

# **ESCUELA POLITÉCNICA NACIONAL**

**ESCUELA DE FORMACIÓN DE TECNÓLOGOS**

**DESARROLLO DE SISTEMA PARA LA GESTIÓN DEL  
INVENTARIO EN FARMECC**

**DESARROLLO DE UN SISTEMA WEB**

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO  
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO SUPERIOR  
EN DESARROLLO DE SOFTWARE**

**DIANA KATHERINE ALMEIDA ANCHATUÑA**

**DIRECTOR: LOARTE CAJAMARCA BYRON GUSTAVO**

**DMQ, septiembre 2022**

## CERTIFICACIONES

Yo, Diana Katherine Almeida Anchatuña declaro que el Trabajo de Integración Curricular aquí detallado es de mi autoría; que no ha sido anteriormente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.



---

**DIANA KATHERINE ALMEIDA ANCHATUÑA**

**diana.almeida01@epn.edu.ec**

**diany1712csb@gmail.com**

Certifico que el presente Trabajo de Integración Curricular fue desarrollado por Diana Katherine Almeida Anchatuña, bajo mi supervisión.



---

**Ing. Byron Loarte, MSc.**

**DIRECTOR**

**byron.loarteb@epn.edu.ec**

## **DECLARACIÓN DE AUTORÍA**

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

Diana Katherine Almeida Anchatuña

## **DEDICATORIA**

Llena de alegría, dedico este proyecto con todo mi corazón a mi preciosa madre María Anchatuña, por brindarme apoyo incondicional y fortaleza durante todo mi camino de estudio, ser mi fuente de motivación e inspiración para superarme y siempre enseñarme con el ejemplo el amor a Dios y cuán grandiosa es su voluntad, hoy no estás conmigo amada madre, pero todas y cada una de tus enseñanzas de vida se mantendrán por siempre en mi corazón y memoria, recordando la maravillosa persona que fuiste.

A mi querido hermano Armando Almeida, por haber sido la base de mi formación, ayudándome a enfrentar las dificultades de la vida con constantes palabras de aliento, aportando grandes cosas a mi vida, por todo su apoyo moral y económico que sin esperar nada a cambio hizo posible ser la persona que soy actualmente y permitir que una de mis anheladas metas sea cumplida.

**Diana Katherine Almeida Anchatuña**

## **AGRADECIMIENTO**

Agradezco principalmente a Dios, por permitirme este gran logro en mi vida, sin su amor y bondad no podría haber superado esta etapa tan importante. Gracias porque su voluntad es grandiosa y me ha brindado a la mejor madre y al mejor hermano para apoyarme en cada decisión y logro. Por supuesto, el camino no ha sido sencillo, pero ustedes, mi querida familia, han hecho posible ver a los problemas como algo superable, pues siempre han creído en mí y en todas mis capacidades. Doy gracias especiales con todo mi corazón a mi amada madre por enseñarme con amor desde muy pequeña que los esfuerzos traen consigo su recompensa, por las experiencias vividas junto a ella, sus consejos, sus exigencias y sobre todo por sus bendiciones. Agradezco también a mi hermano, por todo su amor y por acompañarme en cada paso que he dado, transmitiéndome siempre sus conocimientos para ser una mejor persona en la vida.

**Diana Katherine Almeida Anchatuña**

# ÍNDICE DE CONTENIDO

CERTIFICACIONES.....	I
DECLARACIÓN DE AUTORÍA.....	II
DEDICATORIA.....	III
AGRADECIMIENTO.....	IV
ÍNDICE DE CONTENIDO.....	V
RESUMEN .....	VII
ABSTRACT .....	VIII
1 DESCRIPCIÓN DEL COMPONENTE DESARROLLADO.....	1
1.1. Objetivo general.....	2
1.2. Objetivos específicos .....	2
1.3. Alcance .....	2
1.4. Marco teórico .....	3
2 METODOLOGÍA .....	7
2.1. Metodología de Desarrollo .....	7
Roles.....	8
Artefactos.....	9
2.2. Diseño de Interfaces .....	12
Herramienta utilizada para el diseño.....	12
2.3. Diseño de arquitectura.....	13
Patrón arquitectónico .....	13
2.4. Herramientas de desarrollo.....	14
Librerías.....	15
3 RESULTADOS.....	17
3.1 <i>Sprint 0</i> . Configuración del ambiente de desarrollo.....	17
Definición de requerimientos .....	17
Estructura del proyecto sistema <i>web</i> . .....	20
Roles de usuario.....	20
3.2 <i>Sprint 1</i> . Resultado de las interfaces para los usuarios con perfil administrador.....	21
Consumir varios <i>endpoints</i> para presentar información general de la farmacia .....	22
Consumir un <i>endpoint</i> para presentar información general acerca de los médicos.....	22
Consumir varios <i>endpoints</i> para iniciar sesión y cerrar sesión .....	23

Consumir varios <i>endpoints</i> para gestionar sucursales.....	24
Consumir varios <i>endpoints</i> para visualizar la disponibilidad de <i>stock</i> de productos.....	25
Consumir varios <i>endpoints</i> para visualizar las ventas por sucursales .....	26
Consumir varios <i>endpoints</i> para visualizar el personal.....	27
Consumir varios <i>endpoints</i> para visualizar reportes de cada sucursal .....	28
Consumir varios <i>endpoints</i> para gestionar pedidos con proveedores .....	28
3.3 <i>Sprint 2</i> . Resultado de las interfaces para el usuario cliente. ....	29
Consumir varios <i>endpoints</i> para visualizar productos .....	29
Consumir varios <i>endpoints</i> para realizar cotizaciones de productos. ....	30
3.4 <i>Sprint 3</i> . Pruebas en el sistema <i>web</i> .....	31
Ejecución de las pruebas unitarias y resultados .....	31
Ejecución de las pruebas de rendimiento y resultados .....	32
Ejecución de las pruebas de compatibilidad y resultados .....	33
Ejecución de las pruebas de aceptación y resultados.....	34
3.5 <i>Sprint 4</i> . Despliegue del sistema web. ....	35
Despliegue del sistema web en <i>Heroku</i> .....	35
4 CONCLUSIONES .....	35
5 RECOMENDACIONES.....	36
6 REFERENCIAS BIBLIOGRÁFICAS.....	37
7 ANEXOS.....	42
ANEXO I.....	43
ANEXO II.....	44
ANEXO III.....	96
ANEXO IV .....	97

## RESUMEN

La presencia en línea de una empresa independientemente de la industria a la que pertenece, resulta tener un impacto masivo en su éxito, si bien un sistema *web* trae consigo la posibilidad de aumentar clientes potenciales, es importante también que exista un registro de los movimientos de una empresa en línea de manera que puedan ser administrados independientemente del lugar. En ese sentido, la cadena de farmacias “FARMECC”, debido a su auge durante la pandemia y la apertura de nuevas sucursales, ha tenido dificultades en la administración de los recursos existentes en las distintas sucursales, por lo que, con la finalidad de apoyar a esta empresa, en el presente proyecto se ha desarrollado un sistema *web* para la gestión del inventario, posibilitando por una parte a la interacción de los clientes con información general, catálogo y cotización de productos y por otra parte, la interacción de los administradores en la gestión de sucursales, productos, ventas, pedidos, personal y generación de reportes de ventas. Logrando así crear la presencia digital de esta empresa dedicada a la salud.

El presente trabajo de integración curricular está estructurado de la siguiente manera: en la primera sección se detallan los antecedentes, objetivos, alcance del proyecto junto con el marco teórico. En la segunda sesión se puntualiza la implementación de la metodología *Scrum*, prototipos y herramientas utilizadas para el desarrollo del sistema *web*. En la tercera sección se presenta las actividades por iteración y resultados de cada *Sprint*. Finalmente, en la cuarta sección se especifica las conclusiones y recomendaciones que se han obtenido.

**PALABRAS CLAVE:** Sistema *web*, gestión, inventario, *Scrum*, *Sprint*.



## **ABSTRACT**

The online presence of a company regardless of the industry to which it belongs, turns out to have a massive impact on its success, although a web system brings with it the possibility of increasing potential customers, it is also important that there is a record of the movements of an online business so that they can be managed regardless of location. In this sense, the pharmacy chain "FARMECC", due to its boom during the pandemic and the opening of new branches, has had difficulties in managing the existing resources in the different branches, so, in order to support to this company, in this project a web system has been developed for inventory management, allowing on the one hand the interaction of customers with general information, catalog and product prices and on the other hand, the interaction of administrators in the management of branches, products, sales, orders, personnel and generation of sales reports. Thus, managing to create the digital presence of this company dedicated to health.

This curricular integration work is structured as follows: the first section details the background, objectives, scope of the project along with the theoretical framework. In the second session, the implementation of the Scrum methodology, prototypes and tools used for the development of the web system is specified. The third section presents the activities per iteration and the results of each Sprint. Finally, the fourth section specifies the conclusions and recommendations that have been obtained.

**KEYWORDS:** *Web system, management, inventory, Scrum, Sprint.*

# 1 DESCRIPCIÓN DEL COMPONENTE DESARROLLADO

En la actualidad, la gestión de un inventario orientado a la web hace referencia a la automatización de procesos manuales, disminución de papel, disminución del tiempo, reducción de costos, etc. Además, proporciona a las organizaciones el control del flujo de entrada y salida del *stock* de inventario a través de procesos digitalizados, lógicos y sistematizados, los cuales son considerados cruciales para el crecimiento y sostenibilidad de las empresas [1].

Si bien la gestión de inventario en línea puede ser utilizada por cualquier empresa basada en productos, el control efectivo del inventario afecta a cada parte de una organización, desde los costos de almacenamiento hasta el cumplimiento preciso y oportuno de los pedidos, por lo que puede resultar un grave problema si no se lleva un seguimiento adecuado [2]. Por otra parte, otro de los problemas principales sobre el control de inventarios es la sobreventa de productos, considerada como la más caótica ya que ocurre especialmente durante períodos críticos de compras, cuando no existe suficiente stock para cumplir con la cantidad de pedidos solicitados. Así mismo, la gestión de almacenes lenta e ineficiente es provocada principalmente por la falta de organización del espacio del almacén de acuerdo con los productos y ventas de la empresa. El deterioro de los productos y presencia de “*stock* muerto” son otros de los inconvenientes, pues resultan una inversión perdida y simplemente ocupan un espacio innecesario. De igual forma, la falta de comunicación interna con información precisa provoca que los canales de venta no ofrezcan información válida, lo que en consecuencia hace que el servicio al cliente sea erróneo e inoportuno [3].

Ahora bien, la cadena de farmacias “FARMECC”, desde su lanzamiento en el 2021 ha tenido un crecimiento continuo durante la pandemia, llegando al punto de dar apertura a más sucursales en diferentes sectores de la ciudad de Quito. Sin embargo, no cuentan con el control efectivo del inventario en línea, siendo vital para que la cadena de farmacias funcione exitosamente. Por tal razón, se hace necesaria la implementación de la gestión de inventario por medio de la web para facilitar un mejor seguimiento de los productos, trabajos de producción, costos más precisos, reporte de datos en tiempo real, pronosticar el *stock* necesario para cada producto, etc. Evitando de esta manera mantener elementos estropeados o antiguos y que la gestión de la información de productos sea en tiempo real para todos los empleados de la red de farmacias.

Ante la problemática expuesta y con el propósito de evitar los inconvenientes mencionados anteriormente, en el actual Trabajo de Integración Curricular se desarrolla un sistema *web* de administración de inventarios para la cadena de farmacias

“FARMECC” otorgándoles de esta manera una serie de beneficios como: reducción de ventas perdidas (visualización de productos y ventas realizadas), gestión de productos (liquidación de elementos para liberar efectivo), detección de problemas financieros y brindar un mejor servicio a los clientes gracias al uso de la tecnología.

## **1.1. Objetivo general**

Desarrollar un sistema *web* para la gestión del inventario en FARMECC.

## **1.2. Objetivos específicos**

1. Establecer los requerimientos de los directivos de “FARMECC”.
2. Diseñar los prototipos del sistema *web* en base a los requerimientos.
3. Implementar los módulos para el sistema *web*.
4. Comprobar el funcionamiento correcto del sistema *web*.

## **1.3. Alcance**

El presente proyecto contempla la instauración de un sistema web dedicado a la gestión del inventario de la red de farmacias “FARMECC”, debido a la apertura de nuevas sucursales en la ciudad Quito. El mencionado sistema mantiene una comunicación directa con una aplicación de escritorio a través de una serie de *API's* de tipo *RESTful* para la integración y el intercambio de la información, las cuales son presentadas en los distintos módulos del sistema web.

El sistema web se compone por 10 módulos principales que son: Sección informativa, catálogo, doctores, inicio de sesión, gestión de sucursales, *stock* de productos, reportes de sucursales, venta por sucursales, pedidos y personal. Cada módulo se ha desarrollado en base a los requerimientos de los directivos de la red de farmacias “FARMECC”, una metodología ágil de desarrollo, un patrón de arquitectura, una serie de herramientas y librerías y gracias al consumo de los siguientes endpoints creados a partir de una aplicación de escritorio.

### **Información General**

- Consumir varios *endpoints* para visualizar la página informativa.

### **Catálogo**

- Consumir varios *endpoints* para visualizar productos.

### **Inicio de sesión**

- Consumir varios *endpoints* para iniciar sesión y cerrar sesión.

## **Sucursales**

- Consumir varios *endpoints* para gestionar sucursales.

## **Productos**

- Consumir varios *endpoints* para visualizar la disponibilidad de *stock* de productos.
- Consumir varios *endpoints* para realizar cotizaciones de productos.

## **Ventas**

- Consumir varios *endpoints* para visualizar las ventas por sucursales.

## **Personal**

- Consumir varios *endpoints* para visualizar el personal.

## **Reportes**

- Consumir varios *endpoints* para visualizar reportes de cada sucursal.

## **Pedidos**

- Consumir varios *endpoints* para gestionar pedidos con proveedores.

## **1.4. Marco teórico**

La tecnología *web* con el pasar del tiempo ha ido creciendo llegando al punto de incluir una gran variedad de aplicaciones comerciales, académicas, organizacionales y sociales. Actualmente, es evidente como diversas comunidades de usuarios multiculturales y multilingües dependen de la tecnología web. Esta área de conocimiento cubre el diseño, implementación y la prueba de aplicaciones basadas en la web, incluido el *software*, las bases de datos, las interfaces y los medios digitales relacionados. Así como también cubren cuestiones sociales, éticas y de seguridad que surgen de la web [4].

Hoy en día, un negocio exitoso depende mucho de su presencia en línea, pues se hace necesario la captación del público objetivo mediante los medios digitales posibles. Es así, que el *frontend* se considera sumamente importante al ser la primera impresión para los usuarios en términos de interacciones. Un *frontend* permite construir sitios web centrados en los clientes, satisfaciendo sus necesidades específicas e interactuando con ellos para incrementar la participación de los visitantes y en consecuencia su confianza en el negocio [5]. El impacto visual es uno de los factores que atribuyen credibilidad y atractivo

profesional, de hecho, es comprobado que el 94% de los usuarios utilizan el atractivo visual de los sitios web para formar una opinión de credibilidad de una empresa [6].

Ahora bien, el diseño tanto de la experiencia de usuario (UX) como de la creación de prototipos juegan un papel esencial para generar al cliente una experiencia exitosa. Un prototipo o modelo, ayuda a evitar posibles errores de diseño, permite ahorrar tiempo por la estructura previamente pensada y facilitar la interacción [7]. Para el desarrollo de prototipos digitales es necesario contar con aplicaciones y herramientas de diseño que lo hagan posible. Figma, es una de estas herramientas la cual es una plataforma de diseño de interfaz colaborativa, la cual puede ser accedida a través de un navegador [8], permite a los equipos acceder a archivos de diseño desde cualquier dispositivo debido a su naturaleza basada en la nube, además presenta tecnología vectorial para la creación de prototipos y estructuras listas para la producción [9].

El *software* basado en web, se trata de un *software* que se efectúa en una computadora central, mientras los usuarios se conectan a él desde sus dispositivos electrónicos por medio de un navegador web y el uso de Internet. Por otra parte, brinda información actualizada y en tiempo real a usuarios u otros sistemas/aplicaciones de información utilizando principios basados en hipertexto. Cabe mencionar que la accesibilidad y la integración con otros sistemas son algunos de los beneficios que traen consigo los sistemas basados en la web [10].

Un sistema de gestión de inventario se define como la composición de tecnología física y lógica, además de procesos y procedimientos que facilitan la supervisión de los productos almacenados, ya sean activos, materias primas, suministros o elementos listos para enviarse a los consumidores finales, de manera que tienen un efecto dramático en la productividad y la eficiencia cuando se implementan correctamente. Es importante mencionar que este tipo de sistemas constan fundamentalmente de una Base de datos central, reportes, identificación de cada artículo del inventario y su información asociada [11]. Hacer el seguimiento del inventario de un negocio es primordial en el nivel más básico, pues se depende considerablemente de la demanda de los consumidores [2].

En la gestión de proyectos las metodologías ágiles consisten en formas de dirigir un proyecto por fases, a través de la colaboración constante de los miembros y la mejora continua en cada una de las etapas tratadas. Si bien los equipos pasan por un proceso tanto de planteamiento, composición y pruebas, es vital la presencia de la cooperación, tanto con los grupos de trabajo como con quienes dependan de la efectuación del proyecto [12].

PHP o preprocesador de hipertexto, se trata de un lenguaje de programación de secuencias de comandos del lado del servidor, especialmente manejado para el desarrollo de sitios *web* de tipo estático, dinámico o aplicaciones basadas en la *web* [13]. Para que PHP mantenga las dependencias de los componentes requeridos en *Laravel*, es necesaria la presencia de *Composer*, un administrador de paquetes a nivel de aplicación para ayudar al desarrollador a mantener todas las librerías y paquetes siempre actualizados [14].

*Laravel*, es un marco PHP gratuito y de código abierto el cual brinda una serie de herramientas y recursos para crear aplicaciones PHP modernas. Cuenta con funciones integradas, gran variedad de paquetes y extensiones compatibles, además incluyen ORM (*Object Relational Mapper*) denominado *Eloquent* el cual posee mecanismos integrados para crear migraciones de Bases de datos y *seeders* [15]. Cabe mencionar que *Laravel* trabaja del lado del servidor, lo que posibilita la creación de aplicaciones de pila completa, con funciones que requieren un *backend* como cuentas de usuario, exportaciones, gestión de pedidos, entre otros. En este sentido, un concepto importante es el patrón arquitectónico MVC (modelo-vista-controlador) en PHP *Laravel* utiliza dicho patrón para desarrollar una aplicación en 3 partes fundamentales: Modelo o datos, una interfaz para ver/modificar datos o Vistas y operaciones que se pueden realizar en los datos o también denominados Controladores [16].

Con respecto a las API's (Interfaz de programación de aplicaciones), son mecanismos que permiten la comunicación entre 2 componentes *software* mediante un conjunto de definiciones y protocolos. Se sustenta en el envío de una solicitud de parte de la aplicación (cliente) hacia un servidor, el cual envía su respuesta. En este sentido, se tiene a las API REST, consideradas unas de las más flexibles, pues el servidor utiliza esta entrada del cliente para iniciar las funciones internas y devolver los datos de salida al cliente. REST por su parte define un conjunto de funciones como *GET*, *PUT* o *DELETE* que los clientes pueden usar para acceder a los datos del servidor. Su característica principal es que los servidores no guardan los datos del cliente entre solicitudes pues la respuesta del servidor son datos simples, sin la representación gráfica de una página *web* [17].

Por otra parte, un marco CSS, es una biblioteca de códigos con diseños *web* que permiten su fácil implementación en sistemas y aplicaciones de tipo *web*. Si bien existen varios marcos para el desarrollo *web*, *Bootstrap* y *Materialize*, son los más populares. *Bootstrap*, por su parte, busca aplicar un diseño y proporcionar definiciones de estilos básicas en los elementos HTML de un proyecto *web*. *Materialize*, se enfoca en la aportación de uniformidad en la interfaz de usuario, y permite la implementación de

transiciones y animaciones suaves y ordenadas [18]. Cabe mencionar de ambos marcos incluyen HTML, CSS y JS, por lo que su uso depende del propósito del proyecto que se esté desarrollado.

Por lo que se refiere a las pruebas de *frontend*, involucra todo lo visible durante el uso de la aplicación, es decir, la capa de presentación (cliente). Los probadores *frontend*, conociendo los requisitos del cliente, realizan pruebas de *GUI* (botones, etiquetas, menús, formularios o cualquiera de comportamiento desplegable), así como de usabilidad y funcionamiento del sistema *web* [19]. Existen varios tipos de pruebas de *frontend* que abarcan una serie de estrategias. Las pruebas unitarias, implican probar unidades individuales, tales como validaciones de entrada, cálculos y otras acciones de los elementos de la interfaz. Las pruebas de regresión visual, se enfocan en el código, comparan la interfaz final con la que se espera en el entorno de ensayo. Las pruebas de integración garantizan que los múltiples elementos del sistema estén estrechamente acoplados entre sí. Las pruebas de aceptación, se enfocan en que las configuraciones realizadas en el *frontend* se realicen. Finalmente, las pruebas entre navegadores, se realizan para garantizar que el sistema o aplicación *web* funcione correctamente independientemente del navegador que utilice el usuario. Este proceso de pruebas de *frontend*, conlleva beneficios como la detección de problemas de rendimiento, la validación del comportamiento de la aplicación en otros navegadores y sistemas operativos, así como la mejora de calidad de interacción y la experiencia de usuario [20].

## 2 METODOLOGÍA

El estudio de caso es una metodología de la investigación que analiza detalladamente un tema específico, ya sea persona, grupo o evento, y se utilizan generalmente en estudios de tipo social, educativo, clínico y comercial. Este tipo de diseño de investigación se considera apropiado especialmente para casos de conocimientos concretos, contextuales y profundos, pues generalmente involucra métodos cualitativos que conllevan la descripción, comparación, evaluación y asimilación de las características de un problema de investigación. En otras palabras, un estudio de caso permite mantener un proyecto enfocado y que sea manejable en especial cuando no se puede realizar una investigación a gran escala [21].

En vista de esto, el presente Trabajo de Integración Curricular mantiene un estudio de casos, pues parte de investigaciones para desarrollar el sistema *web* para la gestión de inventarios, por medio de entrevistas con los directivos de la red de farmacias “FARMECC” e investigaciones a través de otros sitios *web* dedicados a la administración de inventarios. Todo esto, permite llevar a cabo la creación de un sistema *web* para que los administradores de la cadena de farmacias puedan gestionar sus productos y ventas en cada una de sus sucursales por medio de la tecnología.

### 2.1. Metodología de Desarrollo

Una metodología de desarrollo de *software*, hace referencia a un sistema que establece un orden para efectuar varios procesos, así como los métodos de evaluación y control [22]. Cuenta con un enfoque estructurado para un proyecto de desarrollo de *software* y generalmente combinan los pasos bien definidos con una filosofía de diseño o proceso [23]. Cabe mencionar que la elección de las metodologías depende de la dirección del proyecto, presupuesto y preferencias. Existen varias metodologías de desarrollo de *software*: cascada, ágil, basada en funciones, esbelto, entre otras [22]. Sin embargo, el presente Trabajo de Integración Curricular se enfoca en la metodología ágil.

La metodología ágil, describe a uno de los procesos más simples y efectivos que promueven la continua repetición del desarrollo y pruebas durante el periodo de implementación de un proyecto [24]. Este enfoque incluye ciertos principios indispensables para generar respuestas en tiempos cortos: división de un proyecto en pasos, desarrollo incremental y entrega de *software*, desarrollo de fases en plazos breves o iteraciones (semanas o meses) y una constante retroalimentación [23]. De este modo *Agile*, impulsa a los equipos a entregar un *software* que funcione de extremo a extremo e impacte a los usuarios [25]. En ese sentido, en las próximas secciones se describe como



se ha efectuado cada uno de los aspectos de la metodología ágil *Scrum* en el desarrollo del sistema *web*.

## **Roles**

Los roles se refieren a la posición de una persona en un equipo, con la finalidad de utilizar las habilidades de cada miembro para lograr un objetivo general, de manera que se genere mayor productividad y el éxito del equipo para los proyectos futuros [26]. Los roles definidos para este proyecto son: *Product Owner*, *Scrum Master* y *Development Team*. A continuación, se presenta los roles que se han definido para el desarrollo del sistema *web*.

### ***Product Owner***

El propietario del producto se considera el responsable de establecer la dirección para el desarrollo del producto o progreso del mismo. Es decir, las funciones del Product Owner se basan en establecer los requisitos del proyecto y la capacidad de comunicarlos a su equipo de trabajo, además de establecer la prioridad de la lista de elementos y recibir la entrega de iteraciones actualizadas de manera continua [27]. En ese contexto, en la **TABLA I** se muestra a la persona encargada para este rol.

### ***Scrum Master***

Es el responsable de asegurar la coordinación del equipo de trabajo y el apoyo entre los miembros del equipo. Esto implica que toma las instrucciones del propietario y se asegura que se realicen. Adicionalmente, este rol incluye facilitar las iniciativas diarias de *Scrum* y *Sprint*, así como proteger a los miembros del equipo de interferencias y distracciones externas. Por tal razón, la persona encargada en la **TABLA I** es el responsable de identificar el mejor enfoque para el desarrollo del proyecto basándose en la autoorganización, el compromiso y el respeto [27].

### ***Development Team***

Los miembros del equipo de desarrollo, son las personas cuyas funciones incluyen las responsabilidades interfuncionales esenciales para transformar un requisito en un producto tangible para los usuarios finales [27]. Muestra de ello, la **TABLA I** presenta a la persona asignada para este rol.

**TABLA I: Asignación de roles.**

<b>ROLES</b>	<b>NOMBRES</b>
<i>Product Owner</i>	Lcdo. Efraín Caza.
<i>Scrum Master</i>	Ing. Byron Loarte, MSc.
<i>Development Team</i>	Diana Almeida

### **Artefactos**

Los artefactos ágiles, hacen referencia a información que un equipo de *Scrum* y las partes interesadas usan para detallar un producto o servicio que está en proceso de desarrollo, así como las actividades planificadas y realizadas en tal proyecto y en producción [28]. Por lo tanto, estos artefactos son esenciales en los equipos *Scrum*, pues ayudan en la planificación de metas, organización de tareas y en el seguimiento de avances para lograr los objetivos [29].

### **Recopilación de Requerimientos**

Se define como la técnica de solicitar y reunir requisitos de alto nivel de todas las partes interesadas, con la finalidad de determinar los objetivos de un proyecto y establecer lo que se necesita para que se cumplan [30]. Al respecto, la **TABLA II** presenta el formato que se ha utilizado para la elaboración de la Recopilación de los requerimientos. No obstante, dicha tabla completa con toda la información se especifica en el **ANEXO II** del actual documento.

