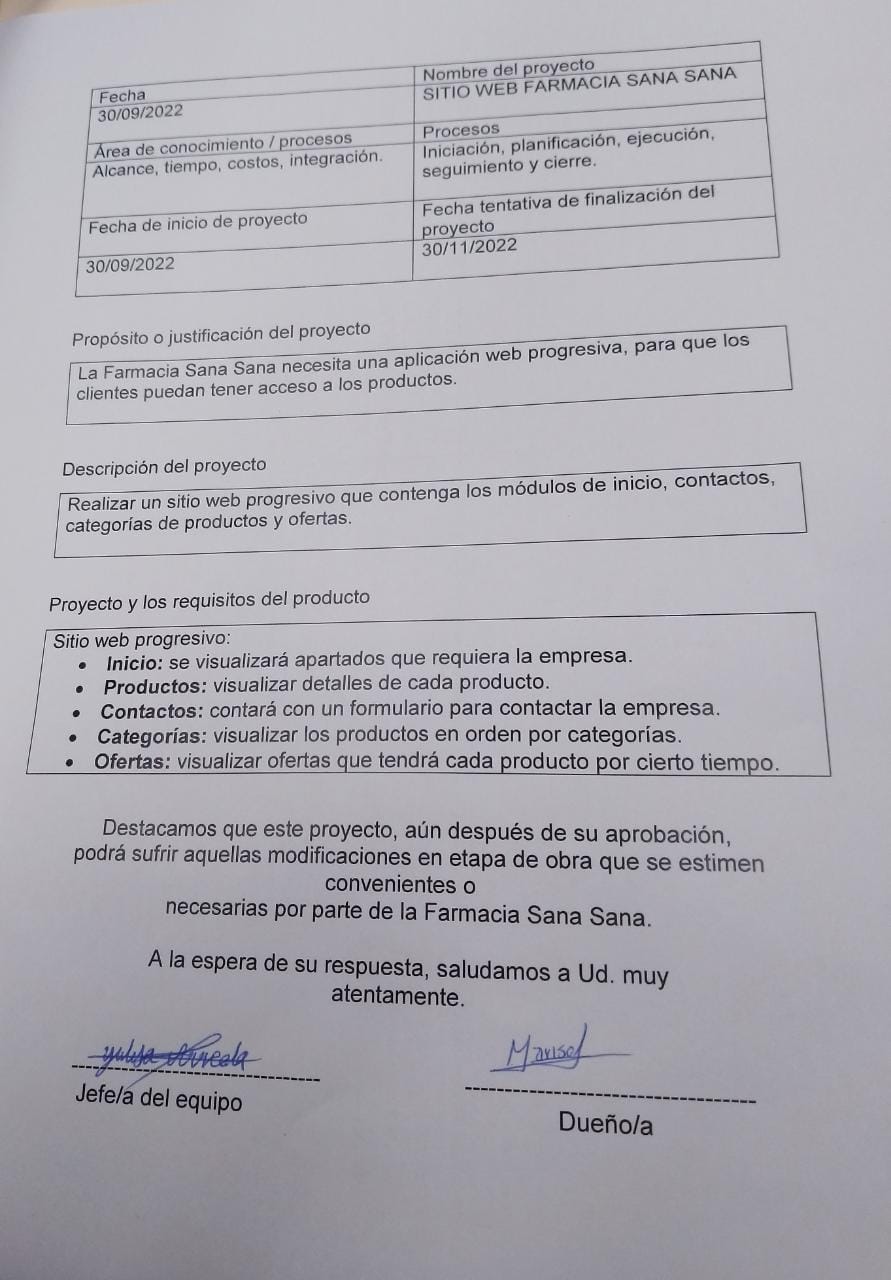
**Iniciación del proyecto.**

**ANEXO 2. ACTA DE INICIO DEL PROYECTO.**



**Acta de inicio del proyecto.**

**ANEXO 3. ACTA DE INICIO DE PROYECTO FIRMADA.**



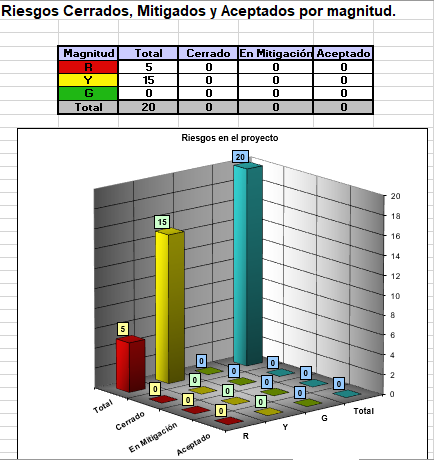
**Acta de inicio del proyecto firmada.**

**Anexo 4 Riesgos.**



**Riesgos del proyecto**

**Anexo 5 Grafico de riesgos.**



**Gráfico de riesgos del proyecto**