**TABLA II: Formato para la Recopilación de Requerimientos.**

<b>Recopilación de requerimientos</b>		
<b>Tipo de sistema</b>	<b>ID-RR</b>	<b>Enunciado del Ítem</b>
<b>Sistema web</b>	<b>RR001</b>	Como usuario administrador y usuario cliente, necesitan realizar lo siguiente: <ul style="list-style-type: none"><li>• Visualizar una página informativa.</li></ul>

	<b>RR002</b>	<p>Como usuario administrador necesita consumir varios <i>endpoints</i> para realizar lo siguiente:</p> <ul style="list-style-type: none"> <li>• Iniciar sesión.</li> <li>• Cerrar sesión.</li> </ul>
--	--------------	---

### Historias de Usuario

Se tratan de descripciones breves y sencillas de una función, se caracterizan por ser redactadas desde la perspectiva del usuario o cliente del sistema y por ser escritas con diferentes niveles de detalles [31]. Al respecto, la **TABLA III** presenta la primera Historia de usuario que se ha utilizado para el desarrollo del actual sistema *web*. Cabe mencionar que el resto de Historias de Usuario se encuentran detalladas en el **ANEXO II** del presente documento.

**TABLA III: Historia de usuario 01.**

<b>HISTORIA DE USUARIO</b>	
<b>Identificador (ID):</b> HU001	<b>Usuario:</b> Administrador y Cliente.
<b>Nombre Historia:</b> Consumir varios <i>endpoints</i> para visualizar la página informativa.	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en desarrollo:</b> Media
<b>Iteración Asignada:</b> 1	
<b>Responsable (es):</b> Diana Almeida	
<p><b>Descripción:</b> El usuario administrador y cliente necesitan consumir varios endpoints para visualizar una página informativa de la red de Farmacias FARMECC para conocer sobre la empresa (misión y visión), sucursales disponibles e interactuar con los módulos:</p> <ul style="list-style-type: none"> <li>• Inicio</li> <li>• Doctores</li> </ul>	
<b>Observación:</b> Ninguna.	

### Product Backlog

Se define como una lista esencial de elementos que contienen todo aquello que debe ser integrado en el producto, por lo tanto, actúa como una fuente de requerimientos para los cambios que se tienen que incluir en el producto o servicio. Se caracteriza por cambiar continuamente según las necesidades y demanda actual [32]. Al respecto, la **TABLA IV** muestra el formato que se ha utilizado para listar el *Product Backlog*, mientras que la

tabla completa se especifica en el **ANEXO II** del actual documento.

**TABLA IV: Formato del *Product Backlog*.**

ELABORACIÓN DEL <i>PRODUCT BACKLOG</i>					
ID – PB	ID – HU	HISTORIA DE USUARIO	ITERACIÓN	ESTADO	PRIORIDAD
PB001	HU001	Consumir varios <i>endpoints</i> para visualizar la página informativa.	1	Finalizado	Media
PB002	HU002	Consumir <i>endpoints</i> para iniciar sesión y cerrar sesión.	2	Finalizado	Alta

### ***Sprint Backlog***

Se considera como un subconjunto del *Product Backlog* y contiene los objetivos que deben adquirirse al final de cada *Sprint* en particular. Actúa como una lista de tareas para cada *Sprint*, es decir, que el desarrollador divide las Historias de Usuario en tareas individuales de modo que se calcule un tiempo estimado para cumplir con cada una de ellas [32]. A continuación, en la **TABLA V** se visualiza el formato que se ha utilizado en el desarrollo de este proyecto para listar los 5 *Sprints*, mientras que la tabla con la información completa se especifica en el **ANEXO II** del actual documento.

**TABLA V: Formato del *Sprint Backlog*.**

ELABORACIÓN DEL <i>SPRINT BACKLOG</i>						
ID – SB	NOMBRE	MÓDULO	ID-HU	HISTORIA DE USUARIO	TAREAS	TIEMPO ESTIMADO

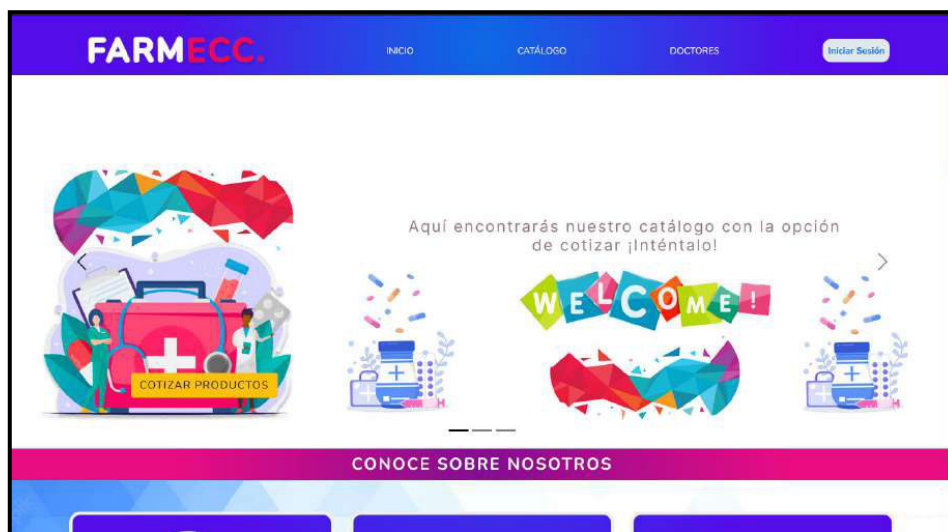
SB000	Configuración del ambiente de desarrollo	-----	---	-----	<ul style="list-style-type: none"> <li>Definición de requerimientos.</li> <li>Estructura del proyecto.</li> <li>Elaboración de prototipos.</li> <li>Diseño de módulos.</li> </ul>	20 H
-------	--	-------	-----	-------	---	------

## 2.2. Diseño de Interfaces

El diseño de interfaces de usuario (UI) hace referencia al método utilizado por los diseñadores para la generación de interfaces ya sea en *software* o en cualquier tipo de dispositivo computarizado, centrándose tanto en la apariencia como en el estilo y su objetivo es construir interfaces fáciles de usar y que sean agradables para los usuarios [33].

### Herramienta utilizada para el diseño

Los prototipos de los diferentes módulos del sistema *web* han sido realizados a través de la herramienta de diseño online *Figma*, con la finalidad de tener una base para su etapa de codificación. La **Fig.1** presenta el prototipo de la página principal con la información de la cadena de farmacias. La página principal se encuentra dividida en las secciones: *header*, *navbar*, *cards*, *carousel* y *footer*. Finalmente, los diseños completos se detallan en el **ANEXO II** del presente documento.



**Fig.1: Prototipo – Página Principal de FARMECC.**

## 2.3. Diseño de arquitectura

Una arquitectura de *software* organizada ayuda a garantizar la longevidad de la calidad interna del *software*. Una buena arquitectura de *software* ayuda a definir atributos como el rendimiento, la calidad, la escalabilidad, la capacidad de mantenimiento, la capacidad de gestión y la usabilidad [34]. La arquitectura *web* implica todos los componentes de una aplicación y ayuda a los desarrolladores *web* a crear diseños que mejoran la experiencia del usuario. Idealmente, un proyecto completo hace posible que el acceso a la información sea fácil y ayuda a comprender cómo navegar por su contenido [35]. Por tal razón, en el desarrollo del presente sistema *web* se ha optado por la integración de un patrón arquitectónico que se explica a continuación.

### Patrón arquitectónico

El patrón arquitectónico MVC (modelo, vista, controlador), ayuda a dividir el código de *frontend* y *backend* en componentes separados. De esta manera, es mucho más fácil administrar y realizar cambios en cualquier lado sin que interfieran entre sí [36]. MVC se relaciona principalmente con la interfaz de usuario/capa de interacción de una aplicación y consta de los siguientes componentes de diseño [37].

- El Modelo abarca los datos puros, por lo que no tiene ninguna lógica para mostrar dichos datos a un usuario.
- La Vista abarca el acceso a los datos de un modelo al usuario, sin la opción de manipulación de estos.
- El controlador atiende a los eventos liberados por la vista y efectúa los procesos adecuados para convocar a un método en el modelo y su resultado se refleja de forma automática en la vista [37].

En ese sentido, el desarrollo del sistema *web* dispone del siguiente patrón arquitectónico, como se muestra en la **Fig. 2**, con el fin de que el sistema conste de un modelo de datos, información de presentación e información de control, manteniendo una estructura organizada y una alta compatibilidad con las herramientas seleccionadas para las etapas de codificación, pruebas y despliegue a producción.

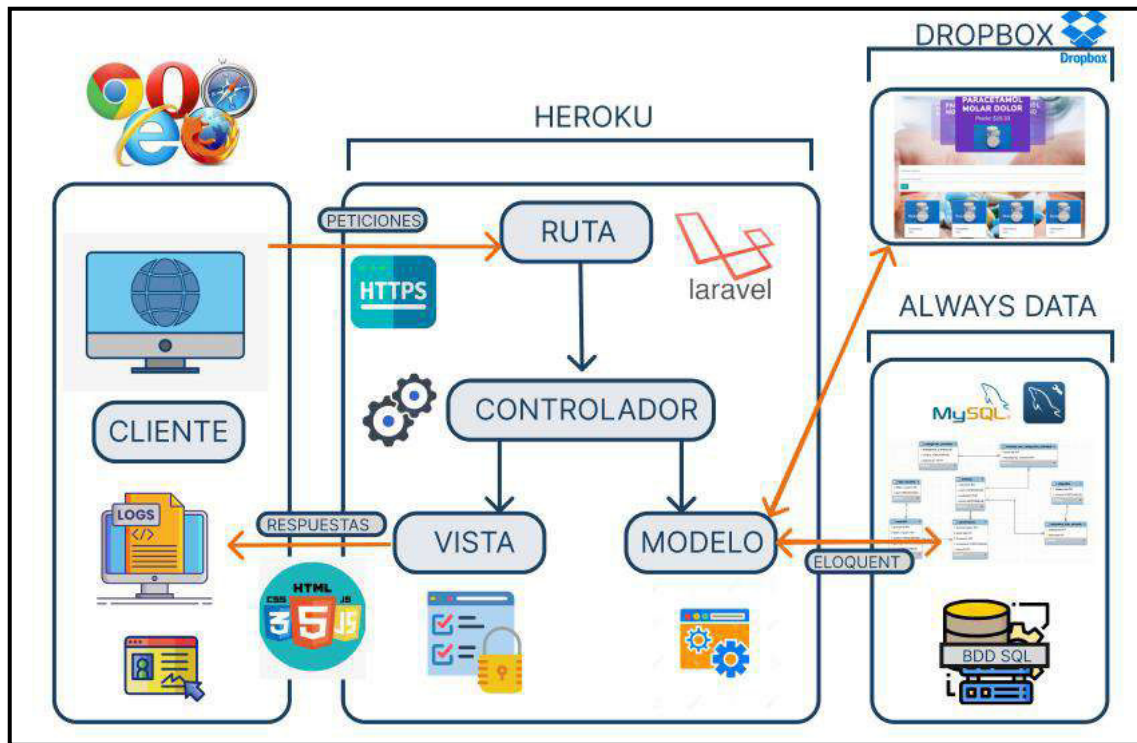


Fig.2: Patrón arquitectónico – Sistema web.

## 2.4. Herramientas de desarrollo

Las herramientas para el desarrollo del presente sistema web han sido seleccionadas en base al análisis de requerimientos y el conocimiento del equipo de desarrollo, permitiendo de esta manera tener un desarrollo mucho más ágil y que las mismas sean compatibles con las librerías que se describen en la siguiente sección. A continuación, en la **TABLA VI** se presenta la justificación de las herramientas que se han utilizado para el desarrollo del actual sistema web.

**TABLA VI: Herramientas para el desarrollo del sistema web.**

HERRAMIENTA	JUSTIFICACIÓN
Laravel	Framework de PHP de tipo MVC (modelo-vista-controlador), cuyo objetivo es ayudar al desarrollo de aplicaciones escritas en este lenguaje de programación, su motor de plantilla Blade ayuda en la construcción de páginas visualmente potentes y eficaces, además de contar con Eloquent ORM para la realización de consultas sobre objetos y en temas de seguridad ofrecer fuertes mecanismos de hash y salt [38].

<b>Bootstrap</b>	<i>Framework CSS</i> cuya finalidad es la construcción de interfaces claras, incluyendo un diseño <i>responsive</i> , mediante las herramientas y funciones que ofrece [39].
<b>Materialize</b>	Biblioteca de interfaz de usuario (UI) similar a <i>Bootstrap</i> , basado en los principios de <i>Material Design</i> y que ofrece una variedad de animaciones para ser implementadas en páginas <i>web</i> [40].
<b>Laravel mix</b>	Brinda una API de tipo fluida para el establecimiento de los pasos de compilación de <i>Webpack</i> para <i>Laravel</i> , mediante el uso de <i>CSS</i> y <i>JavaScript</i> [41].
<b>Heroku</b>	Plataforma en la nube que permite a los desarrolladores la implementación, despliegue y administración de sus aplicaciones incluyendo compatibilidad con varios lenguajes de programación como: <i>PHP</i> , <i>Java</i> , <i>Node.js</i> , <i>Scala</i> y <i>Python</i> [42].

### Librerías

La **TABLA VII**, muestra la descripción de las librerías que se han utilizado para el desarrollo del actual sistema *web*.

**TABLA VII: Herramientas para el desarrollo del sistema *web*.**

<b>LIBRERÍA</b>	<b>DESCRIPCIÓN</b>
<b>laravel/ui</b>	Paquete que proporciona la funcionalidad de autenticación en <i>Laravel</i> , de manera fácil y simple [43].
<b>tinker</b>	Paquete que posibilita la interacción con aplicaciones de <i>Laravel</i> , desde la línea de comandos [44].
<b>migrate generator</b>	Paquete de <i>Laravel</i> utilizado para crear migraciones de una base de datos existente, incluyendo los índices y FKs [44].
<b>artisan view</b>	Permite la creación de vistas mediante la línea de comandos de <i>Laravel</i> [44].



## 3 RESULTADOS

A partir de esta sección, se detallan resultados que se han obtenido en la implementación de los diferentes módulos, así como de los componentes visuales, el resultado de las pruebas y el proceso de despliegue a producción para el sistema web. Dichos resultados se presentan a través de *Sprints*, los cuales han sido planificados con anterioridad y su detalle se presenta en el **ANEXO II** del documento actual.

### 3.1 *Sprint 0*. Configuración del ambiente de desarrollo

Las actividades para el *Sprint 0* tienen que ver con:

- Definición de requerimientos.
- Estructura del proyecto sistema *web*.
- Roles de usuario.

#### **Definición de requerimientos**

##### **Consumir *endpoints* para visualizar la página informativa**

El sistema web por medio de varios métodos a través de un usuario con perfil cliente, tiene la capacidad de consumir varios *endpoints* para visualizar la página informativa de la cadena de farmacias junto con su información general (misión y visión), información de las sucursales disponibles, servicios que ofrece y datos de contacto, así como de los médicos con los que cuenta. Cabe mencionar que la información de la página inicial del sistema *web*, puede ser visualizada por los usuarios con perfiles cliente y administrador.

##### **Consumir *endpoints* para iniciar sesión y cerrar sesión**

El sistema *web* por medio de varios métodos a través de un usuario con perfil administrador, tiene la capacidad de consumir varios *endpoints* para iniciar sesión y cerrar sesión. Las credenciales de acceso (usuario y contraseña) son creadas y designadas por el equipo de desarrollo, por lo que para su cambio o reseteo se deberá contactar con los mismos. Adicionalmente, ningún usuario externo al establecido como administrador puede registrarse.

##### **Consumir *endpoints* para gestionar sucursales**

El sistema web por medio de varios métodos a través de un usuario con perfil administrador, tiene la capacidad de consumir *endpoints* para gestionar sucursales, es decir, tiene la posibilidad de crear, visualizar, modificar y eliminar información respecto a sucursales, nombre, ciudad, dirección y personal asignado. En el caso de dar de baja una

de las sucursales, se debe tomar en cuenta que no se encuentre asociada a ningún producto, personal, etc.

### **Consumir *endpoints* para visualizar productos**

El sistema web por medio de varios métodos a través de un usuario con perfil cliente y administrador, tiene la capacidad de consumir *endpoints* para visualizar productos, es decir, tiene la posibilidad de verificar los productos existentes con la posibilidad de búsqueda.

### **Consumir *endpoints* para visualizar reportes de sucursales**

El sistema web por medio de varios métodos a través de un usuario con perfil administrador, tiene la capacidad de consumir *endpoints* para visualizar los reportes de sucursales, es decir, mediante la selección del tipo de reporte: por sucursal, por rango de fechas o reporte general de ventas, se ingresan los diferentes datos requeridos y se genera un PDF con las ventas actualizadas.

### **Consumir *endpoints* para visualizar la disponibilidad de stock de productos**

El sistema web por medio de varios métodos a través de un usuario con perfil administrador, tiene la capacidad de consumir *endpoints* para visualizar la disponibilidad de productos, es decir, el administrador puede visualizar a través de la selección de una de las sucursales presentadas una lista con la cantidad de existencias por producto, descripción y un mensaje de alerta en caso que un producto esté agotado.

### **Consumir *endpoints* para visualizar la venta de sucursales**

El sistema web por medio de varios métodos a través de un usuario con perfil administrador, tiene la capacidad de consumir *endpoints* para visualizar la venta de sucursales, es decir, el administrador puede visualizar a través de la selección de una de las sucursales presentadas una lista con las ventas realizadas junto con las descripciones de ventas, fechas, valores recaudados por factura y un total de las ventas hechas en la sucursal antes seleccionada.

### **Consumir *endpoints* para gestionar pedidos con proveedores**

El sistema web por medio de varios métodos a través de un usuario con perfil administrador, tiene la capacidad de consumir *endpoints* para gestionar pedidos con proveedores, es decir, el administrador tiene la posibilidad de crear, visualizar, modificar y eliminar información respecto a los pedidos con proveedores. La modificación de pedidos se realiza a través de un formulario que permite actualizar los datos del proveedor,

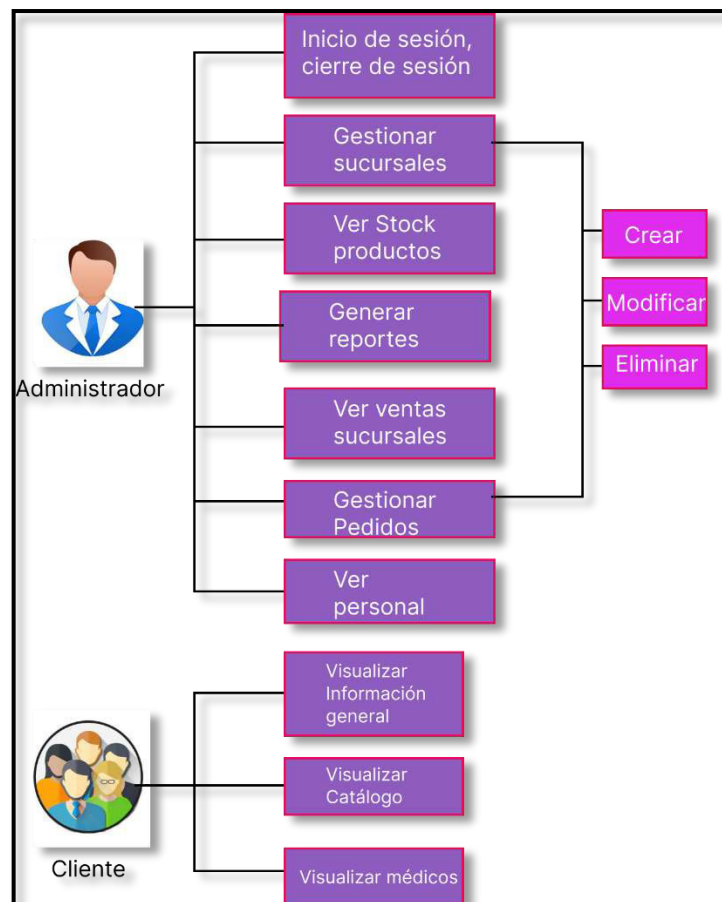
producto y cantidad.

### Consumir *endpoints* para visualizar el personal

El sistema web por medio de varios métodos a través de un usuario con perfil administrador, tiene la capacidad de consumir *endpoints* para visualizar el personal, es decir, se presenta una lista con la información de empleados, rol, experiencia laboral, contactos y horarios. El administrador puede realizar la búsqueda de personal por categoría, ya sea por sucursal, área o función de desempeño, para lo cual es necesario el ingreso del nombre del empleado.

### Consumir *endpoints* para realizar cotizaciones de productos

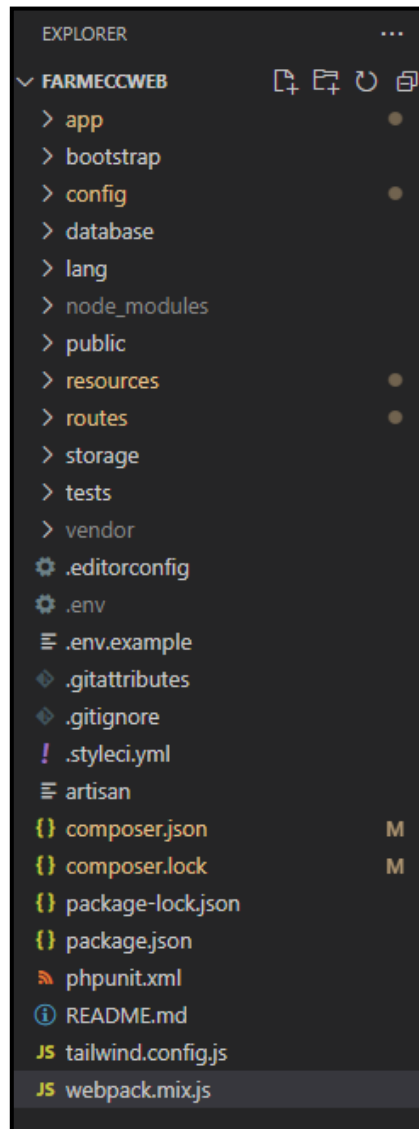
El sistema web por medio de varios métodos a través de un usuario con perfil cliente, tiene la capacidad de consumir *endpoints* para realizar cotizaciones de productos, es decir, un cliente mediante la búsqueda de un producto por selección o por nombre comercial, puede indicar que productos desea cotizar ya sea de forma individual o acumulativa. Finalmente, la **Fig. 3**, muestra las acciones que tanto los usuarios con perfil cliente y perfil administrador pueden realizar dentro del sistema web.



**Fig.3: Usuarios y funcionalidades dentro del sistema web.**

## Estructura del proyecto sistema *web*.

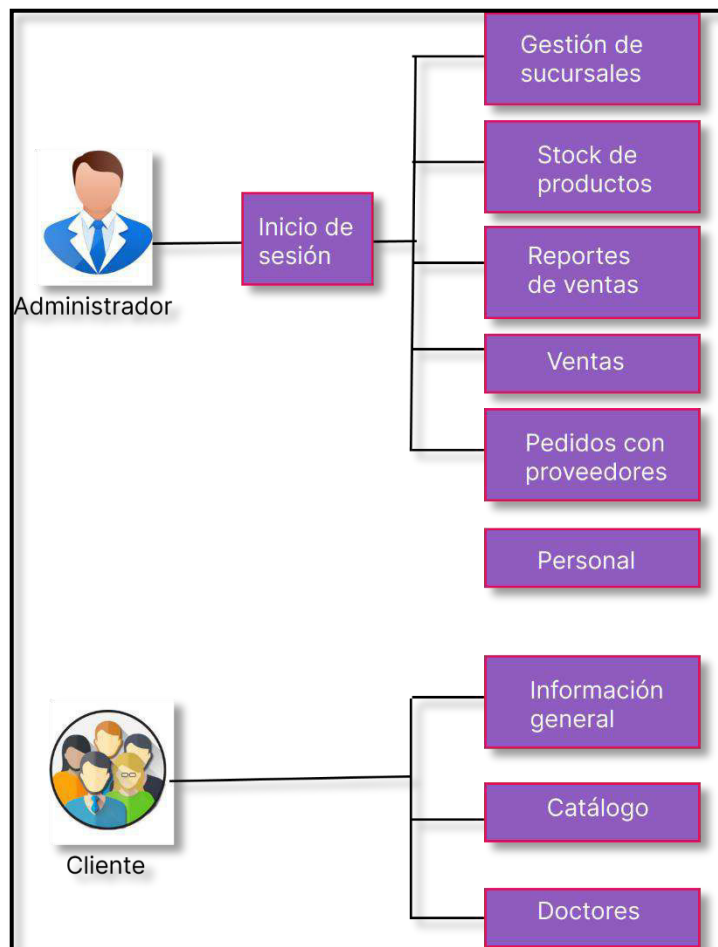
La herramienta *Visual Studio Code* se ha utilizado como editor de código para el sistema web, mediante el panel de inicio se ha establecido la estructura tanto de directorios como de archivos basándose en el patrón arquitectónico previamente seleccionado, es así que la **Fig. 4** muestra la estructura del actual proyecto.



**Fig.4:** Estructura del proyecto sistema *web*.

## Roles de usuario

La **Fig.5** muestra los módulos a los que tienen acceso cada uno de los usuarios: administrador y cliente, indicando si es necesario o no el inicio de sesión.



**Fig.5: Usuarios y módulos del sistema web.**

### **3.2 *Sprint 1. Resultado de las interfaces para los usuarios con perfil administrador.***

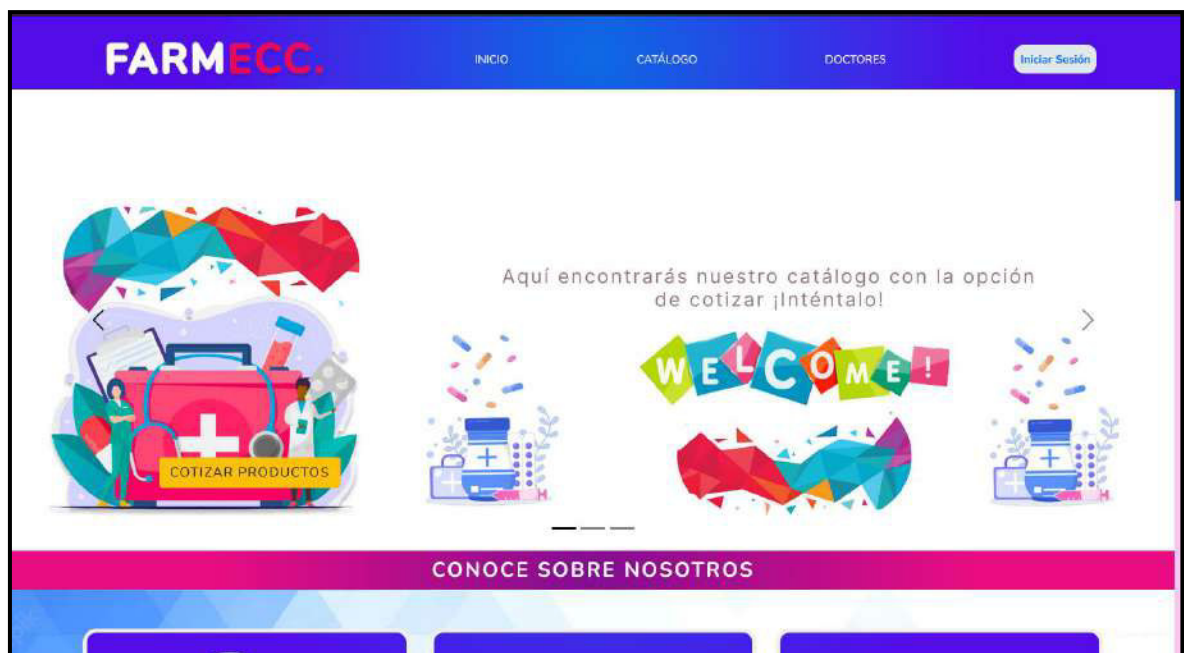
Las actividades dedicadas al *Sprint 1* tienen que ver con:

- Consumir varios *endpoints* para presentar información general de la farmacia.
- Consumir un *endpoint* para presentar información general acerca de los médicos.
- Consumir varios *endpoints* para iniciar sesión y cerrar sesión.
- Consumir varios *endpoints* para gestionar sucursales.
- Consumir varios *endpoints* para visualizar la disponibilidad de *stock* de productos.
- Consumir varios *endpoints* para visualizar las ventas por sucursales.
- Consumir varios *endpoints* para visualizar el personal.
- Consumir varios *endpoints* para visualizar reportes de cada sucursal.

- Consumir varios *endpoints* para gestionar pedidos con proveedores.

### **Consumir varios *endpoints* para presentar información general de la farmacia**

La página principal del sistema *web* presenta toda la información general de la cadena de Farmacias, las secciones que la componen contienen información sobre la misión, visión, anuncios importantes, servicios y sucursales principales. Para la presentación de la información mencionada dentro del sistema *web* se ha utilizado tanto métodos como componentes visuales que permitan el consumo de una ruta pública de tipo GET por medio de los *endpoints* proporcionados por la aplicación de escritorio. En la **Fig. 6** se observa la página principal de FARMECC. En este sentido, el proceso que detalla tanto el consumo como la visualización de la información se encuentra en el **ANEXO III** del actual documento.



**Fig.6: Página principal del sistema *web*.**

### **Consumir un *endpoint* para presentar información general acerca de los médicos**

La sección de doctores de la página principal del sistema *web* presenta la información de los médicos asociados a la cadena de Farmacias, en la cual se detalla los datos personales más esenciales para el contacto de los clientes con los doctores. Para la presentación de la información mencionada se ha consumido una ruta pública de tipo

GET por medio de los *endpoints* proporcionados por la aplicación de escritorio. En la **Fig.7** se observa la sección de doctores. En este sentido, el proceso que detalla tanto el consumo como la visualización de la información se encuentra en el **ANEXO III** del actual documento.



**Fig.7: Página de la sección doctores del sistema web.**

### **Consumir varios *endpoints* para iniciar sesión y cerrar sesión**

Para el inicio de sesión y cierre de sesión en el sistema *web* se ha implementado formularios destinados para los usuarios de perfil administrador asociados a la cadena de Farmacias, en el cual se solicita las credenciales autorizadas para su ingreso. Esto ha permitido el consumo de una ruta de tipo GET por medio de los *endpoints* proporcionados por la aplicación de escritorio para el ingreso de los administradores como se observa en las **Fig.8** y **Fig.9**. En este sentido, el proceso que detalla tanto el consumo se encuentra en el **ANEXO III** del actual documento.



Fig.8: Página de inicio de sesión en el sistema web.

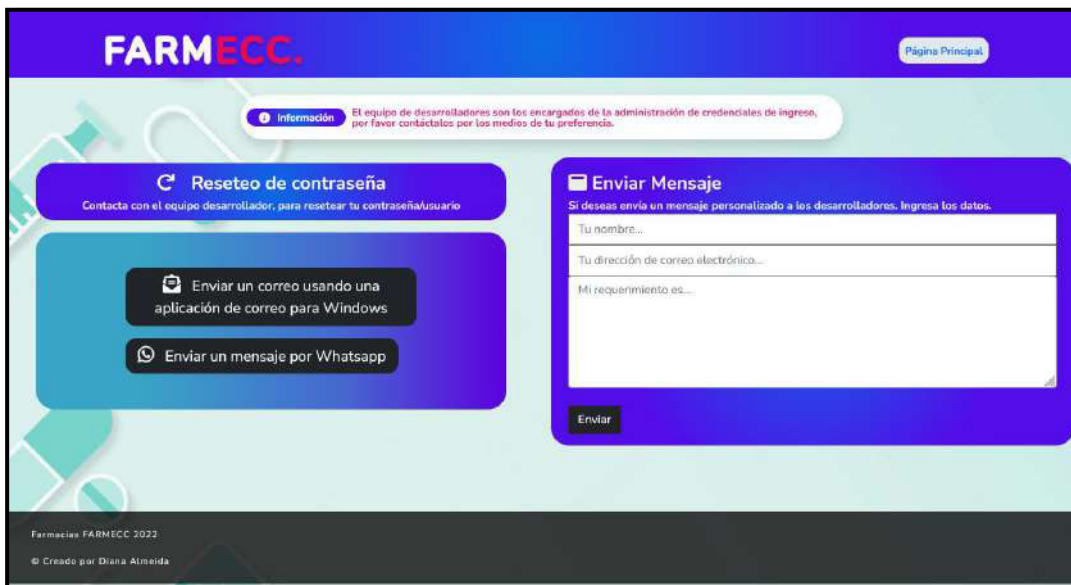


Fig.9: Página de recuperación de contraseña en el sistema web.

### Consumir varios *endpoints* para gestionar sucursales

Para el módulo de gestión de sucursales se ha implementado inicialmente botones que permiten a los administradores seleccionar una sucursal, posteriormente se presenta por medio de formularios la información solicitada, así como secciones que permiten crear, visualizar, modificar y eliminar información. Esto ha permitido el consumo de una ruta de tipo GET por medio de los endpoints proporcionados por la aplicación de escritorio para la gestión de sucursales como se observa en la Fig. 10. En este sentido, el proceso que



detalla el consumo se encuentra en el **ANEXO III** del actual documento.



**Fig.10: Página del módulo gestión de sucursales del sistema web.**

### **Consumir varios *endpoints* para visualizar la disponibilidad de *stock* de productos**

Para el módulo de *stock* de productos se ha implementado una tabla que permite visualizar la información de los productos disponibles de todas las sucursales, de manera que el usuario con perfil administrador pueda verificar el inventario global de la cadena de farmacias. Esto ha permitido el consumo de una ruta de tipo GET por medio de los *endpoints* proporcionados por la aplicación de escritorio para la visualización del *stock* productos disponibles como se observa en la **Fig.11** . En este sentido, el proceso que detalla tanto el consumo y visualización se encuentra en el **ANEXO III** del actual documento.



**Fig.11: Página del módulo *stock* de productos para visualizar disponibilidad de productos.**

### **Consumir varios *endpoints* para visualizar las ventas por sucursales**

Para el módulo de visualización de ventas se ha implementado una tabla que permite observar la información de las ventas de acuerdo a las sucursales, de manera que el usuario con perfil administrador pueda verificar las ventas realizadas en la cadena de farmacias y verificar qué productos son los más consumidos. Esto ha permitido el consumo de una ruta de tipo GET por medio de los *endpoints* proporcionados por la aplicación de escritorio para la visualización de las ventas por sucursales como se observa en la **Fig.12**. En este sentido, el proceso que detalla tanto el consumo y visualización se encuentra en el **ANEXO III** del actual documento.



Fig.12: Página del módulo visualizar ventas del sistema web.

### Consumir varios *endpoints* para visualizar el personal

Para el módulo de visualización de personal se ha implementado una tabla que permite observar la información de los empleados que se encuentran registrados hasta el momento junto con su información personal y en base a las sucursales en las que se encuentran ubicados, de manera que el usuario con perfil administrador pueda verificar el personal de cada una de las sucursales de la cadena de farmacias. Esto ha permitido el consumo de una ruta de tipo GET por medio de los *endpoints* proporcionados por la aplicación de escritorio para la visualización de los empleados contratados como se observa en la Fig.13. En este sentido, el proceso que detalla tanto el consumo y visualización se encuentra en el ANEXO III del actual documento.

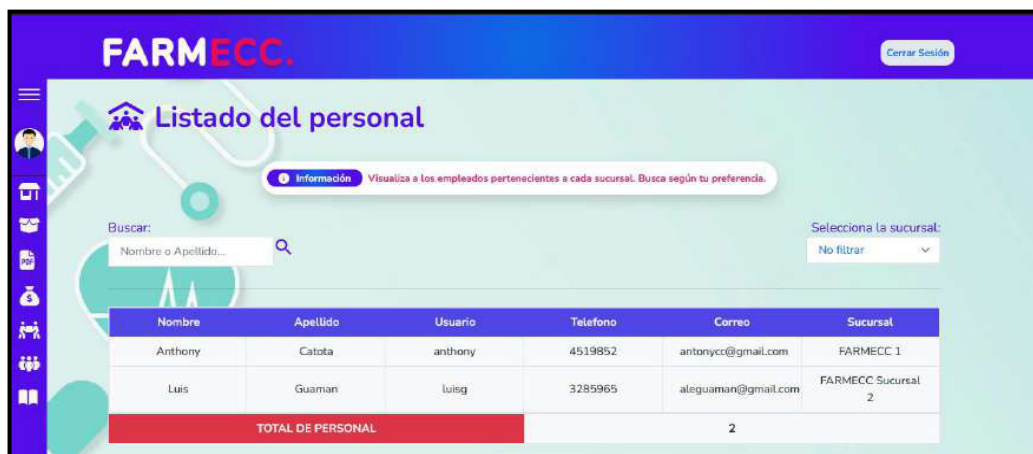
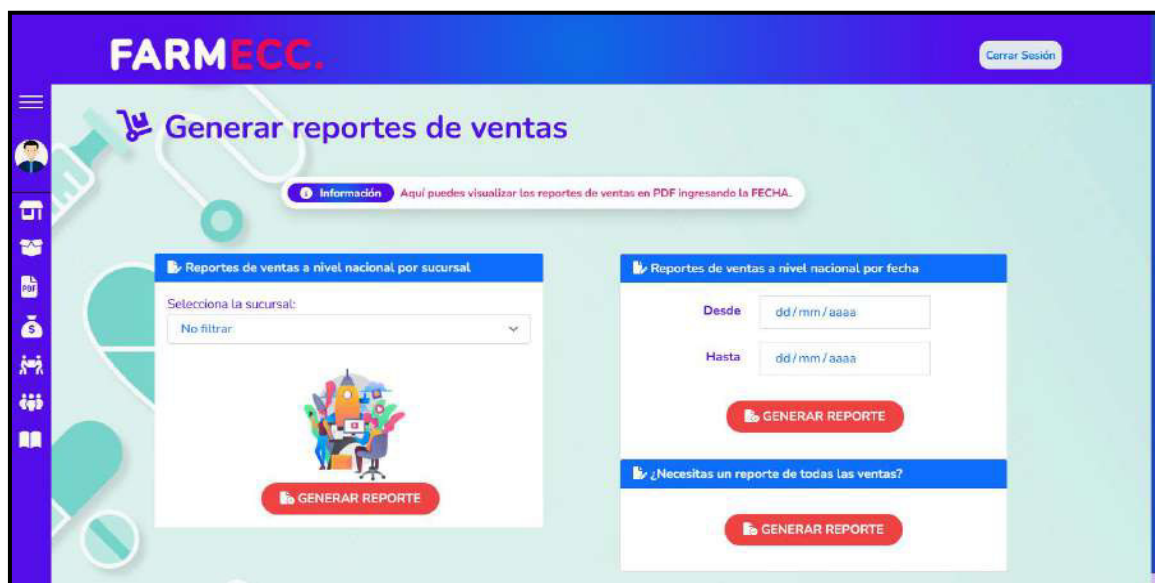


Fig.13: Página del módulo personal del sistema web.

## Consumir varios *endpoints* para visualizar reportes de cada sucursal

Para el módulo de visualización de reportes de sucursales se ha implementado directamente un ícono que permite la visualización y descarga del reporte que se elija en base a su búsqueda de acuerdo a una categoría: sucursal, fecha. Esto ha permitido el consumo de una ruta de tipo GET por medio de los *endpoints* proporcionados por la aplicación de escritorio para la visualización de los reportes por cada una de las sucursales de la cadena de farmacias, como se observa en la **Fig.14**. En este sentido, el proceso que detalla tanto el consumo y visualización se encuentra en el **ANEXO III** del actual documento.



**Fig.14: Página del módulo reportes del sistema web.**

## Consumir varios *endpoints* para gestionar pedidos con proveedores

Para el módulo de gestión de pedidos con proveedores se ha implementado directamente un formulario que permite el ingreso de los datos del proveedor, el producto y cantidad que se desea, adicionalmente, es posible la modificación del pedido antes mencionado, así como su eliminación. Esto ha permitido el consumo de una ruta de tipo GET por medio de los *endpoints* proporcionados por la aplicación de escritorio para gestión de pedidos independientemente de la sucursal en la que se encuentre como se observa en la **Fig.15**: Página del módulo pedidos con proveedores.. En este sentido, el proceso que detalla tanto el consumo y visualización se encuentra en el **ANEXO III** del actual documento.



Fig.15: Página del módulo pedidos con proveedores.

### 3.3 *Sprint 2. Resultado de las interfaces para el usuario cliente.*

La actividad dedicada al *Sprint 2* tienen que ver con:

- Consumir varios *endpoints* para visualizar productos.
- Consumir varios *endpoints* para realizar cotizaciones de productos.

#### **Consumir varios *endpoints* para visualizar productos**

Para el módulo de visualizar productos se ha implementado una tabla que permite visualizar la información de los productos de todas las sucursales. Esto ha permitido el consumo de una ruta de tipo GET por medio de los *endpoints* proporcionados por la aplicación de escritorio para la visualización de productos disponibles como se observa en la **Fig.16**. En este sentido, el proceso que detalla tanto el consumo y visualización se encuentra en el **ANEXO III** del actual documento.

Selecciona la sucursal: FARMECC 1

Busca el producto: Producto...

### Resultados de la búsqueda: FARMECC 1

Nombre Genérico	Nombre Comercial	Descripción	Registro sanitario	Precio de venta	Precio de compra	Cantidad	Sucursal	Stock
TOALLITAS HUMEDAS	PAÑALIN TOALLITAS HUMEDAS	x50	341040	1	0.68	6	FARMECC 1	POR AGOTARSE
TOALLAS INTIMAS	NOSOTRAS BÁSICAS CON ALAS	x10		1.4	1.15	9	FARMECC 1	POR AGOTARSE
TOALLAS INTIMAS X10	NOSOTRAS INVISIBLES RAPIGEL	x10		1.65	1.45	5	FARMECC 1	POR AGOTARSE
PAPEL HIGIÉNICO	FLOR CLÁSICO	x6 Rollos		1	0.89	6	FARMECC 1	POR AGOTARSE

**Fig.16: Página del módulo *stock* de productos para visualizar productos por sucursal.**

**Consumir varios *endpoints* para realizar cotizaciones de productos.**

Para el módulo de realización de cotizaciones para usuarios con el perfil cliente, se ha implementado un buscador que permite al cliente encontrar su producto y agregarlo a la cotización para generar su valor final. Esto ha permitido el consumo de una ruta pública de tipo GET por medio de los *endpoints* proporcionados por la aplicación de escritorio para la realización de cotizaciones como se observa en la **Fig. 17**. En este sentido, el proceso que detalla tanto el consumo y visualización se encuentra en el **ANEXO III** del actual documento.

**FARMECC.** INICIO CATALOGO DOCTORES Iniciar Sesión

## CATALOGO DE PRODUCTOS

Busca un producto: Producto...

Revisa el catálogo completo

PRODUCTO	PRECIO
PAÑALIN TOALLITAS HUMEDAS	1
NOSOTRAS BÁSICAS CON ALAS	1.4
<b>TOTAL</b>	<b>\$ 2.4</b>

VACIAR IMPRIMIR

**Fig.17: Página del módulo catálogo para realizar cotizaciones.**



### 3.4 *Sprint 3. Pruebas en el sistema web.*

Finalizada la fase de codificación de los componentes visuales por parte del sistema web, se detallan las actividades para el *Sprint 3* tienen que ver con:

- Ejecución de las pruebas unitarias y resultados.
- Ejecución de las pruebas de rendimiento y resultados.
- Ejecución de las pruebas de compatibilidad y resultados.
- Ejecución de las pruebas de aceptación y resultados.

#### **Ejecución de las pruebas unitarias y resultados**

Las pruebas unitarias o *unit testing* son un tipo de prueba de *software* que consiste en verificar el funcionamiento correcto de los componentes individuales con el propósito de validar cada una de estas unidades [45]. En este aspecto, dado que Laravel incluye un entorno de pruebas previamente configurado, se utiliza el *Framework* de *testing PHP Unit*, el cual se trata de una biblioteca independiente que facilita la refactorización y depuración del código sin el riesgo de actualizaciones que alteren la funcionalidad del *software* [46].

La **Fig. 18** presenta una parte del código para la apertura de la página principal de sistema *web*, tomando en cuenta su componente en *Livewire*, el cual ha sido implementado para una de las pruebas, adicional, en la **Fig. 19** se puede observar el resultado posterior a la ejecución de la prueba antes mencionada. La especificación completa de esta ejecución, así como los resultados del resto de pruebas se encuentra en el **ANEXO II** del actual documento.

```
4
5 use Illuminate\Foundation\Testing\RefreshDatabase;
6 use Tests\TestCase;
7
8 class ExampleTest extends TestCase
9 {
10     /**
11      * A basic test example.
12      *
13      * @return void
14      */
15     public function test_abrir()
16     {
17         $response = $this->get('/');
18
19         $response->assertStatus(200);
20     }
21 }
```

**Fig. 18: Fragmento de código para abrir la página principal del sistema *web*.**

```
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb> vendor/bin/phpunit --filter test_abrir
PHPUnit 9.5.20 #StandWithUkraine
-
1 / 1 (100%)

Time: 00:05.524, Memory: 24.00 MB

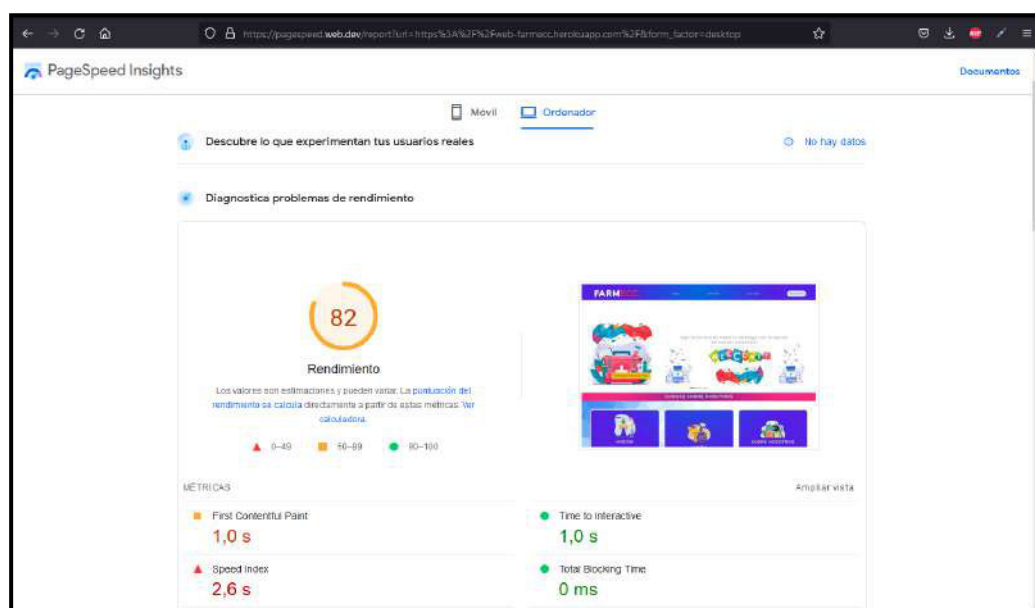
OK (1 test, 1 assertion)
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb>
```

**Fig. 19: Resultado de la prueba.**

Culminada la etapa de pruebas unitarias y basándose en los resultados que se han obtenido, se establece que los módulos del sistema *web* funcionan correctamente, sin problemas a nivel de código y validaciones respectivamente.

### **Ejecución de las pruebas de rendimiento y resultados**

Las pruebas de rendimiento hacen posible la identificación de tiempos de respuesta de un software. Para realizar este proceso en el presente sistema *web* y componentes internos, se ha utilizado la herramienta “*PageSpeed Insights*” como se observa a continuación.



**Fig. 20: Resultado de la prueba de rendimiento.**

Culminada esta etapa de pruebas de rendimiento y basándose en los resultados que se han obtenido, se establece que el sistema *web* funciona correctamente y de manera óptima.

### **Ejecución de las pruebas de compatibilidad y resultados**

Las pruebas de compatibilidad, permite verificar si un sistema es capaz de ejecutarse correctamente en diferentes entornos, ya sea *hardware*, sistemas operativos o aplicaciones [47]. Por ello, en la **TABLA VIII** se presenta los navegadores que se han



utilizado para comprobar si el contenido es adaptable a cualquier dispositivo. La descripción tanto de las ejecuciones como los resultados del resto de pruebas de compatibilidad se encuentran en el **ANEXO II** del actual documento.

**TABLA VIII: Navegadores que han sido utilizados para pruebas de compatibilidad.**

NOMBRE	VERSIÓN	OBSERVACIÓN
<i>Mozilla Firefox</i>	101.0.1	Completamente funcional
<i>Microsoft Edge</i>	102.0.1245.44	Completamente funcional
<i>Google Chrome</i>	102.0.5005.115	Completamente funcional

Culminada la etapa de pruebas de compatibilidad y basándose en los resultados que se han obtenido en las ejecuciones de las respectivas pruebas en los navegadores, se contempla que el sistema *web* es compatible con cada uno de estos, tanto en funcionamiento como en visibilidad del contenido sin presentar ningún tipo de error.

### **Ejecución de las pruebas de aceptación y resultados**

Las pruebas de aceptación si bien son una de las partes finales del proceso de pruebas del *software*, es la etapa más importante, pues en esta se garantiza que el sistema desarrollado es lo suficientemente intuitivo para que un usuario navegue sin problemas, cubra sus expectativas y satisfaga sus necesidades [48]. En base a los requerimientos inicialmente solicitados por los directivos de la cadena de farmacias FARMECC, en la **TABLA IX** se muestra el detalle de la primera prueba de aceptación conjuntamente con su resultado, la descripción de las demás pruebas se encuentra en el **ANEXO II** del actual documento junto con sus resultados.

**TABLA IX: Prueba de Aceptación N°1 – Consumir varios *endpoints* para visualizar la página informativa.**

PRUEBA DE ACEPTACIÓN	
Identificador (ID): PA001	Identificador historia de Usuario: HU001
Nombre: Consumir varios <i>endpoints</i> para visualizar la página informativa.	

<p><b>Descripción:</b> El usuario administrador y cliente necesitan consumir varios endpoints para visualizar una página informativa de la red de Farmacias FARMECC para conocer sobre la empresa (historia, visión y misión) e interactuar con los módulos:</p> <ul style="list-style-type: none"> <li>• Inicio</li> <li>• Doctores</li> </ul>
<p><b>Pasos de ejecución:</b></p> <ul style="list-style-type: none"> <li>• Dirigirse a la URL del sistema <i>web</i> en el navegador.</li> <li>• Seleccionar el módulo “Inicio”, en el panel superior para visualizar la información general, misión y visión.</li> <li>• Seleccionar el módulo “Doctores”, en el panel superior para visualizar la información de los médicos disponibles.</li> </ul>
<p><b>Resultado deseado:</b></p> <p>El sistema <i>web</i> permite visualizar información general de la cadena de farmacias FARMECC.</p>
<p><b>Evaluación de la prueba:</b></p> <p>El cliente aprueba al 100% con la verificación de los resultados esperados.</p>

Culminada la etapa de pruebas de aceptación y basándose en los resultados que se han obtenido, se contempla una aceptación del 100% de parte de los usuario clientes, tanto en funcionalidad como en interacción entre los módulos presentados por el sistema *web*. En este sentido se verifica el cumplimiento de los requerimientos iniciales, dando paso a la siguiente etapa.

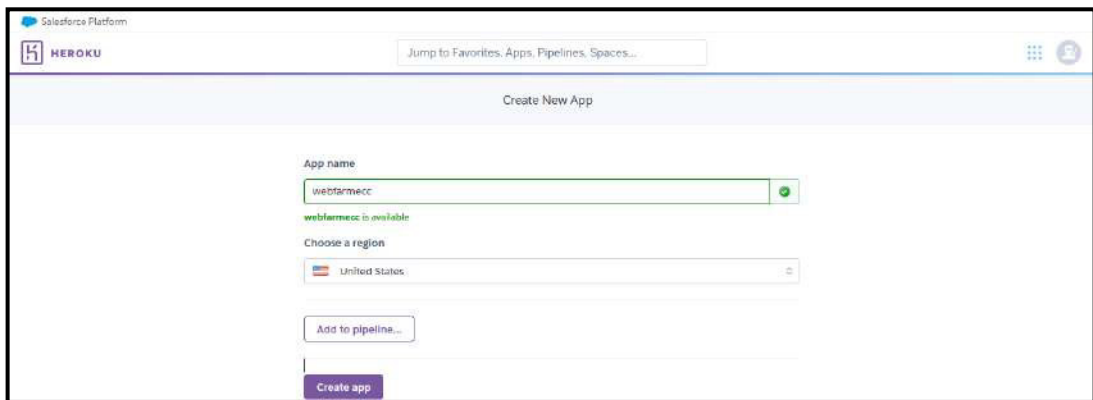
### 3.5 *Sprint* 4. Despliegue del sistema web.

Tras la finalización de la etapa de codificación y pruebas de los diferentes módulos visuales por parte del sistema *web*, las actividades para el *Sprint* 4 tienen que ver con el despliegue del sistema web en *Heroku*.

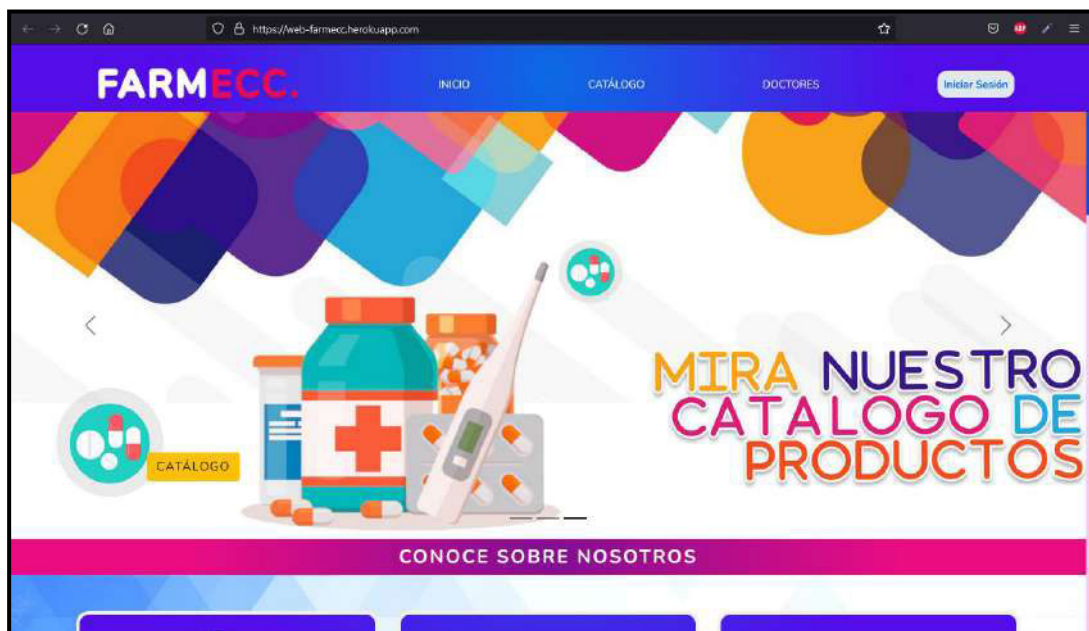
#### **Despliegue del sistema web en *Heroku***

En este apartado se describe el proceso que se ha realizado para el despliegue a producción del sistema *web* utilizando la plataforma *Heroku*. El detalle de esta sección se encuentra en el **ANEXO IV** del actual documento. La etapa inicial de este proceso se observa en la **Fig. 21** correspondiente a la creación de un nuevo proyecto en *Heroku* para alojar el sistema *web*, por otro lado, la **Fig. 22** presenta el despliegue apropiado del sistema *web* bajo un dominio proporcionado por la plataforma.

<https://web-farmecc.herokuapp.com/>



**Fig. 21: Creación de nuevo proyecto en Heroku.**



**Fig. 22: Despliegue del sistema web en Heroku**

Finalmente, los directivos de la cadena de farmacias FARM ECC han generado un certificado confirmando el acatamiento de todos los requerimientos solicitados inicialmente, así como de las funcionalidades del sistema web. El certificado se adjunta en el **ANEXO II**.

## 4 CONCLUSIONES

A continuación, se presentan las conclusiones que se han obtenido a lo largo del desarrollo del actual trabajo de integración curricular.

- El sistema *web* cumple tanto con el alcance del proyecto, como con los objetivos previamente establecidos, dando la posibilidad de que la cadena de farmacias FARMECC gracias a la tecnología cuente con un sistema web de ayuda para la gestión del inventario de sus productos de manera que organice sus ventas y genere nuevos ingresos.
- El sistema web ha sido implementado en base a los requerimientos iniciales, donde se estableció las necesidades de los usuarios y con ello se llevó a cabo su desarrollo.
- La metodología ágil *Scrum* que se ha utilizado para el proceso de desarrollo del presente trabajo de integración curricular el cual dio apertura a que el sistema pueda ser completado etapa por etapa y validando las funcionalidades en los distintos periodos de tiempo que se han planificado.
- Guiarse del patrón arquitectónico modelo-vista-controlador hizo posible el consumo correcto de los *endpoints* generados por la aplicación de escritorio para que puedan ser presentados en la interfaz del sistema *web*.
- Durante la fase de codificación el uso de las herramientas y bibliotecas para el desarrollo del sistema web, fue de gran importancia pues de esa manera se ha obtenido un buen resultado en cuanto al funcionamiento del mismo.
- Las distintas fases de pruebas para el sistema *web* han permitido verificar la eficiencia del sistema tanto en módulos como en funcionalidades y comprobar si realmente se cumple con los requerimientos que se han establecido.

## 5 RECOMENDACIONES

A continuación, se presentan las recomendaciones obtenidas a lo largo del desarrollo del actual trabajo de integración curricular.

- Se recomienda que el sistema *web* que se ha desarrollado en este trabajo de integración curricular sea reforzado con nuevas funcionalidades y seguridades, en caso de aumentar el número de sucursales en distintos lugares del país de la cadena de farmacias FARMECC.
- En caso que el usuario administrador requiera cambiar o actualizar sus credenciales de inicio de sesión en el sistema *web*, se recomienda comunicarse con el equipo de desarrolladores del presente proyecto.
- Con el fin de salvaguardar la información importante sobre las ventas realizadas, se recomienda realizar la descarga de los reportes generados cada cierto periodo de tiempo, así como un *backup* de la Base de Datos.
- Si se requiere actualizar algún módulo del sistema, se recomienda comunicarse con el equipo de desarrollo para en conjunto con las nuevas necesidades lograr mantener actualizado el sistema web.

## 6 REFERENCIAS BIBLIOGRÁFICAS

- [1] UNLEASHED, «Gestión de inventario en línea: qué es y por qué es importante,» 12 abril 2021. [En línea]. Available: <https://www.unleashedsoftware.com/blog/online-inventory-management-what-it-is-and-why-its-important#:~:text=Online%20inventory%20management%20is%20exactly,outward%20flow%20of%20inventory%20stock..> [Último acceso: 27 febrero 2022].
- [2] Brightpearl, «¿Qué es un sistema de gestión de inventario?,» 2020. [En línea]. Available: <https://www.brightpearl.com/inventory-management-system>. [Último acceso: 2022 febrero 27].
- [3] sales layer, «7 problemas comunes de gestión de inventario en línea y soluciones en empresas industriales,» 09 octubre 2018. [En línea]. Available: <https://blog.saleslayer.com/7-common-online-inventory-management-problems-solutions-industrial-companies>. [Último acceso: 2022 febrero 27].
- [4] S. University, «Information and Computer Technology: Web Systems,» 11 05 2022. [En línea]. Available: <https://libguides.sullivan.edu/ICT/websites>. [Último acceso: 13 05 2022].
- [5] MYRA Technolabs, «Importancia del desarrollo front-end para el éxito empresarial,» 04 03 2021. [En línea]. Available: <https://www.myratechnolabs.com/blog/front-end-development-for-business/>. [Último acceso: 18 05 2022].
- [6] BBN TIMES, «La importancia del Front-end en todo proyecto de diseño web,» 05 10 2018. [En línea]. Available: <https://www.bbntimes.com/technology/the-importance-of-front-end-in-every-web-design-project>. [Último acceso: 18 05 2022].
- [7] FULCRUM, «¿Cómo crear un prototipo para su proyecto de diseño?,» 01 03 2021. [En línea]. Available: <https://fulcrum.rocks/blog/how-to-create-a-prototype/>. [Último acceso: 18 05 2022].
- [8] C. Egbude, «Figma: Todo lo que necesitas saber,» 19 04 2021. [En línea]. Available: <https://medium.com/@tobiiegbude/figma-all-you-need-to-know-9f8ac90ff49d>. [Último acceso: 18 05 2022].
- [9] builtin, «¿Qué es Figma?,» [En línea]. Available: <https://builtin.com/learn/figma>. [Último acceso: 18 05 2022].
- [10] SidmartinBio, «What is web-based system?,» 04 05 2020. [En línea]. Available: <https://www.sidmartinbio.org/what-is-web-based-system/>. [Último acceso: 13 05 2022].
- [11] Camcode, «¿Qué es un sistema de gestión de inventario?,» 18 02 2022. [En línea]. Available: <https://www.camcode.com/blog/what-is-an-inventory-management-system/>. [Último acceso: 13 05 2022].
- [12] Write, «¿Qué es la metodología ágil en la gestión de proyectos?,» [En línea]. Available: [https://www-wrike-com.translate.google/project-management-guide/faq/what-is-agile-methodology-in-project-management/?\\_x\\_tr\\_sl=en&\\_x\\_tr\\_tl=es&\\_x\\_tr\\_hl=es&\\_x\\_tr\\_pto=sc](https://www-wrike-com.translate.google/project-management-guide/faq/what-is-agile-methodology-in-project-management/?_x_tr_sl=en&_x_tr_tl=es&_x_tr_hl=es&_x_tr_pto=sc). [Último acceso: 13 05

2022].

- [13] P. Jackson, «What is PHP? Write your first PHP Program,» 16 04 2022. [En línea]. Available: <https://www.guru99.com/what-is-php-first-php-program.html>. [Último acceso: 14 05 2022].
- [14] S. Ahmed, «How to Use Composer in Laravel 8,» 03 09 2021. [En línea]. Available: <https://www.cloudways.com/blog/composer-with-laravel/>. [Último acceso: 14 05 2022].
- [15] D. Ocean, «What is Laravel?,» 03 02 2021. [En línea]. Available: <https://www.digitalocean.com/community/tutorials/what-is-laravel>. [Último acceso: 14 05 2022].
- [16] C. Tim, «What is Laravel? Explain it like I'm five.,» 11 12 2019. [En línea]. Available: [https://dev.to/creativetim\\_official/what-is-laravel-explain-it-like-i-m-five-19eb](https://dev.to/creativetim_official/what-is-laravel-explain-it-like-i-m-five-19eb). [Último acceso: 14 05 2022].
- [17] aws-amazon, «What is an API?,» 2022. [En línea]. Available: [https://aws-amazon-com.translate.goog/what-is/api/?\\_x\\_tr\\_sl=en&\\_x\\_tr\\_tl=es&\\_x\\_tr\\_hl=es&\\_x\\_tr\\_pto=sc](https://aws.amazon.com.translate.goog/what-is/api/?_x_tr_sl=en&_x_tr_tl=es&_x_tr_hl=es&_x_tr_pto=sc). [Último acceso: 14 05 2022].
- [18] <ORDINARY>CODERS, «Bootstrap vs Materialize CSS,» 07 04 2021. [En línea]. Available: <https://ordinarycoders.com/blog/article/bootstrap-vs-materialize-css>. [Último acceso: 18 05 2022].
- [19] testiM, «Pruebas de front-end: una descripción conceptual completa,» 11 12 2019. [En línea]. Available: <https://www.testim.io/blog/front-end-testing-complete-overview/>. [Último acceso: 18 05 2022].
- [20] PERFORCE, «Una guía completa para las pruebas de front-end,» 22 07 2020. [En línea]. Available: <https://www.perfecto.io/blog/comprehensive-guide-front-end-testing>. [Último acceso: 18 05 2022].
- [21] Scribbr, «Estudio de caso | Definición, ejemplos y métodos,» 08 05 2019. [En línea]. Available: <https://www.scribbr.com/methodology/case-study/>. [Último acceso: 18 05 2022].
- [22] YOJJI, «Las metodologías de desarrollo de software más comunes,» 30 07 2021. [En línea]. Available: <https://yojji.io/blog/the-most-common-software-development-methodologies>. [Último acceso: 18 05 2022].
- [23] indeed, «11 metodologías de desarrollo de software (además de cómo elegir una),» 29 03 2022. [En línea]. Available: <https://www.indeed.com/career-advice/career-development/software-development-methodologies>. [Último acceso: 18 05 2022].
- [24] Guru99, «Metodología Ágil: ¿Qué es el Modelo Ágil en las Pruebas de Software?,» 23 04 2022. [En línea]. Available: <https://www.guru99.com/agile-scrum-extreme-testing.html>. [Último acceso: 18 05 2022].
- [25] InfoWorld, «¿Qué es la metodología ágil? Desarrollo de software moderno explicado,» 06 04 2022. [En línea]. Available: <https://www.infoworld.com/article/3237508/what-is-agile-methodology-modern-software-development-explained.html>. [Último acceso: 18 05 2022].
- [26] indeed, «Cómo definir los roles y responsabilidades de su equipo,» 09 06 2021. [En línea]. Available: <https://www.indeed.com/career-advice/career-development/team-roles>.

- and-responsibilities#:~:text=What%20are%20roles%20and%20responsibilities,several%20tasks%20in%20the%20workplace.. [Último acceso: 19 05 2022].
- [27] Mohamed Raza, «Funciones y responsabilidades ágiles,» 13 05 2019. [En línea]. Available: <https://www.bmc.com/blogs/agile-roles-responsibilities/>. [Último acceso: 18 05 2022].
- [28] C. Harris, «Artefactos ágiles de scrum,» 2022. [En línea]. Available: <https://www.atlassian.com/agile/scrum/artifacts#:~:text=Summary%3A%20Agile%20scrum%20artifacts%20are,%2C%20sprint%20backlog%2C%20and%20increments..> [Último acceso: 19 05 2022].
- [29] P. Parsi, «¿Qué son los artefactos de Scrum y cómo los uso?,» 23 06 2021. [En línea]. Available: <https://niftypm.com/blog/scrum-artifacts/>. [Último acceso: 19 05 2022].
- [30] K. Boogaard, «Lo que necesita saber sobre la recopilación de requisitos,» 06 05 2022. [En línea]. Available: <https://www.wrike.com/blog/requirements-gathering-guide/#What-is-requirements-gathering-in-project-management>. [Último acceso: 19 05 2022].
- [31] MOUNTAIN GOAT SOFTWARE, «What is a user story?,» [En línea]. Available: <https://www.mountaingoatsoftware.com/agile/user-stories>. [Último acceso: 19 05 2022].
- [32] PremierAgile, «Product Backlog Vs. Sprint Backlog,» [En línea]. Available: <https://premieragile.com/product-backlog-vs-sprint-backlog/>. [Último acceso: 19 05 2022].
- [33] Interaction Design Foundation, «¿Qué es el diseño de interfaz de usuario?,» [En línea]. Available: <https://www.interaction-design.org/literature/topics/ui-design>. [Último acceso: 19 05 2022].
- [34] A. MANCHANDA, «What is Software Architecture in Software Engineering?,» 12 07 2021. [En línea]. Available: <https://www.netsolutions.com/insights/why-software-architecture-matters-to-build-scalable-solutions/>. [Último acceso: 19 05 2022].
- [35] Indeed Editorial Team, «What Is Web Architecture? (With Definition and Tips),» 26 08 2021. [En línea]. Available: <https://www.indeed.com/career-advice/finding-a-job/what-is-web-architecture#:~:text=Web%20architecture%20is%20the%20process,particular%20purpose%2C%20company%20or%20brand..> [Último acceso: 19 05 2022].
- [36] R. D. Hernandez, «The Model View Controller Pattern – MVC Architecture and Frameworks Explained,» 19 04 2021. [En línea]. Available: <https://www.freecodecamp.org/news/the-model-view-controller-pattern-mvc-architecture-and-frameworks-explained/>. [Último acceso: 19 05 2022].
- [37] GeeksforGeeks, «MVC Design Pattern,» 08 02 2018. [En línea]. Available: <https://www.geeksforgeeks.org/mvc-design-pattern/>. [Último acceso: 19 05 2022].
- [38] R. A. Vera, «Laravel - qué es, características y ventajas,» 31 03 2021. [En línea]. Available: <https://openwebinars.net/blog/que-es-laravel-caracteristicas-y-ventajas/>. [Último acceso: 19 05 2022].
- [39] axarnet, «Bootstrap: qué es y cómo funcione este framework,» [En línea]. Available:



- <https://axarnet.es/blog/bootstrap>. [Último acceso: 19 05 2022].
- [40] Quora, «¿Qué es Materializar CSS?,» 2016. [En línea]. Available: <https://www.quora.com/What-is-Materialize-CSS>. [Último acceso: 19 05 2022].
- [41] Styde, «Compilación De Assets (Laravel Mix) – Documentación de Laravel 6,» 29 05 2020. [En línea]. Available: <https://styde.net/laravel-6-doc-compilacion-de-assets-laravel-mix/>. [Último acceso: 19 05 2022].
- [42] definitions.net, «Definitions for heroku,» [En línea]. Available: <https://www.definitions.net/definition/heroku>. [Último acceso: 19 05 2022].
- [43] D. Pacheco, «Autenticación de Laravel 8 con el tutorial de Laravel UI,» [En línea]. Available: <https://diegooo.com/laravel-ui-autenticacion-de-laravel-8/>. [Último acceso: 26 05 2022].
- [44] Programacion.net, «10 paquetes imprescindibles para Laravel que no debes dejar pasar,» [En línea]. Available: [https://programacion.net/articulo/10\\_paquetes\\_imprescindibles\\_para\\_laravel\\_que\\_no\\_debes\\_dejar\\_pasar\\_1914](https://programacion.net/articulo/10_paquetes_imprescindibles_para_laravel_que_no_debes_dejar_pasar_1914). [Último acceso: 19 05 2022].
- [45] T. Hamilton, «Unit Testing Tutorial: What is, Types, Tools & Test EXAMPLE,» 16 04 2022. [En línea]. Available: <https://www.guru99.com/unit-testing-guide.html>. [Último acceso: 16 06 2022].
- [46] Andy Carter, «PHPUnit: ¿Qué, por qué, cómo?,» 21 01 2018. [En línea]. Available: <https://andy-carter.com/blog/phpunit-what-why-how>. [Último acceso: 16 06 2022].
- [47] T. Hamilton, «What is Compatibility Testing? Forward & Backward Testing (Example),» 30 04 2022. [En línea]. Available: <https://www.guru99.com/compatibility-testing.html>. [Último acceso: 16 06 2022].
- [48] Performance Lab, «¿Qué es la prueba de aceptación del usuario (UAT) para sitios web? Definición y ejemplos,» [En línea]. Available: <https://performancelabus.com/what-is-uat-for-websites/>. [Último acceso: 16 06 2022].
- [49] L. Carvajal, Metodología de la Investigación Científica. Curso general y aplicado, 28 ed., Santiago de Cali: U.S.C., 2006, p. 139.
- [50] Orderhive, «STOCK REPLENISHMENT,» 01 abril 2020. [En línea]. Available: <https://www.orderhive.com/blog/stock-replenishment>. [Último acceso: 2022 febrero 27].
- [51] asset, «Why Inventory Management Plays a Crucial Role in the Logistics Industry?,» 09 septiembre 2020. [En línea]. Available: <https://www.assetinfinity.com/blog/logistics-inventory-management#:~:text=Inventory%20logistics%20management%20is%20an,flow%20and%20storage%20of%20goods..> [Último acceso: 27 febrero 2022].
- [52] IGI Global, «What is Continuous Review System,» [En línea]. Available: <https://www.igi-global.com/dictionary/continuous-review-system/41832>. [Último acceso: 27 febrero 2022].
- [53] Wasp, «Inventory Management 101,» [En línea]. Available: <https://www.waspbarcode.com/buzz/inventory-management-101>. [Último acceso: 2022

febrero 27].

- [54] CHRON, «What Is the Difference Between a Periodic and Continuous Inventory Review Policy?,» 31 enero 2019. [En línea]. Available: <https://smallbusiness.chron.com/difference-between-periodic-continuous-inventory-review-policy-30967.html>. [Último acceso: 2022 marzo 05].
- [55] IndustryStar, «Logistics & Inventory Management: Intentional & Coordinated Actions,» 28 marzo 2016. [En línea]. Available: <https://www.industrystar.com/blog/2016/03/logistics-inventory-management/>. [Último acceso: 05 marzo 2022].
- [56] ScriptPro, «Gestión de Inventario de Farmacia: Oferta y Demanda de Medicamentos,» 05 marzo 2019. [En línea]. Available: <https://www.scriptpro.com/blog/pharmacy-inventory-management-supply-and-demand-of-medication>. [Último acceso: 05 marzo 2022].

## 7 ANEXOS

A continuación, se muestra la división de los Anexos utilizados para el desarrollo del sistema *web*.

- **ANEXO I.** Resultado del programa antiplagio *Turnitin*.
- **ANEXO II.** Manual Técnico.
- **ANEXO III.** Manual de Usuario.
- **ANEXO IV.** Manual de Instalación

## ANEXO I

A continuación, se presenta el certificado que el Director de Tesis ha emitido y en donde se evidencia el resultado que se ha obtenido en la herramienta antiplagio Turnitin.



**ESCUELA POLITÉCNICA NACIONAL  
ESCUELA DE FORMACIÓN DE TECNÓLOGOS  
CAMPUS POLITÉCNICO "ING. JOSÉ RUBÉN ORELLANA"**

### CERTIFICADO DE ORIGINALIDAD

Quito, D.M. 22 de agosto de 2022

De mi consideración:

Yo, Loarte Cajamarca Byron Gustavo, en calidad de Director del Trabajo de Integración Curricular titulado Desarrollo de un sistema web asociado al DESARROLLO DE SISTEMA PARA LA GESTIÓN DEL INVENTARIO EN FARMECC elaborado por la estudiante Diana Katherine Almeida Anchatuña de la carrera en Tecnología Superior en Desarrollo de Software, certifico que he empleado la herramienta Turnitin para la revisión de originalidad del documento escrito secciones: Descripción del componente desarrollado, Metodología, Resultados, Conclusiones y Recomendaciones, producto del Trabajo de Integración Curricular indicado.

El documento escrito tiene un índice de similitud del 10%.

Es todo cuanto puedo certificar en honor a la verdad, pudiendo el interesado hacer uso del presente documento para los trámites de titulación.

NOTA: Se adjunta el informe generado por la herramienta Turnitin.

Atentamente,

**Loarte Cajamarca Byron Gustavo  
Profesor Ocasional a Tiempo Completo  
Escuela de Formación de Tecnólogos**

## ANEXO II

### Recopilación de requerimientos

En la **TABLA X** se muestra los requerimientos que se han obtenido al inicio del proyecto de acuerdo con lo solicitado por el dueño del producto.

**TABLA X: Recopilación de requerimientos.**

Recopilación de requerimientos		
Tipo de sistema	ID-RR	Enunciado del Ítem
<b>Sistema web</b>	<b>RR003</b>	Como usuario administrador necesita consumir varios <i>endpoints</i> para: <ul style="list-style-type: none"> <li>• Gestionar sucursales.</li> </ul>
	<b>RR004</b>	Como usuario administrador y usuario cliente, necesitan consumir varios <i>endpoints</i> para: <ul style="list-style-type: none"> <li>• Visualizar productos.</li> </ul>
	<b>RR005</b>	Como usuario administrador necesita consumir varios <i>endpoints</i> para: <ul style="list-style-type: none"> <li>• Visualizar reportes de sucursales.</li> </ul>
	<b>RR006</b>	Como usuario administrador necesita consumir varios <i>endpoints</i> para: <ul style="list-style-type: none"> <li>• Visualizar la disponibilidad de stock de productos.</li> </ul>
	<b>RR007</b>	Como usuario administrador, necesita consumir un <i>endpoint</i> para <ul style="list-style-type: none"> <li>• Visualizar las ventas por sucursales.</li> </ul>
	<b>RR008</b>	Como usuario administrador, necesita consumir un <i>endpoint</i> para <ul style="list-style-type: none"> <li>• Gestionar pedidos con proveedores.</li> </ul>
	<b>RR009</b>	Como usuario administrador necesita consumir varios <i>endpoints</i> para: <ul style="list-style-type: none"> <li>• Visualizar personal.</li> </ul>
	<b>RR010</b>	Como usuario cliente necesita consumir varios <i>endpoints</i> para:

		<ul style="list-style-type: none"> <li>Realizar cotización.</li> </ul>
--	--	--

## Historias de Usuario

Culminada la etapa de Recopilación de requerimientos, se procede a desarrollar cada una de las Historias de Usuario para el sistema *web*. En ese sentido, se presentan las 09 Historias de Usuario escritas en base a los requerimientos del *Product Owner* que van desde la **TABLA XII** hasta la **TABLA XX**.

**TABLA XII: Historia de usuario 02 – Consumir endpoints para iniciar sesión y cerrar sesión.**

HISTORIA DE USUARIO	
<b>Identificador (ID):</b> HU002	<b>Usuario:</b> Administrador.
<b>Nombre Historia:</b> Consumir <i>endpoints</i> para iniciar sesión y cerrar sesión.	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Iteración Asignada:</b> 2	
<b>Responsable (es):</b> Diana Almeida	
<b>Descripción:</b> El usuario administrador necesita consumir varios <i>endpoints</i> para iniciar sesión y cerrar sesión. Al ingresar a la opción de “administrador” de la página inicial se despliega un formulario de inicio de sesión para ser llenado, para el cierre de sesión el administrador deberá seleccionar la opción “cerrar sesión” y para un posible cambio de contraseña se requiere que el solicitante se contacte con el equipo desarrollador por medio de las opciones presentadas en la página respectiva.	
<b>Observación:</b> El usuario administrador debe iniciar sesión para consumir los <i>endpoints</i> del cierre de sesión.	

**TABLA XIII: Historia de usuario 03 – Consumir varios endpoints para gestionar sucursales.**

HISTORIA DE USUARIO	
<b>Identificador (ID):</b> HU003	<b>Usuario:</b> Administrador
<b>Nombre Historia:</b> Consumir <i>endpoints</i> para gestionar sucursales.	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta

<b>Iteración Asignada: 2</b>
<b>Responsable (es):</b> Diana Almeida
<p><b>Descripción:</b> El usuario administrador tiene la posibilidad de crear, visualizar, modificar y eliminar información respecto a:</p> <ul style="list-style-type: none"> <li>• Sucursales.</li> </ul> <p>Para el listado, registro y modificación de una sucursal se lo realiza mediante un formulario donde se ingresa la siguiente información:</p> <ul style="list-style-type: none"> <li>• Nombre.</li> <li>• Ciudad.</li> <li>• Teléfono.</li> <li>• Dirección.</li> </ul>
<p><b>Observación:</b> Para eliminar una sucursal el usuario debe presionar el botón eliminar, con lo cual se elimina toda la información de la Base de Datos. No obstante, para dar de baja una sucursal no debe estar asociado ningún personal, productos, etc.</p>

**TABLA XIV: Historia de usuario 04 – Consumir varios endpoints para visualizar productos.**

<b>HISTORIA DE USUARIO</b>	
<b>Identificador (ID):</b> HU004	<b>Usuario:</b> Administrador y cliente
<b>Nombre Historia:</b> Consumir varios <i>endpoints</i> para visualizar productos.	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en desarrollo:</b> Media
<b>Iteración Asignada: 1</b>	
<b>Responsable (es):</b> Diana Almeida	
<p><b>Descripción:</b> El usuario administrador y cliente necesitan consumir varios endpoints para visualizar productos. A través de la página principal del sistema <i>web</i>, en el módulo “Catálogo” los usuarios tienen la posibilidad de ver todos los productos existentes.</p>	
<b>Observación:</b> Ninguna.	

**TABLA XV: Historia de usuario 05 – Consumir varios endpoints para visualizar reportes de sucursales.**

<b>HISTORIA DE USUARIO</b>	
<b>Identificador (ID):</b> HU005	<b>Usuario:</b> Administrador.

<b>Nombre Historia:</b> Consumir varios <i>endpoints</i> visualizar reportes de sucursales.	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en desarrollo:</b> Media
<b>Iteración Asignada:</b> 1	
<b>Responsable (es):</b> Diana Almeida	
<p><b>Descripción:</b> El usuario administrador necesita consumir varios <i>endpoints</i> para visualizar los reportes de ventas de las sucursales. En este sentido, es necesario ingresar al módulo “Reportes de sucursales” y seleccionar cómo se requiere generar el reporte:</p> <ul style="list-style-type: none"> <li>• Reporte por sucursal.</li> <li>• Reporte por rango de fechas (inicial y final).</li> <li>• Reporte general de ventas.</li> </ul> <p>posteriormente, se ingresa los datos requeridos respectivamente y se genera un reporte.</p>	
<p><b>Observación:</b> El ingreso de datos en el formulario es por selección, siendo las opciones presentadas las únicas válidas para la visualización de reportes.</p>	

**TABLA XVI: Historia de usuario 06 – Consumir varios endpoints para visualizar la disponibilidad de stock de productos.**

<b>HISTORIA DE USUARIO</b>	
<b>Identificador (ID):</b> HU006	<b>Usuario:</b> Administrador.
<b>Nombre Historia:</b> Consumir varios <i>endpoints</i> para visualizar la disponibilidad de <i>stock</i> de productos.	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Iteración Asignada:</b> 2	
<b>Responsable (es):</b> Diana Almeida	
<p><b>Descripción:</b> El usuario administrador necesita consumir varios <i>endpoints</i> para visualizar disponibilidad de stock de productos. Mediante el módulo “Stock de productos”, el usuario busca la sucursal que desea consultar, una vez seleccionada, se presenta los productos con los que cuenta y su estado: agotado, por agotarse y disponible.</p>	
<p><b>Observación:</b> La búsqueda de sucursales para esta función es por selección, siendo las opciones presentadas las únicas válidas para la visualización de disponibilidad de <i>stock</i> de productos. En caso de tener un producto agotado se tiene un botón auxiliar que permite dirigirse al módulo “Pedidos”.</p>	



**TABLA XVII: Historia de usuario 07 – Consumir varios endpoints para visualizar la venta por sucursales.**

<b>HISTORIA DE USUARIO</b>	
<b>Identificador (ID):</b> HU007	<b>Usuario:</b> Administrador.
<b>Nombre Historia:</b> Consumir varios <i>endpoints</i> para visualizar la venta por sucursales.	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en desarrollo:</b> Media
<b>Iteración Asignada:</b> 1	
<b>Responsable (es):</b> Diana Almeida	
<b>Descripción:</b> El usuario administrador necesita consumir varios <i>endpoints</i> para visualizar la venta de sucursales. Mediante el módulo “Ventas”, el usuario busca la sucursal que desea consultar, una vez seleccionada, se presenta una lista con las ventas realizadas junto con las descripciones de ventas, fechas, valores recaudados por factura o comprobante de venta y al final, un total de las ventas de la sucursal seleccionada.	
<b>Observación:</b> La búsqueda de sucursales para esta función es por selección, siendo las opciones presentadas las únicas válidas para la visualización de ventas por sucursales. En esta función, el usuario administrador no puede realizar ningún tipo de modificación.	

**TABLA XVIII: Historia de usuario 08 – Consumir varios endpoints para gestionar pedidos con proveedores.**

<b>HISTORIA DE USUARIO</b>	
<b>Identificador (ID):</b> HU008	<b>Usuario:</b> Administrador.
<b>Nombre Historia:</b> Consumir varios <i>endpoints</i> para gestionar pedidos con proveedores.	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Iteración Asignada:</b> 2	
<b>Responsable (es):</b> Diana Almeida	
<b>Descripción:</b> El usuario administrador necesita consumir varios endpoints para gestionar pedidos con proveedores. En este sentido, el administrador tiene la posibilidad de crear, visualizar, modificar y eliminar información respecto a:	

<ul style="list-style-type: none"> <li>• Pedidos con proveedores.</li> <li>• Proveedores</li> </ul> <p>Para el registro y modificación de pedidos se realiza mediante un formulario donde se ingresa la siguiente información:</p> <ul style="list-style-type: none"> <li>• Nombre del producto.</li> <li>• Nombre del proveedor.</li> <li>• Cantidad.</li> </ul>
<p><b>Observación:</b> El ingreso de datos en el formulario para proveedor por selección, siendo las opciones presentadas las únicas válidas para la visualización de proveedores. El número de pedidos por día no tiene un límite y al ser ejecutados serán actualizados dentro del stock de productos.</p>

**TABLA XIX: Historia de usuario 09 – Consumir varios endpoints para visualizar el personal.**

<b>HISTORIA DE USUARIO</b>	
<b>Identificador (ID):</b> HU009	<b>Usuario:</b> Administrador.
<b>Nombre Historia:</b> Consumir varios <i>endpoints</i> para visualizar el personal.	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en desarrollo:</b> Media
<b>Iteración Asignada:</b> 1	
<b>Responsable (es):</b> Diana Almeida	
<p><b>Descripción:</b> El usuario administrador necesita consumir varios <i>endpoints</i> para visualizar el personal de la red de farmacias FARMECC de forma actualizada. Mediante el módulo “Personal” el administrador puede visualizar la información de empleados, rol, experiencia laboral, contactos y horarios. Si desea el administrador, se puede categorizar la información de los empleados por sucursal. Para la búsqueda de empleados, es necesario el ingreso del nombre completo del empleado.</p>	
<p><b>Observación:</b> Para la categorización de la información de empleados, se presenta un menú de opciones, siendo estas las únicas opciones válidas de filtrado. El formulario de búsqueda de empleados cuenta únicamente con 2 opciones válidas.</p>	

**TABLA XX: Historia de usuario 10 – Consumir varios endpoints para realizar cotizaciones de productos.**

<b>HISTORIA DE USUARIO</b>	
<b>Identificador (ID):</b> HU010	<b>Usuario:</b> Cliente
<b>Nombre Historia:</b> Consumir varios <i>endpoints</i> para realizar cotizaciones de productos.	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Iteración Asignada:</b> 2	
<b>Responsable (es):</b> Diana Almeida	
<b>Descripción:</b> El usuario cliente necesita consumir varios endpoints para realizar cotizaciones de mercadería. Mediante el módulo “Cotizar” los clientes pueden hacer la búsqueda individual o acumulativa de productos y finalmente generar la cotización.	
<b>Observación:</b> El usuario cliente no necesita iniciar sesión para consumir los <i>endpoints</i> y poder realizar cotizaciones de productos. La búsqueda de productos es por selección, siendo las opciones presentadas las únicas válidas para realizar las cotizaciones.	

### **Product Backlog**

La **TABLA XXI** enumera la prioridad de cada requisito que se ha implementado en el sistema *web*. Estos requisitos se clasifican de acuerdo a las necesidades del dueño del producto y a la complejidad del desarrollo.

**TABLA XXI: Product Backlog.**

<b>ELABORACIÓN DEL PRODUCT BACKLOG</b>					
<b>ID – PB</b>	<b>ID – HU</b>	<b>HISTORIA DE USUARIO</b>	<b>ITERACIÓN</b>	<b>ESTADO</b>	<b>PRIORIDAD</b>
HU002	HU002	Consumir <i>endpoints</i> para iniciar sesión y cerrar sesión.	2	Finalizado	Alta
HU003	HU003	Consumir <i>endpoints</i> para gestionar sucursales.	2	Finalizado	Alta

HU004	HU004	Consumir varios <i>endpoints</i> para visualizar productos.	1	Finalizado	Media
HU005	HU005	Consumir varios <i>endpoints</i> para visualizar reportes de sucursales.	1	Finalizado	Media
HU006	HU006	Consumir varios <i>endpoints</i> para visualizar la disponibilidad de stock de productos.	2	Finalizado	Alta
HU007	HU007	Consumir varios <i>endpoints</i> para visualizar la venta por sucursales.	1	Finalizado	Media
HU008	HU008	Consumir varios <i>endpoints</i> para gestionar pedidos con proveedores.	2	Finalizado	Alta
HU009	HU009	Consumir varios <i>endpoints</i> para visualizar el personal.	1	Finalizado	Media
HU010	HU010	Consumir varios <i>endpoints</i> para realizar cotizaciones de productos.	2	Finalizado	Alta

## Sprint Backlog

La **TABLA XXII** presenta los cinco *Sprints* en los que se ha desarrollado el sistema *web*, listando las actividades y el tiempo determinado para cumplir con los entregables que se han establecido con el dueño del producto.

**TABLA XXII: *Sprint Backlog*.**

ELABORACIÓN DEL <i>SPRINT BACKLOG</i>						
ID – SB	NOMBRE	MÓDULO	ID-HU	HISTORIA DE USUARIO	TAREAS	TIEMPO ESTIMADO
SB001	Diseño e implementación de interfaz del usuario administrador y módulos a su cargo	Sección informativa	HU001	Consumir varios <i>endpoints</i> para visualizar la página informativa.	<ul style="list-style-type: none"><li>• Diseño e implementación de interfaz para la página informativa.</li><li>• Presentación de información proporcionada por parte del <i>backend</i>.</li><li>• Pruebas respectivas.</li></ul>	140 H

		Módulo inicio de sesión	HU002	Consumir <i>endpoints</i> para iniciar sesión, cerrar sesión.	<ul style="list-style-type: none"> <li>• Diseño e implementación de interfaz para el inicio y cierre de sesión del usuario.</li> <li>• Implementación de formularios para iniciar sesión y cerrar de sesión.</li> <li>• Implementación de formularios para el cambio de contraseña.</li> <li>• Validar campos requeridos.</li> <li>• Verificar credenciales del usuario.</li> <li>• Carga de módulos asignados.</li> <li>• Pruebas respectivas.</li> </ul>	
		Módulo gestión de sucursales	HU003	Consumir <i>endpoints</i> para gestionar sucursales.	<ul style="list-style-type: none"> <li>• Diseño e implementación de formulario y botones para crear, visualizar, modificar y eliminar sucursales.</li> <li>• Validar campos requeridos.</li> <li>• Gestionar datos por parte del backend.</li> <li>• Pruebas respectivas.</li> </ul>	

		Módulo productos	HU004	Consumir varios <i>endpoints</i> para visualizar productos.	<ul style="list-style-type: none"> <li>• Diseño e implementación de interfaz para visualizar productos.</li> <li>• Visualizar listado de productos existentes.</li> <li>• Visualizar un producto en base a su búsqueda.</li> <li>• Pruebas respectivas.</li> </ul>	
		Módulo reportes de sucursales	HU005	Consumir varios <i>endpoints</i> visualizar reportes de sucursales.	<ul style="list-style-type: none"> <li>• Diseño e implementación de formulario y botones para visualizar reportes de sucursales.</li> <li>• Validar campos requeridos</li> <li>• Generar reportes de sucursales a través de la búsqueda en base al nombre.</li> <li>• Visualizar listado de reportes por fechas.</li> <li>• Pruebas respectivas.</li> </ul>	

		Módulo stock de productos	HU006	Consumir varios <i>endpoints</i> para visualizar la disponibilidad de stock de productos.	<ul style="list-style-type: none"> <li>• Diseño e implementación de formulario y botones para visualizar la disponibilidad de stock de productos.</li> <li>• Validar campos requeridos.</li> <li>• Visualizar listado de productos.</li> <li>• Generar mensajes de alerta para productos agotados.</li> <li>• Implementación de botón auxiliar para gestionar pedidos.</li> <li>• Pruebas respectivas.</li> </ul>	
		Módulo venta por sucursales	HU007	Consumir varios <i>endpoints</i> para visualizar la venta por sucursales.	<ul style="list-style-type: none"> <li>• Diseño e implementación de formulario y botones para visualizar la venta por sucursales.</li> <li>• Visualizar listado de ventas detallado de cada producto.</li> <li>• Visualizar valores recaudados por cada factura o comprobante de venta.</li> <li>• Pruebas respectivas.</li> </ul>	



		Módulo Pedidos	HU008	Consumir varios <i>endpoints</i> para gestionar pedidos con proveedores.	<ul style="list-style-type: none"> <li>• Diseño e implementación de formulario y botones para crear, visualizar, modificar y eliminar pedidos con proveedores.</li> <li>• Validar campos requeridos.</li> <li>• Gestionar datos por parte del <i>backend</i>.</li> <li>• Pruebas respectivas.</li> </ul>	
		Módulo Personal	HU009	Consumir varios <i>endpoints</i> para visualizar el personal.	<ul style="list-style-type: none"> <li>• Diseño e implementación de formulario para visualizar el personal.</li> <li>• Visualizar el listado de empleados.</li> <li>• Implementar sección de búsqueda por categoría.</li> <li>• Implementar formularios de búsqueda.</li> <li>• Validar los campos requeridos.</li> <li>• Listar empleados en base a la búsqueda.</li> <li>• Pruebas respectivas.</li> </ul>	

SB002	Diseño e implementación de interfaz del usuario cliente y módulo a su cargo	Módulo cotizaciones	HU010	Consumir varios <i>endpoints</i> para realizar cotizaciones de productos.	<ul style="list-style-type: none"> <li>• Diseño e implementación de formulario para realizar cotizaciones de productos.</li> <li>• Validar campos requeridos.</li> <li>• Generar cotizaciones para clientes.</li> <li>• Pruebas respectivas.</li> </ul>	
SB003	Pruebas en el sistema <i>web</i>	<ul style="list-style-type: none"> <li>• Pruebas unitarias.</li> <li>• Pruebas de rendimiento.</li> <li>• Prueba de compatibilidad.</li> <li>• Prueba de aceptación.</li> </ul>				20 H
SB004	Despliegue del sistema <i>web</i>	<ul style="list-style-type: none"> <li>• Despliegue del sistema <i>web</i> en <i>Heroku</i>.</li> </ul>				10 H
Documentación		<ul style="list-style-type: none"> <li>• Trabajo de Integración Curricular.</li> <li>• Anexos.</li> </ul>				50 H
<b>TOTAL</b>						<b>240 H</b>

## Diseño de interfaces

A continuación, se presentan los prototipos de cada uno de los módulos del sistema *web*, en los cuales se observa el diseño de las interfaces para cada rol de usuario: cliente y administrador. Desde la **Fig. 22** hasta la **Fig. 50** presentan las interfaces del sistema *web*.

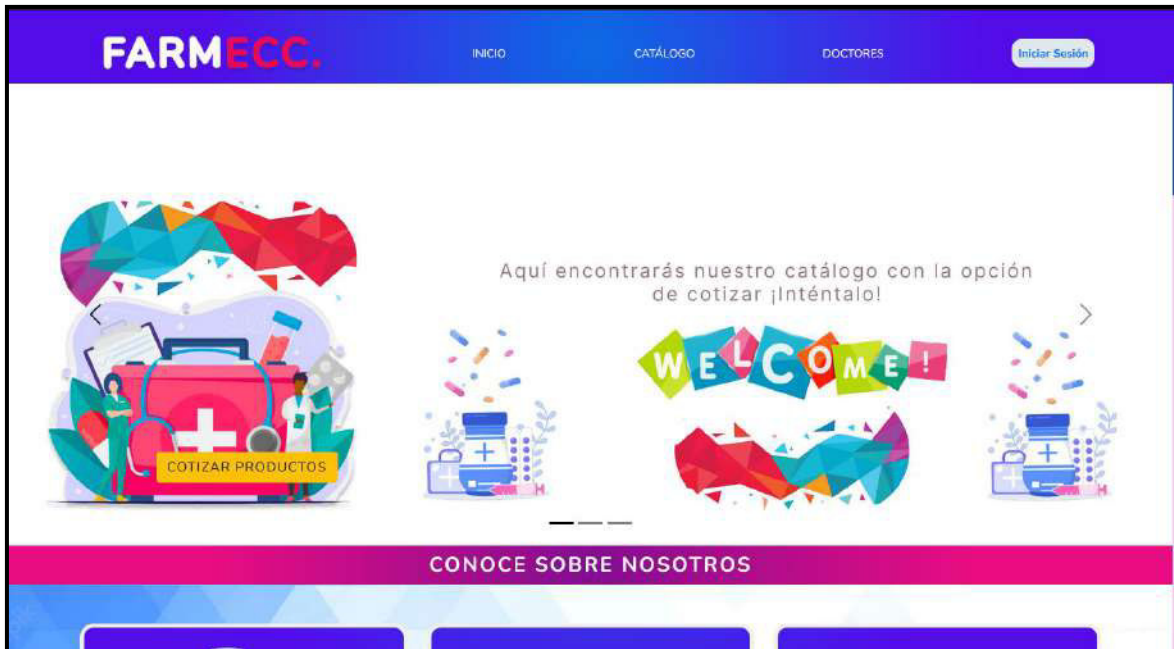


Fig. 22: Diseño de Interfaz – Página Informativa.

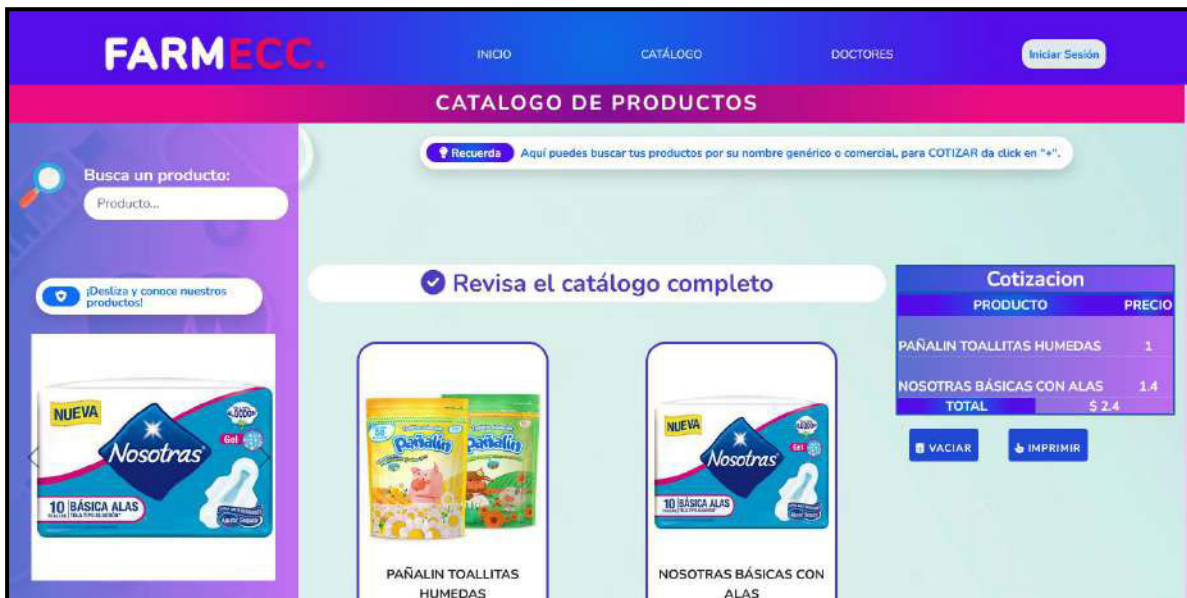


Fig. 23: Diseño de Interfaz – Módulo de catálogo y cotizaciones (Usuario cliente).



Fig. 24: Diseño de Interfaz – Módulo de doctores (Usuario cliente).

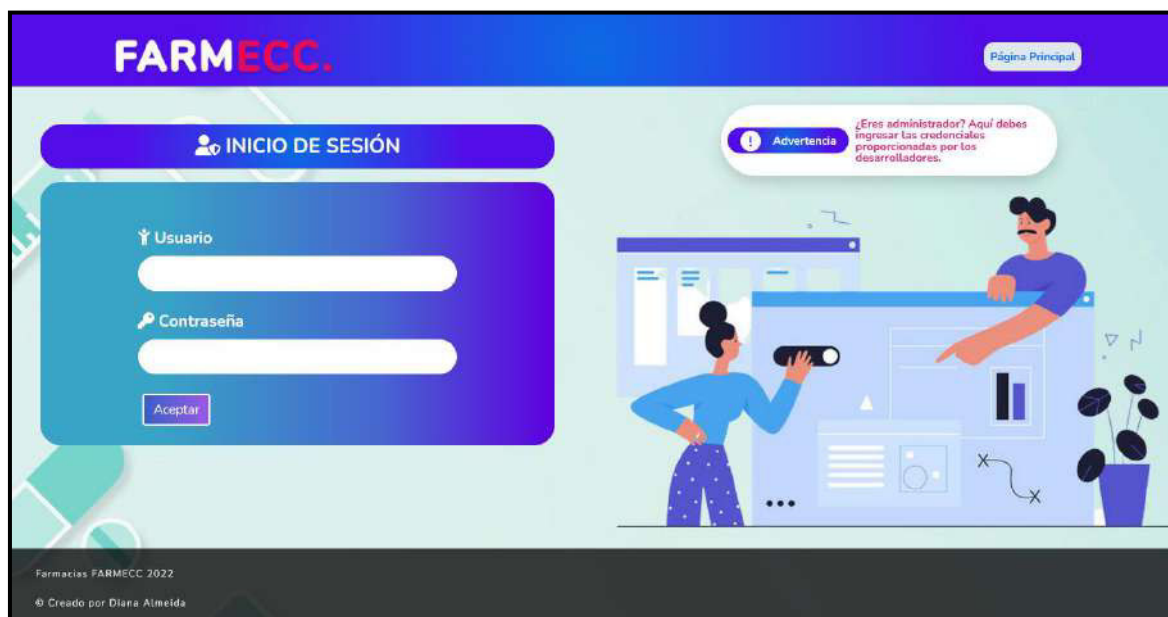


Fig. 25: Diseño de Interfaz – Inicio de sesión.

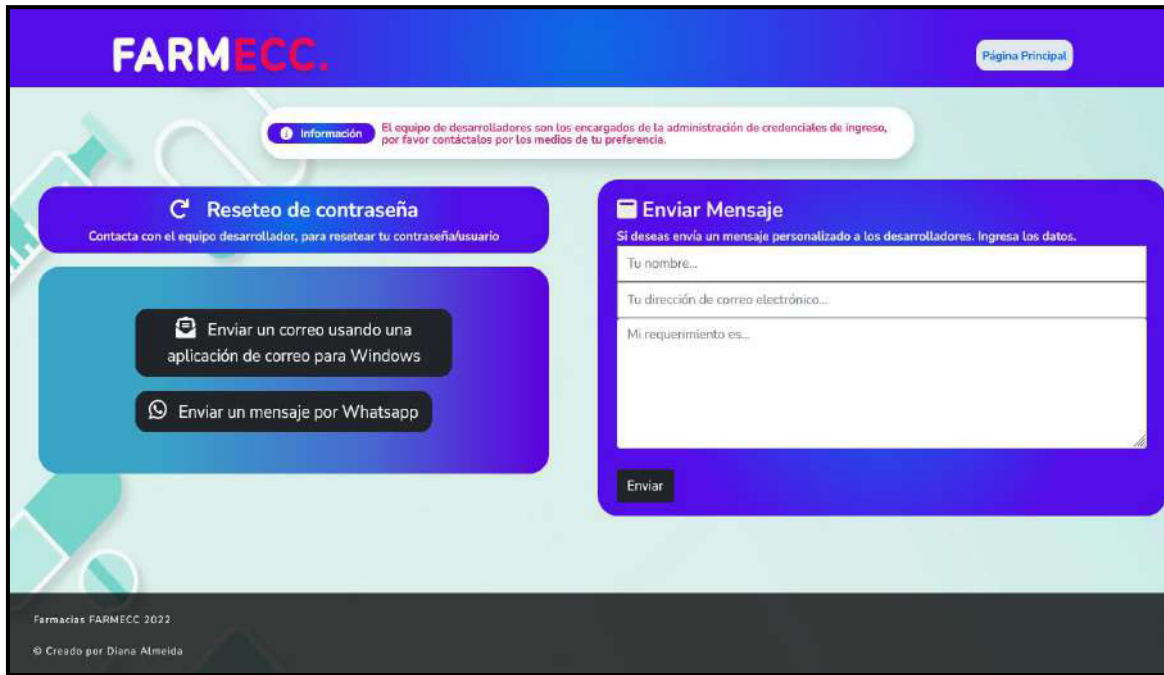


Fig. 26: Diseño de Interfaz – Reseteo de contraseña.



Fig. 27: Diseño de Interfaz – Página principal de módulos.



Fig. 28: Diseño de Interfaz – Módulo gestión de sucursales.

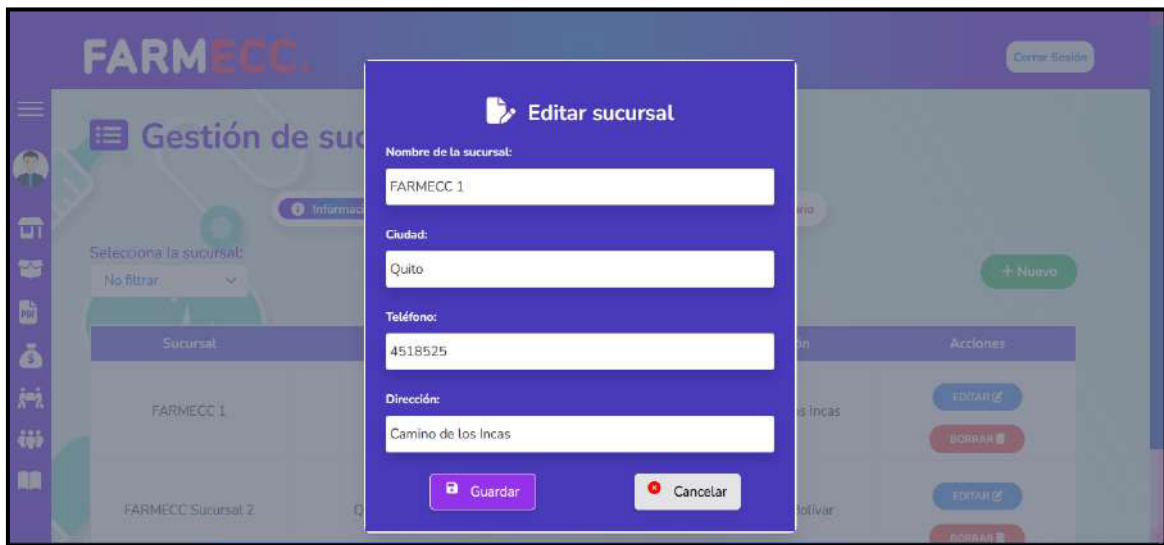


Fig. 29: Diseño de Interfaz – Modal para editar sucursal.

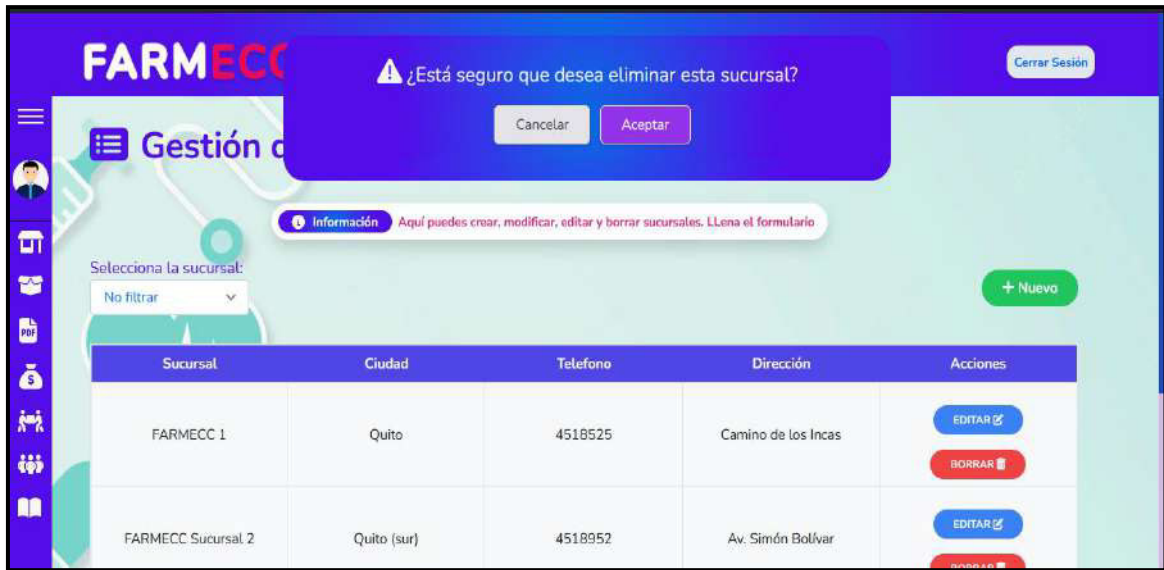


Fig. 30: Diseño de Interfaz – Modal de confirmación para borrar sucursal.

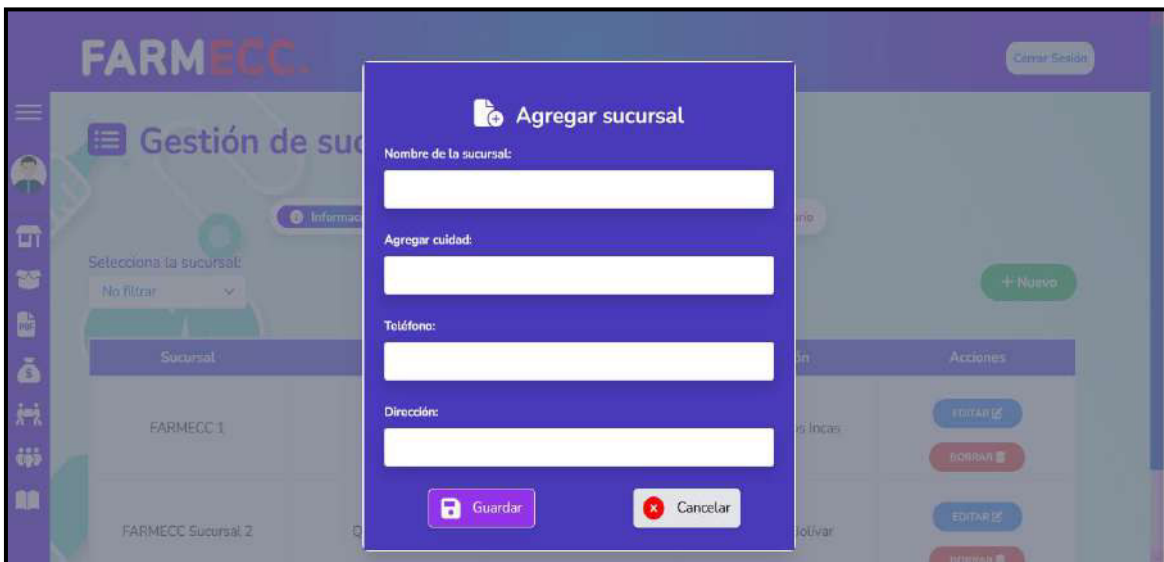


Fig. 31: Diseño de Interfaz – Modal de añadir una sucursal.





Fig. 32: Diseño de Interfaz – Resultados de búsqueda de sucursal.



Fig. 33: Diseño de Interfaz – Módulo stock de productos.



Selecciona la sucursal: FARMECC 1

Busca el producto: Producto...

### Resultados de la búsqueda: FARMECC 1

Nombre Genérico	Nombre Comercial	Descripción	Registro sanitario	Precio de venta	Precio de compra	Cantidad	Sucursal	Stock
TOALLITAS HUMEDAS	PAÑALIN TOALLITAS HUMEDAS	x50	341040	1	0.68	6	FARMECC 1	POR AGOTARSE
TOALLAS INTIMAS	NOSOTRAS BÁSICAS CON ALAS	x10		1.4	1.15	9	FARMECC 1	POR AGOTARSE
TOALLAS INTIMAS X10	NOSOTRAS INVISIBLES RAPIGEL	x10		1.65	1.45	5	FARMECC 1	POR AGOTARSE
PAPEL HIGIÉNICO	FLOR CLÁSICO	x6 Rollos		1	0.89	6	FARMECC 1	POR AGOTARSE

Fig. 34: Diseño de Interfaz – Filtrado de productos por sucursal.

Stock de productos

Información Aquí puedes visualizar los productos existentes en las sucursales de FARMECC.

Selecciona la sucursal: No filtrar

Busca el producto: toallitas

### Resultados de la búsqueda: toallitas

Nombre Genérico	Nombre Comercial	Descripción	Registro sanitario	Precio de venta	Precio de compra	Cantidad	Sucursal	Stock
TOALLITAS HUMEDAS	PAÑALIN TOALLITAS HUMEDAS	x50	341040	1	0.68	6	FARMECC 1	POR AGOTARSE
TOALLITAS HUMEDAS X100	TITOS	TOALLITAS HUMEDAS X100	2022-03-26	2.25	1.4	2	FARMECC Sucursal 2	POR AGOTARSE

Fig. 35: Diseño de Interfaz – Resultados de búsqueda de productos.



Fig. 36: Diseño de Interfaz – Módulo generar reportes de ventas.



Fig. 37: Diseño de Interfaz – Módulo ventas.



Fig. 38: Diseño de Interfaz – Filtrado de ventas por sucursal.



Fig. 39: Diseño de Interfaz – Módulo pedidos a proveedores.



Fig. 40: Diseño de Interfaz – Modal para editar un pedido.



Fig. 41: Diseño de Interfaz – Modal de confirmación para borrar un pedido.



**Fig. 42: Diseño de Interfaz – Modal de añadir un pedido.**



**Fig. 43: Diseño de Interfaz – Listado de proveedores.**



**Fig. 44: Diseño de Interfaz – Modal para editar un proveedor.**



**Fig. 45: Diseño de Interfaz – Modal de confirmación para borrar un proveedor.**





Fig. 46: Diseño de Interfaz – Modal de añadir un proveedor.



Fig. 47: Diseño de Interfaz – Filtrado de proveedores.

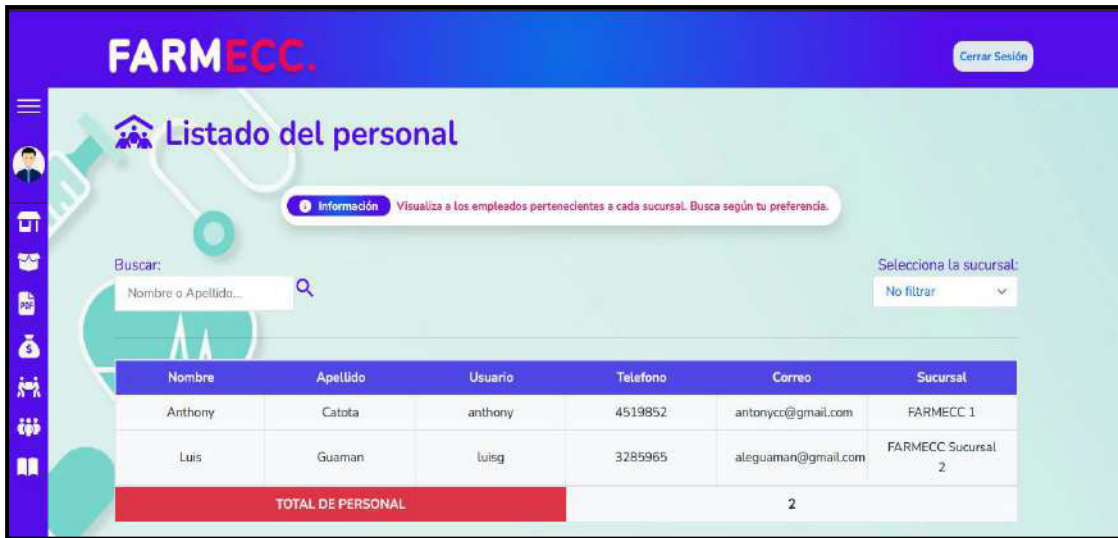


Fig. 48: Diseño de Interfaz – Módulo personal.



Fig. 49: Diseño de Interfaz – Búsqueda de empleado.

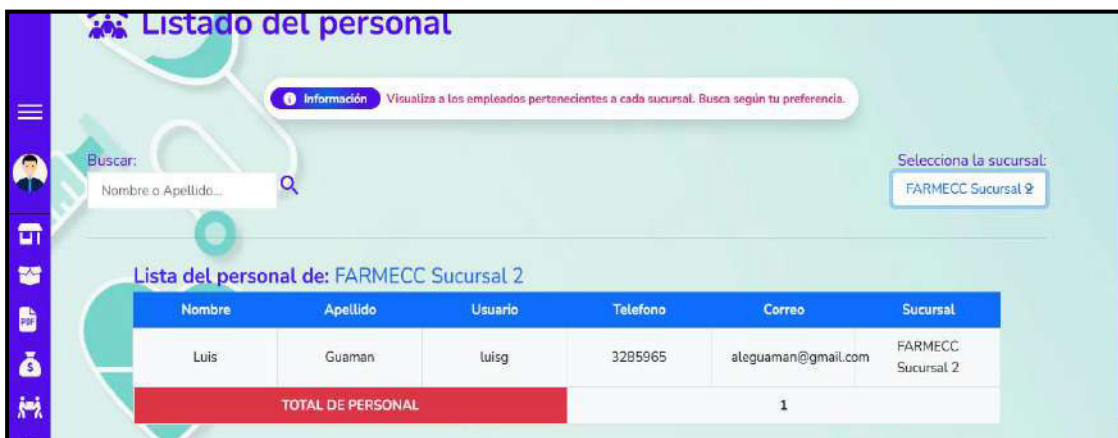


Fig. 50: Diseño de Interfaz – Filtrado de empleados por sucursal.



## Pruebas

Posterior a la finalización de la codificación del sistema *web* se ha desarrollado la ejecución de pruebas unitarias, compatibilidad y de aceptación con la finalidad de comprobar la calidad del código y los módulos.

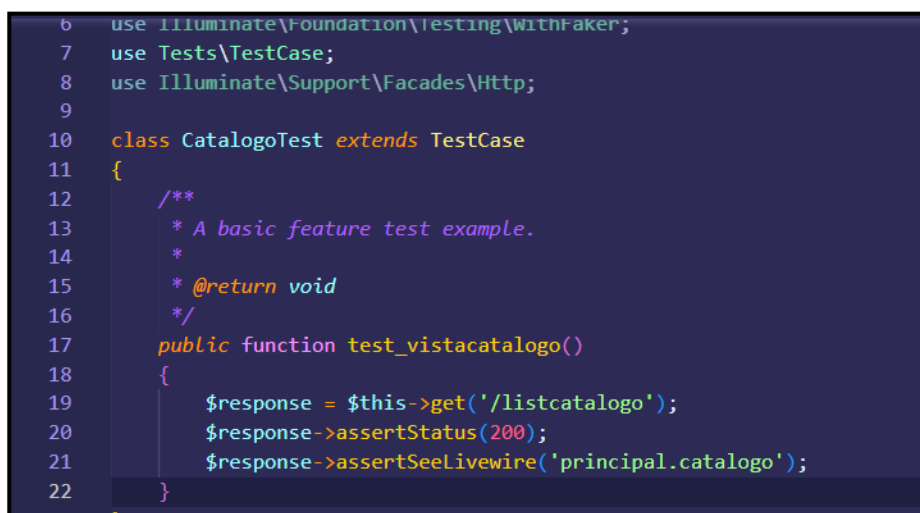
### Pruebas unitarias

La **Fig. 51** muestra la configuración del proyecto (archivo `phpunit.xml`) previa la ejecución de las pruebas unitarias de las principales funcionalidades del sistema *web*. Posteriormente, mediante la construcción de un *test* se crea una clase con funciones para cada uno de los módulos del sistema desarrollado, A continuación, desde la **Fig. 52** hasta la **Fig. 69**, se presenta el proceso antes mencionado, junto con los resultados de cada una de las pruebas unitarias.

A screenshot of a code editor showing the configuration of the PHPUnit XML file. The file is named 'phpunit.xml' and contains XML tags for environment variables. The code is as follows:

```
19 </coverage>
20 <php>
21     <env name="APP_ENV" value="testing"/>
22     <env name="BCRYPT_ROUNDS" value="4"/>
23     <env name="CACHE_DRIVER" value="array"/>
24     <!-- <env name="DB_CONNECTION" value="sqlite"/> -->
25     <!-- <env name="DB_DATABASE" value=":memory:" /> -->
26     <env name="MAIL_MAILER" value="array"/>
27     <env name="QUEUE_CONNECTION" value="sync"/>
28     <env name="SESSION_DRIVER" value="array"/>
29     <env name="TELESCOPE_ENABLED" value="false"/>
30 </php>
31 </phpunit>
```

**Fig. 51:** Configuración del entorno de pruebas para Laravel.

A screenshot of a code editor showing a unit test class named 'CatalogoTest'. The class extends 'TestCase' and contains a single test method 'test\_vistacatalogo()'. The code is as follows:

```
6 use Illuminate\Foundation\Testing\WithFaker;
7 use Tests\TestCase;
8 use Illuminate\Support\Facades\Http;
9
10 class CatalogoTest extends TestCase
11 {
12     /**
13      * A basic feature test example.
14      *
15      * @return void
16      */
17     public function test_vistacatalogo()
18     {
19         $response = $this->get('/listcatalogo');
20         $response->assertStatus(200);
21         $response->assertSeeLivewire('principal.catalogo');
22     }
23 }
```

**Fig. 52:** Prueba unitaria #1 del módulo catálogo de productos.

```

PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb> vendor/bin/phpunit --filter test_vistacatalogo
PHPUnit 9.5.20 #StandWithUkraine

.

1 / 1 (100%)

Time: 00:05.071, Memory: 30.00 MB

OK (1 test, 2 assertions)
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb>

```

Fig. 53: Resultado de la prueba unitaria #1.

```

7 use Tests\TestCase;
8
9 class DoctoresTest extends TestCase
10 {
11     /**
12      * A basic feature test example.
13      *
14      * @return void
15      */
16     public function test_doctores()
17     {
18         $response = $this->get('/contactos');
19         $response->assertStatus(200);
20         $response->assertSeeLivewire('principal.doctores');
21     }
22 }

```

Fig. 54: Prueba unitaria #2 del módulo doctores.

```

PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb> vendor/bin/phpunit --filter test_doctores
PHPUnit 9.5.20 #StandWithUkraine

.

1 / 1 (100%)

Time: 00:04.045, Memory: 24.00 MB

OK (1 test, 2 assertions)
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb>

```

Fig. 55: Resultado de la prueba unitaria #2.

```

7 use Tests\TestCase;
8 use Illuminate\Support\Facades\Http;
9 use Livewire\Livewire;
10
11 class LoginTest extends TestCase
12 {
13     /**
14      * A basic feature test example.
15      *
16      * @return void
17      */
18     public function test_login()
19     {
20         $response = $this->get('/login');
21         $response->assertStatus(200);
22         $response->assertSeeLivewire('login');
23     }
24 }

```

Fig. 56: Prueba unitaria #3 del inicio de sesión.

```
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb> vendor/bin/phpunit --filter test_login
PHPUnit 9.5.20 #StandWithUkraine

.

1 / 1 (100%)

Time: 00:00.731, Memory: 24.00 MB

OK (1 test, 2 assertions)
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb>
```

Fig. 57: Resultado de la prueba unitaria #3.

```
6 use Illuminate\Foundation\Testing\WithFaker;
7 use Tests\TestCase;
8
9 class GestionSucurTest extends TestCase
10 {
11     /**
12      * A basic feature test example.
13      *
14      * @return void
15      */
16     public function test_gestsucursal()
17     {
18         $response = $this->get('/gestsucursales');
19         $response->assertStatus(200);
20         $response->assertSeeLivewire('gestsucursal');
21     }
22 }
```

Fig. 58: Prueba unitaria #4 del módulo gestión de sucursales.

```
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb> vendor/bin/phpunit --filter test_gestsucursal
PHPUnit 9.5.20 #StandWithUkraine

.

1 / 1 (100%)

Time: 00:02.495, Memory: 24.00 MB

OK (1 test, 2 assertions)
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb>
```

Fig. 59: Resultado de la prueba unitaria #4.

```
7 use Tests\TestCase;
8
9 class StockTest extends TestCase
10 {
11     /**
12      * A basic feature test example.
13      *
14      * @return void
15      */
16     public function test_stockdeproductos()
17     {
18         $response = $this->get('/productos');
19         $response->assertStatus(200);
20         $response->assertSeeLivewire('visualproductos');
21     }
22 }
```

Fig. 60: Prueba unitaria #5 del módulo stock de productos.

```
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb> vendor/bin/phpunit --filter test_stockdeproductos
PHPUnit 9.5.20 #StandWithUkraine

.

1 / 1 (100%)

Time: 00:04.940, Memory: 26.00 MB

OK (1 test, 2 assertions)
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb>
```

Fig. 61: Resultado de la prueba unitaria #5.

```
7 use Tests\TestCase;
8
9 class VentasTest extends TestCase
10 {
11     /**
12      * A basic feature test example.
13      *
14      * @return void
15      */
16     public function test_ventas()
17     {
18         $response = $this->get('/ventas');
19         $response->assertStatus(200);
20         $response->assertSeeLivewire('visualventas');
21     }
22 }
```

Fig. 62: Prueba unitaria #6 del módulo ventas.

```
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb> vendor/bin/phpunit --filter test_ventas
PHPUnit 9.5.20 #StandWithUkraine

.

1 / 1 (100%)

Time: 00:01.984, Memory: 24.00 MB

OK (1 test, 2 assertions)
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb>
```

Fig. 63: Resultado de la prueba unitaria #6.

```
7 use Tests\TestCase;
8
9 class ReportesTest extends TestCase
10 {
11     /**
12      * A basic feature test example.
13      *
14      * @return void
15      */
16     public function test_reporte()
17     {
18         $response = $this->get('/repsucursales');
19         $response->assertStatus(200);
20         $response->assertSeeLivewire('reporte');
21     }
22 }
```

Fig. 64: Prueba unitaria #7 del módulo reportes de ventas.

```
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb> vendor/bin/phpunit --filter test_reporte
PHPUnit 9.5.20 #StandWithUkraine

.

1 / 1 (100%)

Time: 00:01.770, Memory: 24.00 MB

OK (1 test, 2 assertions)
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb>
```

Fig. 65: Resultado de la prueba unitaria #7.

```
6 use Illuminate\Foundation\Testing\WithFaker;
7 use Tests\TestCase;
8
9 class PedidosTest extends TestCase
10 {
11     /**
12      * A basic feature test example.
13      *
14      * @return void
15      */
16     public function test_pedidosP()
17     {
18         $response = $this->get('/pedidos');
19         $response->assertStatus(200);
20         $response->assertSeeLivewire('pedidosproveedores');
21     }
22 }
```

Fig. 66: Prueba unitaria #8 del módulo pedidos a proveedores.

```
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb> vendor/bin/phpunit --filter test_pedidosP
PHPUnit 9.5.20 #StandWithUkraine

.

1 / 1 (100%)

Time: 00:05.838, Memory: 30.00 MB

OK (1 test, 2 assertions)
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb>
```

Fig. 67: Resultado de la prueba unitaria #8.

```
6 use Illuminate\Foundation\Testing\WithFaker;
7 use Tests\TestCase;
8
9 class PersonalTest extends TestCase
10 {
11     /**
12      * A basic feature test example.
13      *
14      * @return void
15      */
16     public function test_personal()
17     {
18         $response = $this->get('/personal');
19         $response->assertStatus(200);
20         $response->assertSeeLivewire('personal');
21     }
22 }
```

Fig. 68: Prueba unitaria #10 del módulo personal.

```
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb> vendor/bin/phpunit --filter test_personal
PHPUnit 9.5.20 #StandWithUkraine
.
1 / 1 (100%)

Time: 00:05.426, Memory: 30.00 MB

OK (1 test, 2 assertions)
PS C:\Users\Diana\Desktop\FARMECC\FarmeccWeb>
```

Fig. 69: Resultado de la prueba unitaria #10.

## Pruebas de compatibilidad

Para esta sección se verifica la correcta funcionalidad del sistema *web* en distintos navegadores con el fin de detectar posibles fallas y si existen cambios en la presentación del sistema. A continuación, desde la **Fig. 70** hasta la **Fig. 87** se presentan los resultados obtenidos en los navegadores *web*.

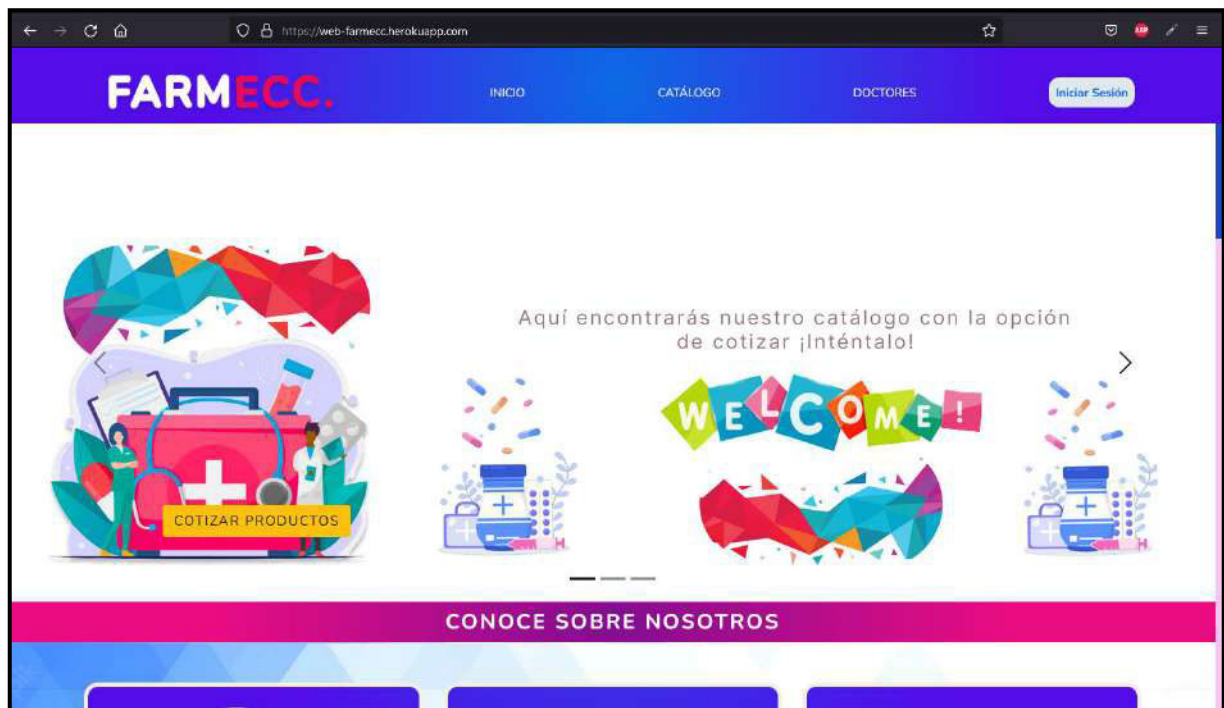


Fig. 70: Página principal con información general vista desde el navegador *Mozilla Firefox*.

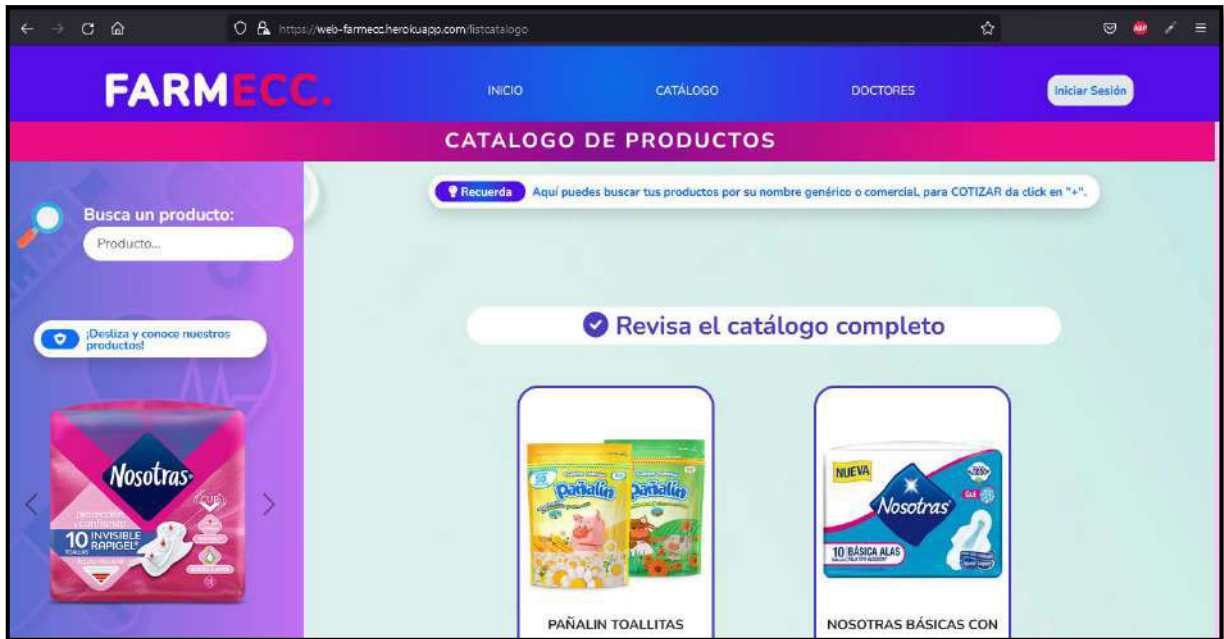


Fig. 71: Módulo catálogo de productos visto desde el navegador *Mozilla Firefox*.



Fig. 72: Módulo doctores visto desde el navegador *Mozilla Firefox*.



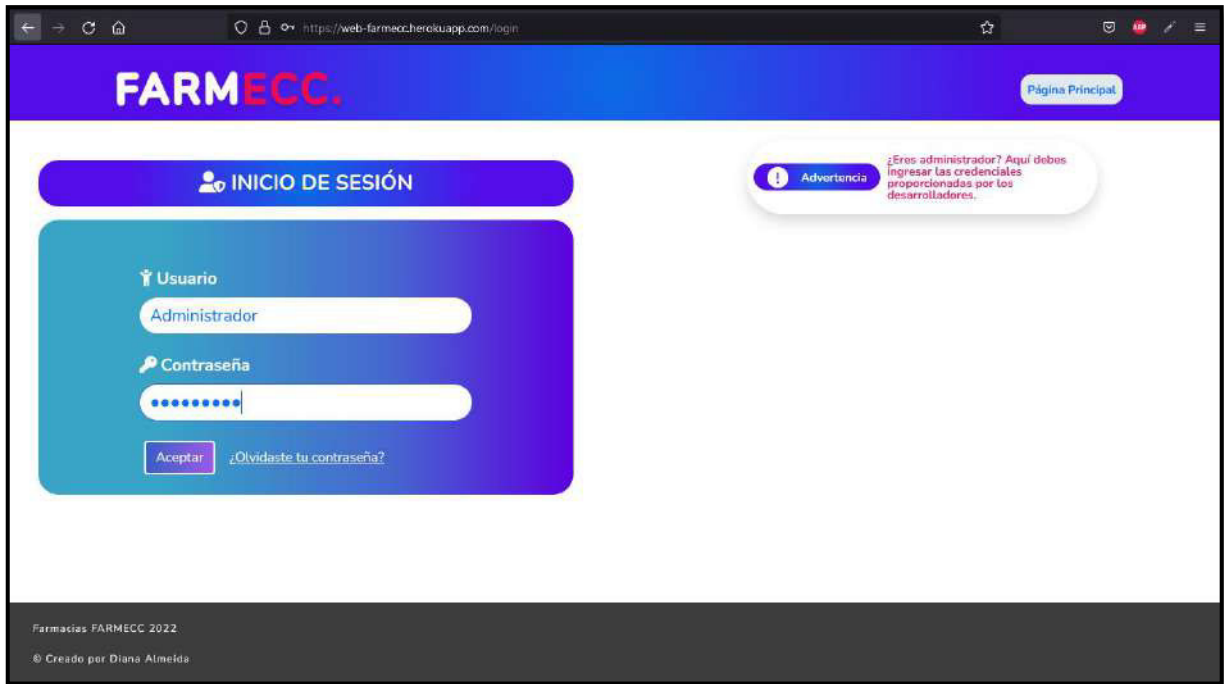


Fig. 73: Página de inicio de sesión vista desde el navegador *Mozilla Firefox*.

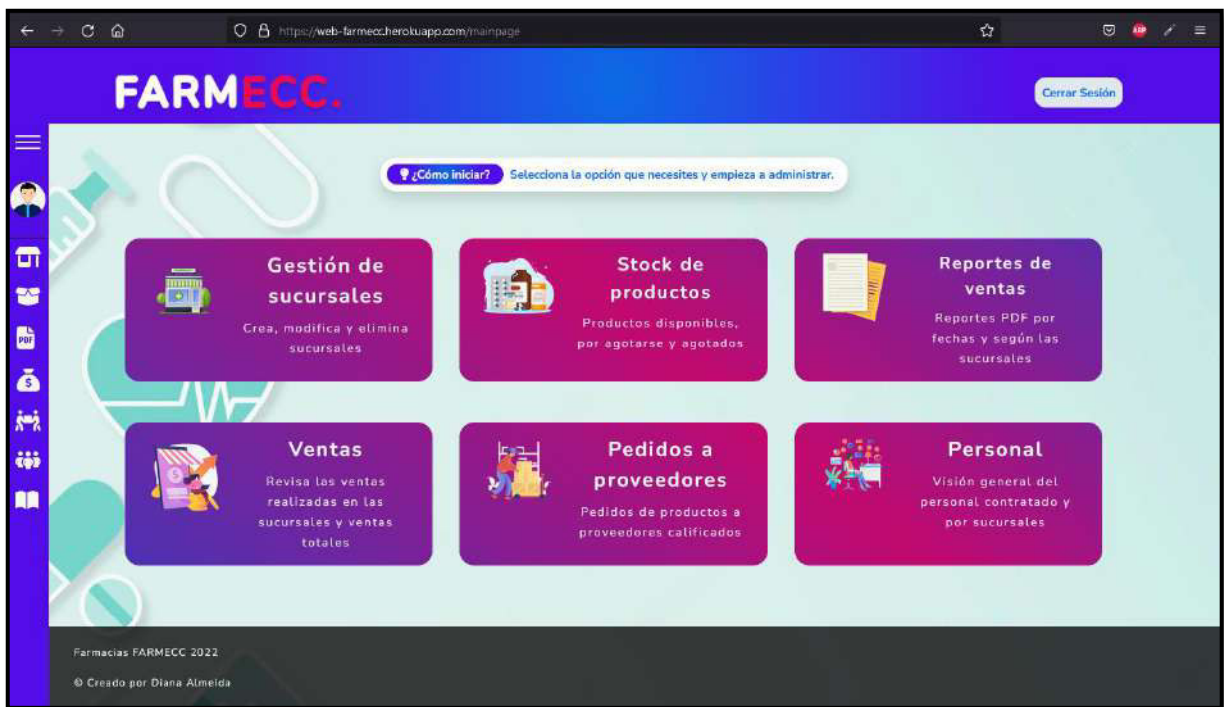


Fig. 74: Página inicial de módulos vista desde el navegador *Mozilla Firefox*.



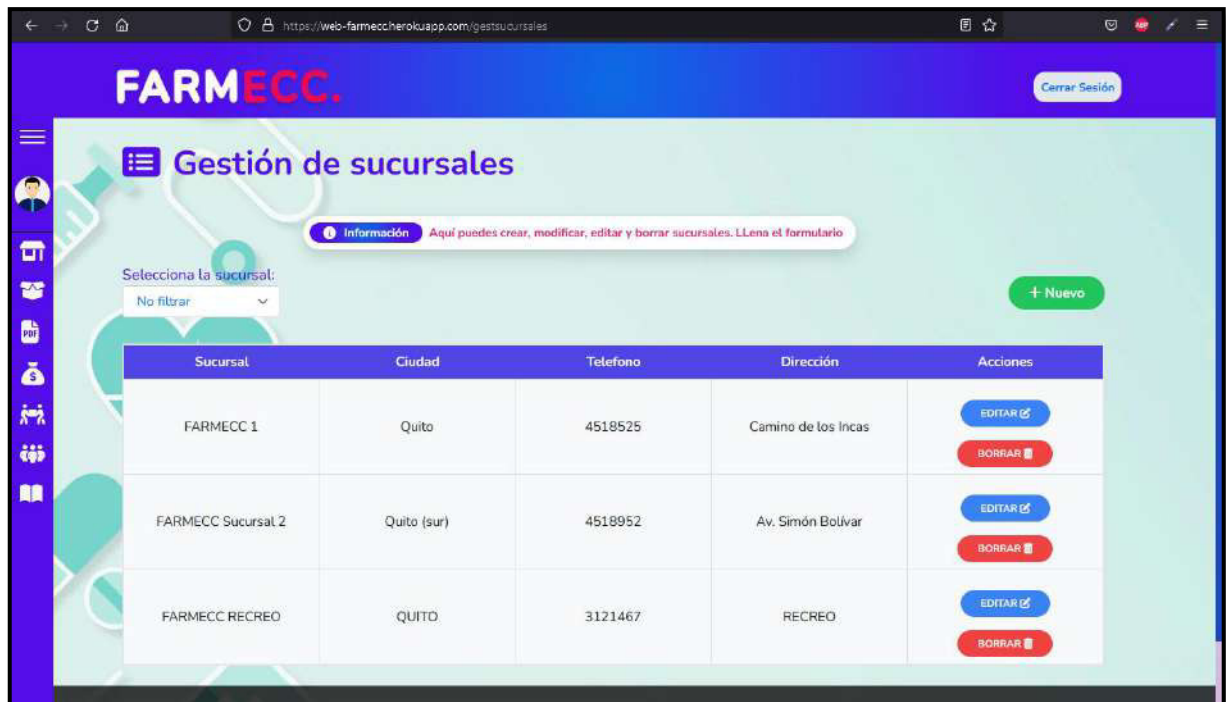


Fig. 75: Módulo gestión de sucursales visto desde el navegador *Mozilla Firefox*.

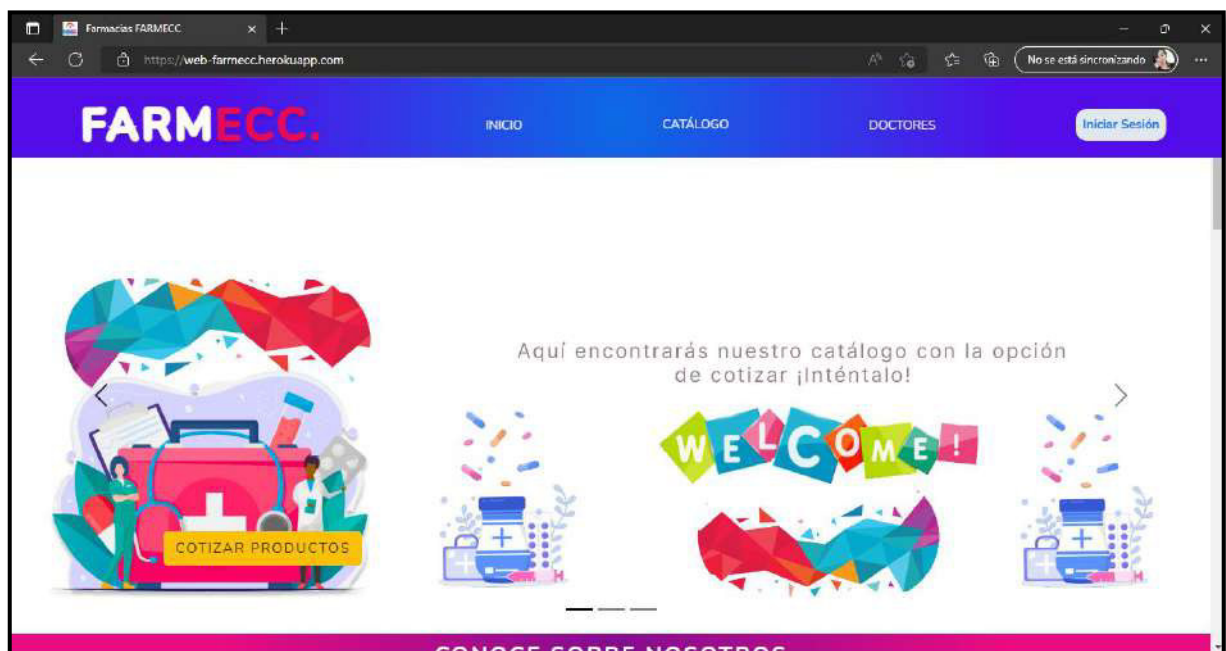


Fig. 76: Página principal vista desde el navegador *Microsoft Edge*.

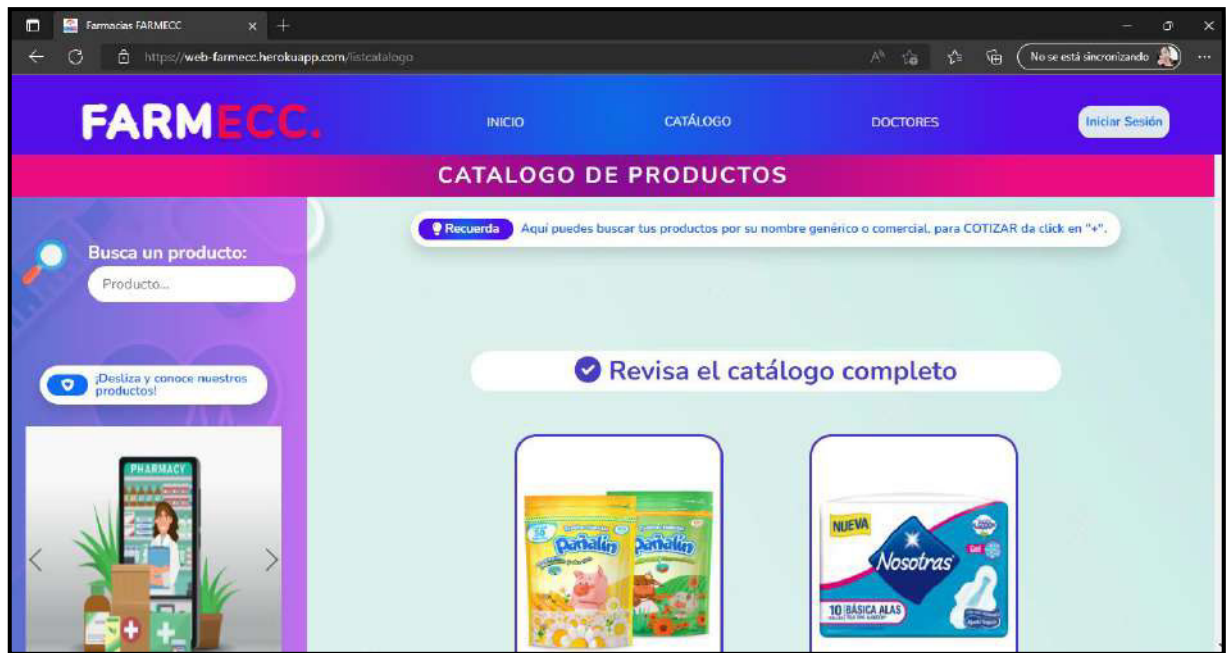


Fig. 77: Módulo de catálogo de productos visto desde el navegador *Microsoft Edge*.



Fig. 78: Módulo doctores visto desde el navegador *Microsoft Edge*.

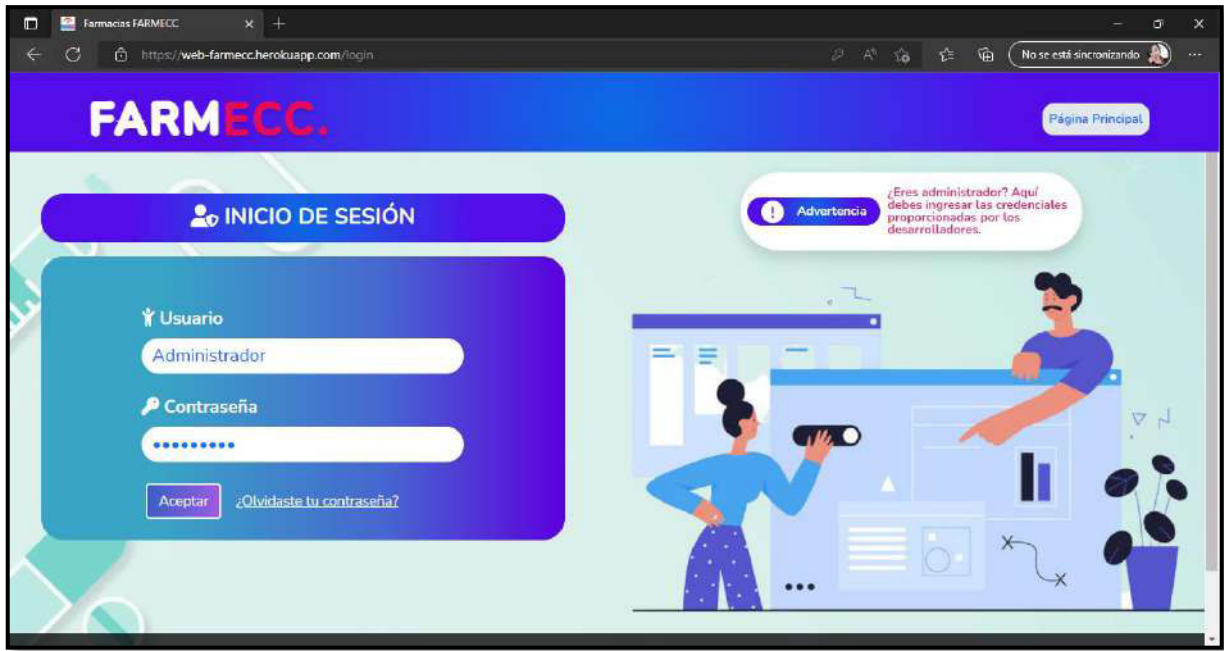


Fig. 79: Página de inicio de sesión vista desde el navegador *Microsoft Edge*.

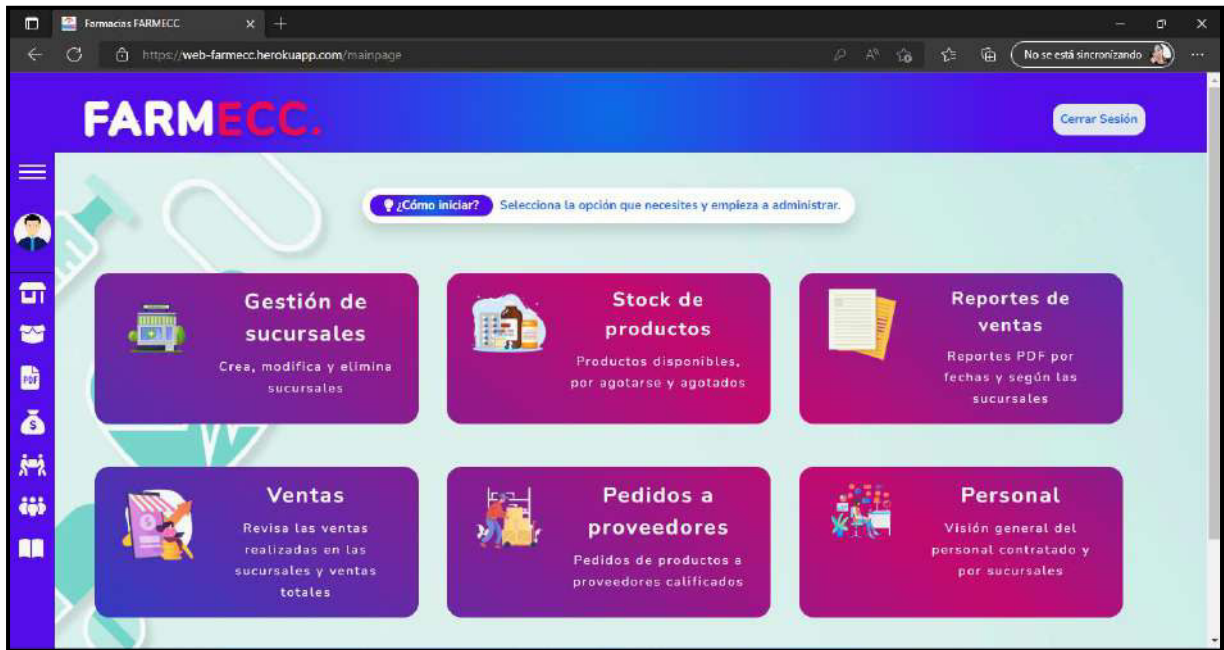


Fig. 80: Página inicial de módulos vista desde el navegador *Microsoft Edge*.

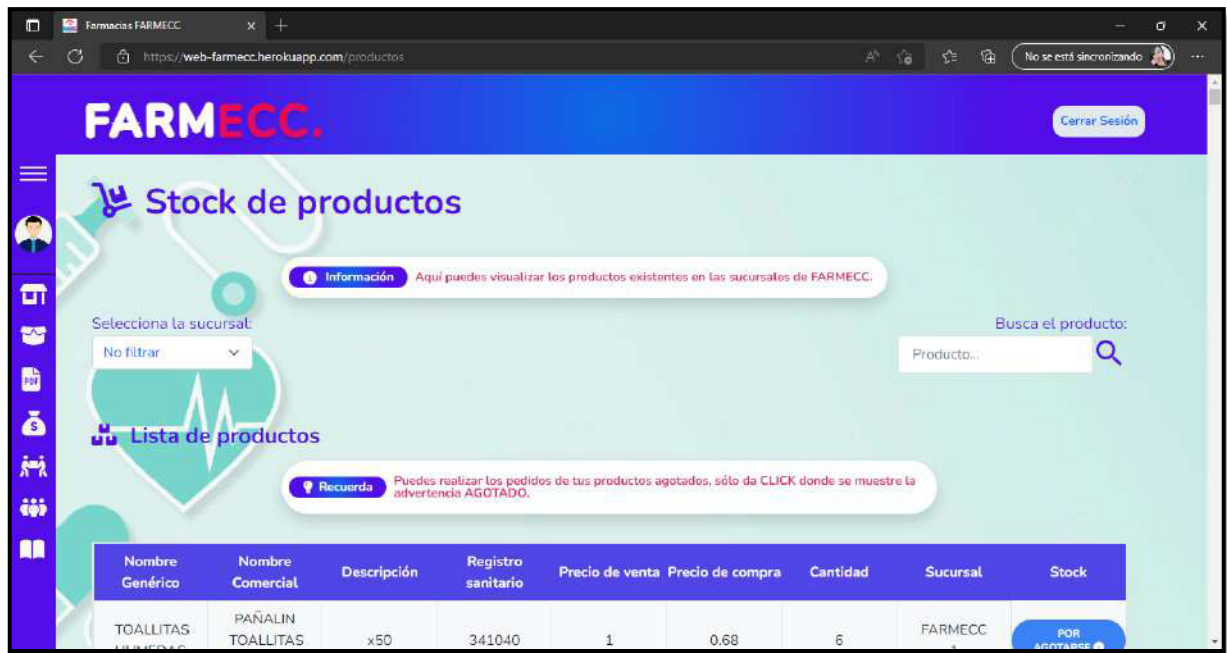


Fig. 81: Módulo stock de productos visto desde el navegador *Microsoft Edge*.

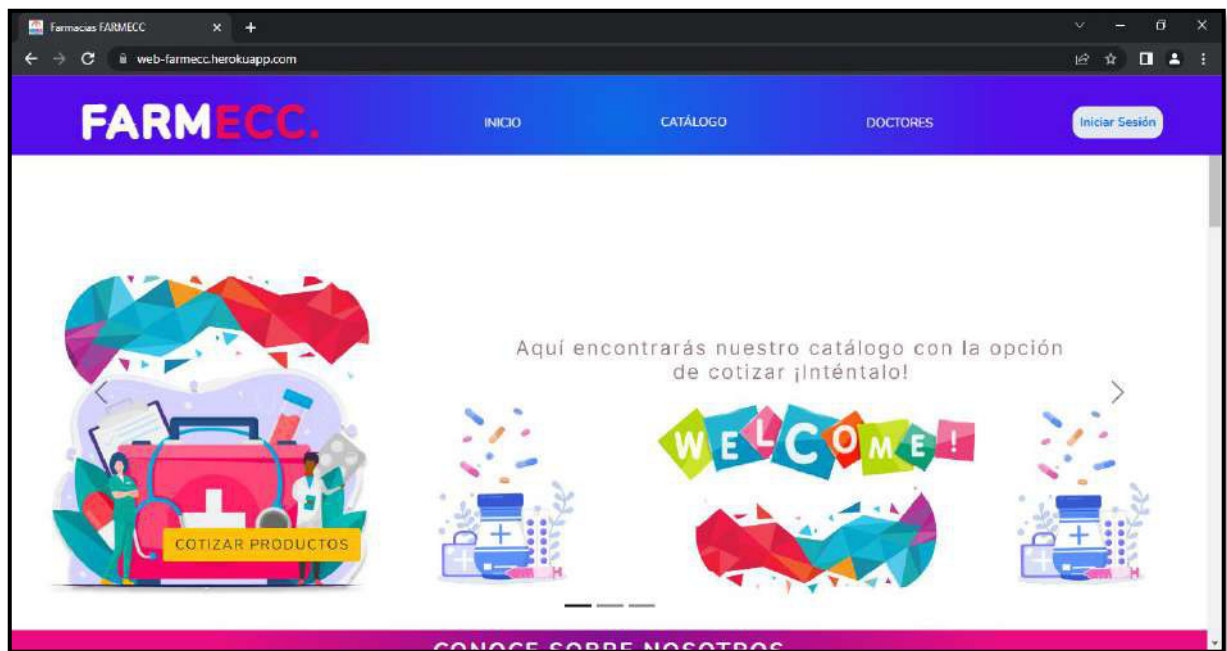


Fig. 82: Página principal vista desde el navegador *Google Chrome*.



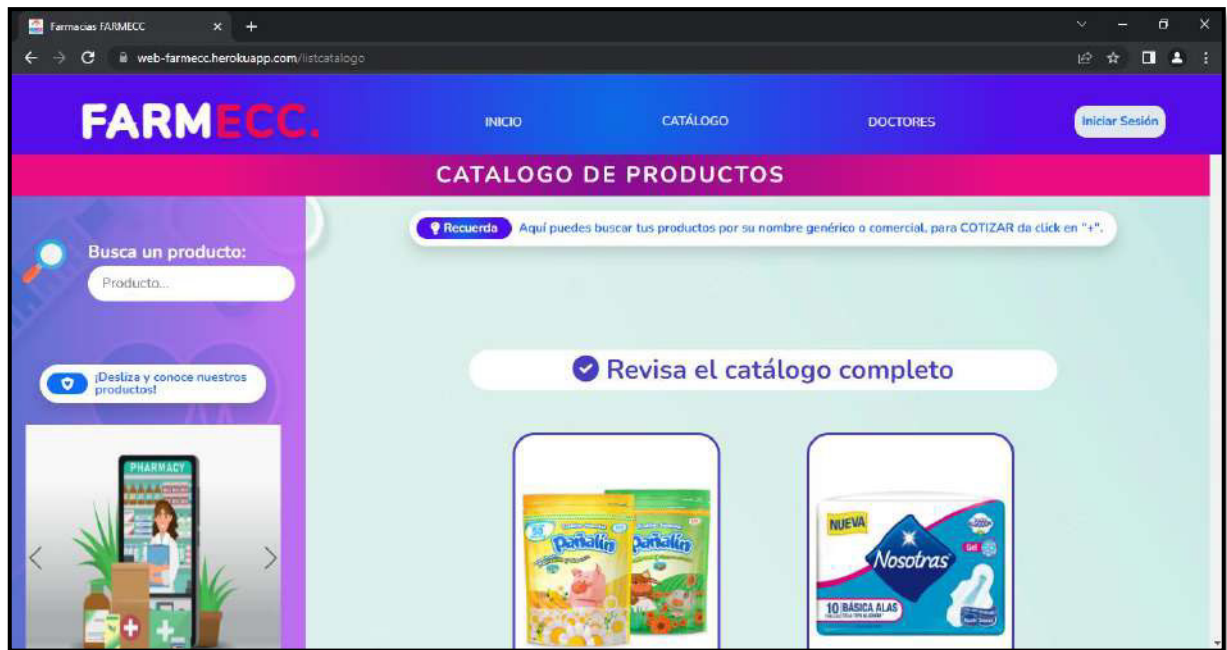


Fig. 83: Módulo catálogo de productos visto desde el navegador *Google Chrome*.

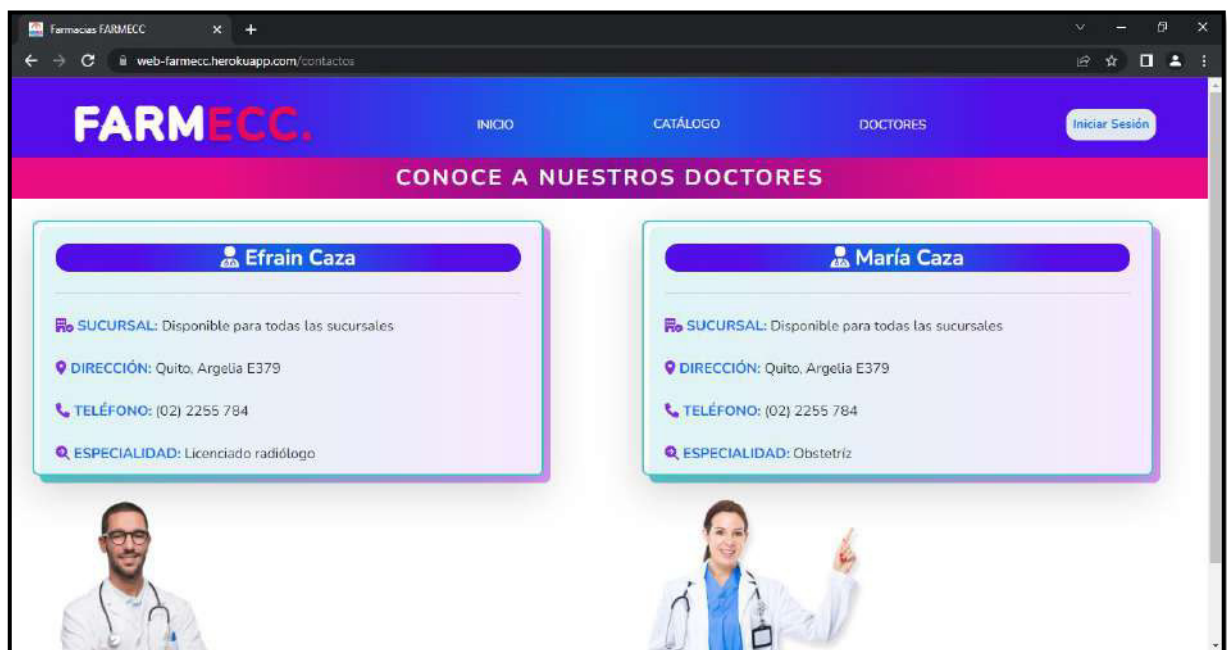


Fig. 84: Módulo doctores visto desde el navegador *Google Chrome*.

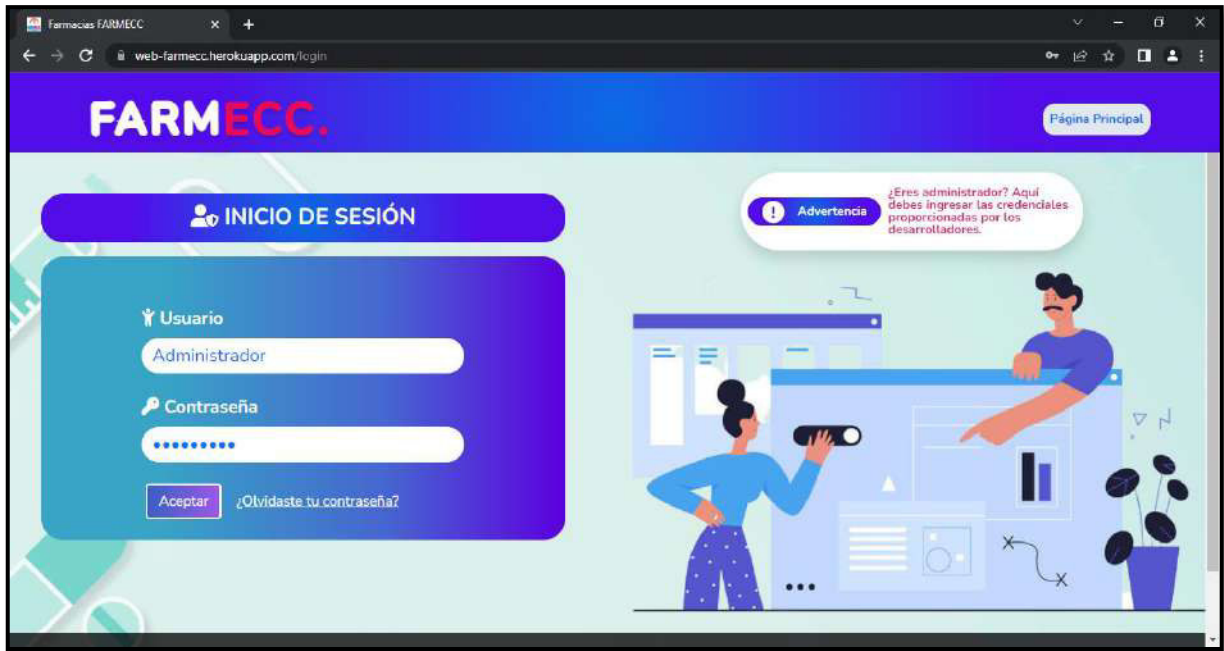


Fig. 85: Página de inicio de sesión visto desde el navegador *Google Chrome*.

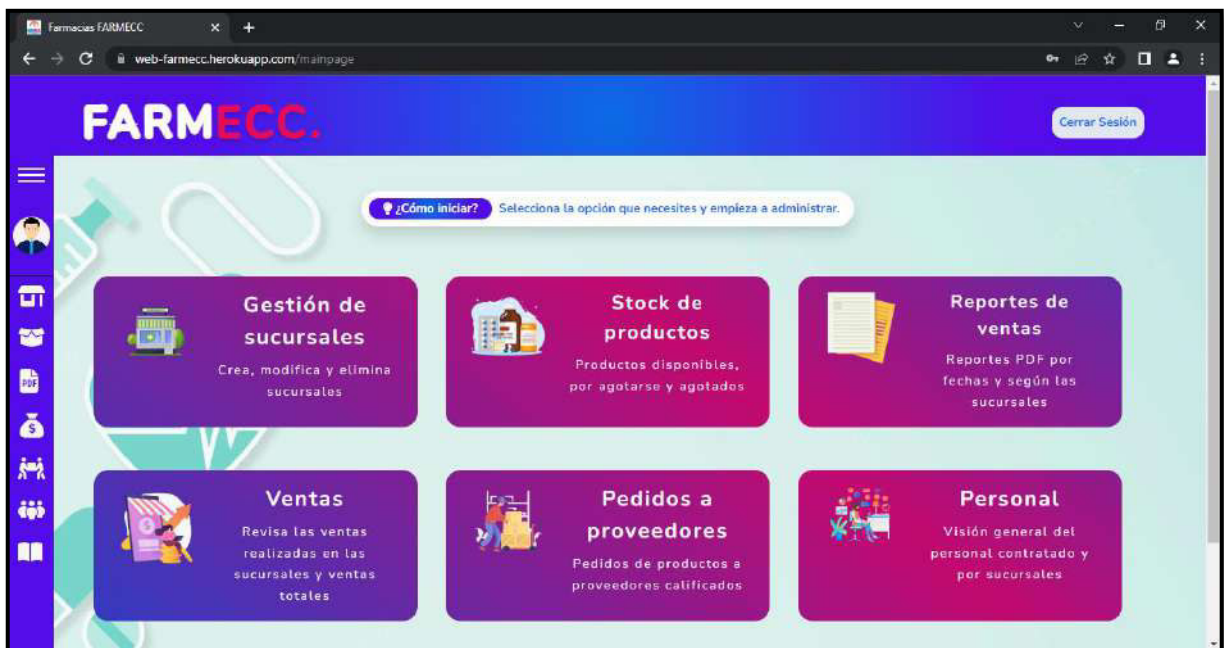
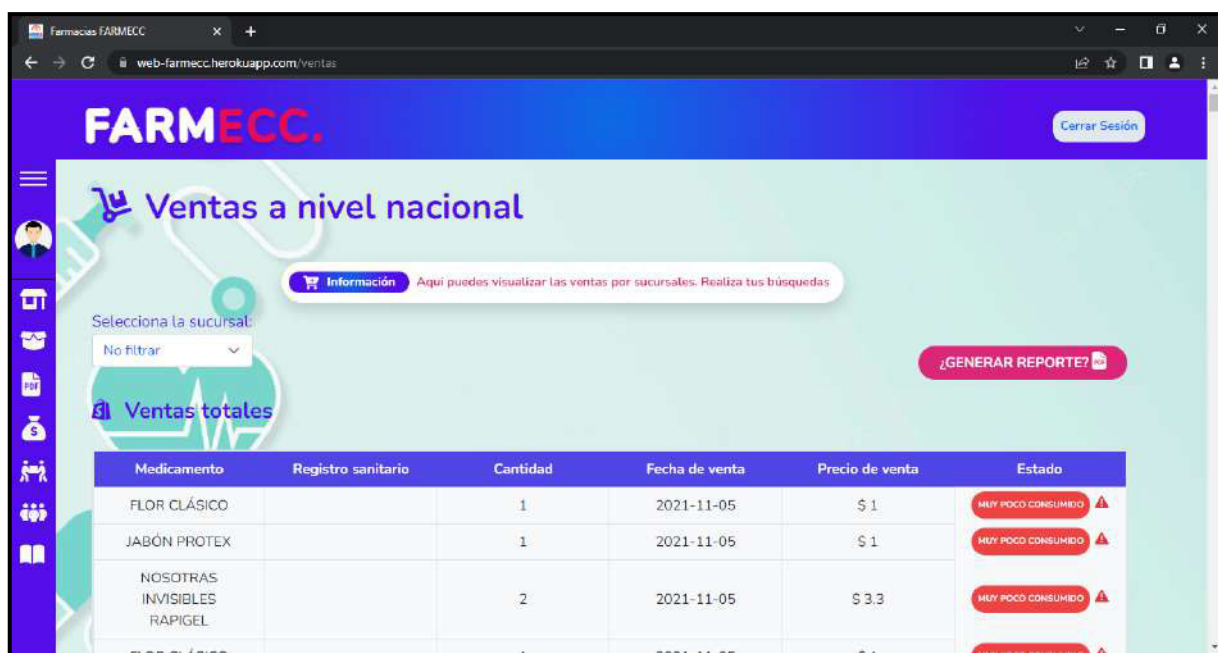


Fig. 86: Página inicial de módulos vista desde el navegador *Google Chrome*.



**Fig. 87: Módulo de ventas visto desde el navegador *Google Chrome*.**

## Pruebas de aceptación

A continuación, desde la **TABLA XXIII** hasta la **TABLA XXXI**, se presentan las 9 Pruebas de aceptación restantes. Cabe mencionar que cada prueba expone los procedimientos especificados para cada rol de usuario y su adecuada interacción, ejecución, verificación y aprobación de las mismas.

### **TABLA XXIII: Prueba de Aceptación N°2 – Consumir varios *endpoints* para iniciar sesión y cerrar sesión.**

PRUEBA DE ACEPTACIÓN	
<b>Identificador (ID):</b> PA002	<b>Identificador historia de Usuario:</b> HU002
<b>Nombre:</b> Consumir varios <i>endpoints</i> para iniciar sesión y cerrar sesión.	
<b>Descripción:</b> El usuario administrador necesita consumir varios <i>endpoints</i> para iniciar sesión y cerrar sesión. Al ingresar a la opción de “administrador” de la página inicial se despliega un formulario de inicio de sesión para ser llenado, para el cierre de sesión el administrador deberá seleccionar la opción “cerrar sesión” y para un posible cambio de contraseña se requiere que el solicitante se contacte con el equipo desarrollador por medio de las opciones presentadas en la página respectiva.	

**Pasos de ejecución:**

- Dirigirse a la URL del sistema *web* en el navegador.
- Seleccionar “Iniciar sesión”, en el panel superior.
- Digitar usuario y contraseña de Administrador de acuerdo a las credenciales proporcionadas por el equipo desarrollador.
- Para cambiar la contraseña, seleccionar “Olvidaste la contraseña” y en la siguiente pantalla indicar un medio de contacto con el equipo de desarrollo.

**Resultado deseado:**

El sistema *web* permite iniciar sesión al usuario de rol administrador y contactarse para un posible cambio de contraseña.

**Evaluación de la prueba:**

El cliente aprueba al 100% con la verificación de los resultados esperados.

**TABLA XXIV: Prueba de Aceptación N°3 – Consumir varios *endpoints* para gestionar sucursales.**

<b>PRUEBA DE ACEPTACIÓN</b>	
<b>Identificador (ID):</b> PA003	<b>Identificador historia de Usuario:</b> HU03
<b>Nombre:</b> Consumir <i>endpoints</i> para gestionar sucursales	
<p><b>Descripción:</b> El usuario administrador tiene la posibilidad de crear, visualizar, modificar y eliminar información respecto a:</p> <ul style="list-style-type: none"> <li>• Sucursales.</li> </ul> <p>Para el listado, registro y modificación de una sucursal se lo realiza mediante un formulario donde se ingresa la siguiente información:</p> <ul style="list-style-type: none"> <li>• Nombre.</li> <li>• Ciudad.</li> <li>• Teléfono.</li> <li>• Dirección.</li> </ul>	
<b>Pasos de ejecución:</b>	
<ul style="list-style-type: none"> <li>• Previo el inicio de sesión del usuario del rol administrador, en la página de módulos seleccionar “Gestión de sucursales”.</li> <li>• Se visualiza el listado de sucursales con su respectiva información.</li> <li>• Seleccionar “Nuevo” para registrar una nueva sucursal, a través de un formulario.</li> <li>• Seleccionar “Editar” en la sucursal que se desea modificar y llenar el formulario.</li> </ul>	



- Seleccionar “Borrar” en la sucursal que se desea eliminar y confirmar.
- Seleccionar una de las opciones de filtrado para buscar la información de una de las sucursales.

**Resultado deseado:**

El sistema *web* permite visualizar las sucursales de la cadena de farmacias FARMECC con la opción de editar y actualizar información sobre las mismas, además de añadir nuevas sucursales.

**Evaluación de la prueba:**

El cliente aprueba al 100% con la verificación de los resultados esperados.

**TABLA XXV: Prueba de Aceptación N°4 – Consumir varios *endpoints* para visualizar productos.**

<b>PRUEBA DE ACEPTACIÓN</b>	
<b>Identificador (ID):</b> PA004	<b>Identificador historia de Usuario:</b> HU004
<b>Nombre:</b> Consumir varios <i>endpoints</i> para visualizar productos.	
<b>Descripción:</b> El usuario administrador y cliente necesitan consumir varios <i>endpoints</i> para visualizar productos. A través de la página principal del Sistema Web, en el módulo “Catálogo” los usuarios tienen la posibilidad de ver todos los productos existentes.	
<b>Pasos de ejecución:</b> <ul style="list-style-type: none"> <li>• Dirigirse a la URL del sistema <i>web</i> en el navegador.</li> <li>• Seleccionar el módulo “Catálogo”, en el panel superior para ver un listado de los productos existentes.</li> </ul>	
<b>Resultado deseado:</b> <p>El sistema <i>web</i> permite ver información de los productos existentes en las diferentes sucursales, verificando su estado y resumen de cantidades.</p>	
<b>Evaluación de la prueba:</b> <p>El cliente aprueba al 100% con la verificación de los resultados esperados.</p>	

**TABLA XXVI: Prueba de Aceptación N°5 – Consumir varios *endpoints* para visualizar reportes de sucursales.**

<b>PRUEBA DE ACEPTACIÓN</b>	
<b>Identificador (ID):</b> PA005	<b>Identificador historia de Usuario:</b> HU005
<b>Nombre:</b> Consumir varios <i>endpoints</i> visualizar reportes de sucursales.	
<p><b>Descripción:</b> El usuario administrador necesita consumir varios <i>endpoints</i> para visualizar los reportes de ventas de las sucursales. En este sentido, es necesario ingresar al módulo “Reportes de sucursales” y seleccionar cómo se requiere generar el reporte:</p> <ul style="list-style-type: none"> <li>• Reporte por sucursal.</li> <li>• Reporte por rango de fechas (inicial y final).</li> <li>• Reporte general de ventas.</li> </ul> <p>posteriormente, se ingresa los datos requeridos respectivamente y se genera un reporte.</p>	
<p><b>Pasos de ejecución:</b></p> <ul style="list-style-type: none"> <li>• Previo el inicio de sesión del usuario del rol administrador, en panel lateral de módulos seleccionar “Reportes de sucursales”.</li> <li>• Se visualiza 3 opciones para la generación de reportes.</li> <li>• Si el usuario selecciona “Reportes por sucursales” únicamente debe especificar el nombre de la sucursal y presionar “Generar reporte”.</li> <li>• Si el usuario selecciona “Reportes por fechas” se debe ingresar una fecha inicial y final” y presionar “Generar reporte”.</li> <li>• Si el usuario necesita un reporte total de ventas se debe únicamente presionar “Generar reporte”.</li> </ul>	
<p><b>Resultado deseado:</b></p> <p>El sistema <i>web</i> permite al usuario administrador seleccionar 3 opciones de generación de reportes en PDF según lo requiera.</p>	
<p><b>Evaluación de la prueba:</b></p> <p>El cliente aprueba al 100% con la verificación de los resultados esperados.</p>	

**TABLA XXVII: Prueba de Aceptación N°6 – Consumir varios *endpoints* para visualizar disponibilidad de *stock* de productos**

<b>PRUEBA DE ACEPTACIÓN</b>	
<b>Identificador (ID):</b> PA006	<b>Identificador historia de Usuario:</b> HU006
<b>Nombre:</b> Consumir varios <i>endpoints</i> para visualizar disponibilidad de <i>stock</i> de productos.	
<b>Descripción:</b> El usuario administrador necesita consumir varios <i>endpoints</i> para visualizar disponibilidad de <i>stock</i> de productos. Mediante el módulo “Stock de productos”, el usuario busca la sucursal que desea consultar, una vez seleccionada, se presenta los productos con los que cuenta y su estado: agotado, por agotarse y disponible.	
<b>Pasos de ejecución:</b> <ul style="list-style-type: none"> <li>• Previo el inicio de sesión del usuario del rol administrador, en panel lateral de módulos seleccionar “<i>Stock</i> de productos”.</li> <li>• Se visualiza el listado de productos con su respectiva información de sucursal y estado (disponible, por agotarse y agotado).</li> <li>• Seleccionar la advertencia “Agotado” para dirigirse al módulo de “Pedidos a proveedores</li> <li>• Para filtrar los productos por sucursal, únicamente especificar el nombre de la sucursal.</li> <li>• Para buscar un producto en especial, ingresar en el campo de búsqueda.</li> </ul>	
<b>Resultado deseado:</b> El sistema <i>web</i> permite la búsqueda de los productos existentes en las sucursales y determinar el estado de los mismos.	
<b>Evaluación de la prueba:</b> El cliente aprueba al 100% con la verificación de los resultados esperados.	

**TABLA XXVIII: Prueba de Aceptación N°7 – Consumir varios *endpoints* para visualizar la venta por sucursales.**

<b>PRUEBA DE ACEPTACIÓN</b>	
<b>Identificador (ID):</b> PA007	<b>Identificador historia de Usuario:</b> HU007
<b>Nombre:</b> Consumir varios <i>endpoints</i> para visualizar la venta por sucursales.	

<p><b>Descripción:</b> El usuario administrador necesita consumir varios <i>endpoints</i> para visualizar la venta de sucursales. Mediante el módulo “Ventas”, el usuario busca la sucursal que desea consultar, una vez seleccionada, se presenta una lista con las ventas realizadas junto con las descripciones de ventas, fechas, valores recaudados por factura o comprobante de venta y al final, un total de las ventas de la sucursal seleccionada.</p>
<p><b>Pasos de ejecución:</b></p> <ul style="list-style-type: none"> <li>• Previo el inicio de sesión del usuario del rol administrador, en panel lateral de módulos seleccionar “Ventas”.</li> <li>• Se visualiza el listado de ventas totales a nivel nacional de todas las sucursales.</li> <li>• Para filtrar, indicar la sucursal a la que se requiere conocer las ventas.</li> <li>• Seleccionar “Generar reporte” en caso se desee redirigirse a la página de generación de reportes.</li> </ul>
<p><b>Resultado deseado:</b></p> <p>El sistema <i>web</i> permite visualizar las ventas en tiempo real, junto con un estado de venta y las cantidades totales de las sucursales según se requiera.</p>
<p><b>Evaluación de la prueba:</b></p> <p>El cliente aprueba al 100% con la verificación de los resultados esperados.</p>

**TABLA XXIX: Prueba de Aceptación N°8 – Consumir varios *endpoints* para gestionar pedidos con proveedores.**

PRUEBA DE ACEPTACIÓN	
Identificador (ID): PA008	Identificador historia de Usuario: HU008
<b>Nombre:</b> Consumir varios <i>endpoints</i> para gestionar pedidos con proveedores.	
<p><b>Descripción:</b> El usuario administrador necesita consumir varios endpoints para gestionar pedidos con proveedores. En este sentido, el administrador tiene la posibilidad de crear, visualizar, modificar y eliminar información respecto a:</p> <ul style="list-style-type: none"> <li>• Pedidos con proveedores.</li> <li>• Proveedores</li> </ul> <p>Para el registro y modificación de pedidos se realiza mediante un formulario donde se ingresa la siguiente información:</p> <ul style="list-style-type: none"> <li>• Nombre del producto.</li> <li>• Nombre del proveedor.</li> </ul>	

<ul style="list-style-type: none"> <li>• Cantidad.</li> </ul>
<p><b>Pasos de ejecución:</b></p> <ul style="list-style-type: none"> <li>• Previo el inicio de sesión del usuario del rol administrador, en panel lateral de módulos seleccionar “Pedidos a proveedores”.</li> <li>• Seleccionar “Nuevo” para agregar un pedido a través de un formulario.</li> <li>• Seleccionar “Editar” para modificar un pedido.</li> <li>• Seleccionar “Borrar” para eliminar un registro de pedido.</li> <li>• Dirigirse a la parte inferior de la página para visualizar los proveedores disponibles.</li> <li>• Seleccionar “Nuevo” de la sección para agregar un nuevo proveedor a través de un formulario.</li> <li>• Seleccionar “Editar” para modificar un proveedor.</li> <li>• Seleccionar “Borrar” para eliminar un registro de proveedor.</li> </ul>
<p><b>Resultado deseado:</b></p> <p>El sistema <i>web</i> permite gestionar pedidos a proveedores (crear, modificar y eliminar) así como gestionar los proveedores disponibles.</p>
<p><b>Evaluación de la prueba:</b></p> <p>El cliente aprueba al 100% con la verificación de los resultados esperados.</p>

**TABLA XXX: Prueba de Aceptación N°9 – Consumir varios *endpoints* para visualizar el personal.**

<b>PRUEBA DE ACEPTACIÓN</b>	
<b>Identificador (ID):</b> PA009	<b>Identificador historia de Usuario:</b> HU009
<b>Nombre:</b> Consumir varios <i>endpoints</i> para visualizar el personal.	
<b>Descripción:</b> El usuario administrador necesita consumir varios <i>endpoints</i> para visualizar el personal de la red de farmacias FARMECC de forma actualizada. Mediante el módulo “Personal” el administrador puede visualizar la información de empleados, rol, experiencia laboral, contactos y horarios. Si desea el administrador, se puede categorizar la información de los empleados por sucursal. Para la búsqueda de empleados, es necesario el ingreso del nombre completo del empleado.	

**Pasos de ejecución:**

- Previo el inicio de sesión del usuario del rol administrador, en panel lateral de módulos seleccionar “Personal”.
- Para filtrar, indicar la sucursal a la que se quiere referirse.
- Para la búsqueda, simplemente llenar el campo de búsqueda.

**Resultado deseado:**

El sistema *web* permite visualizar a los empleados contratados en las distintas sucursales, además que permitir el filtrado por sucursal y nombre o apellido.

**Evaluación de la prueba:**

El cliente aprueba al 100% con la verificación de los resultados esperados.

**TABLA XXXI: Prueba de Aceptación N°10 – Consumir varios *endpoints* para realizar cotizaciones de productos.**

<b>PRUEBA DE ACEPTACIÓN</b>	
<b>Identificador (ID):</b> PA010	<b>Identificador historia de Usuario:</b> HU010
<b>Nombre:</b> Consumir varios <i>endpoints</i> para realizar cotizaciones de productos.	
<b>Descripción:</b> El usuario cliente necesita consumir varios endpoints para realizar cotizaciones de mercadería. Mediante el módulo “Catálogo” de la página principal del sistema <i>web</i> , los clientes pueden hacer la búsqueda individual o acumulativa de productos y finalmente generar la cotización.	
<b>Pasos de ejecución:</b>	
<ul style="list-style-type: none"> <li>• Dirigirse a la URL del sistema <i>web</i> en el navegador.</li> <li>• Seleccionar el módulo “Catálogo”, en el panel superior para visualizar los productos disponibles.</li> <li>• Seleccionar “+” para iniciar una cotización ingresando los productos de preferencia.</li> <li>• Seleccionar “Imprimir” para emitir un impreso de la cotización realizada.</li> <li>• Seleccionar “Vaciar” para comenzar una nueva lista de productos a cotizar.</li> </ul>	

**Resultado deseado:**

El sistema *web* permite visualizar el cotizador *web* de la cadena de farmacias FARMECC con un módulo que genera una vista previa de Cotización productos así mismo la sumatoria de PVP de los productos seleccionados por el usuario consumidor.

**Evaluación de la prueba:**

El cliente aprueba al 100% con la verificación de los resultados esperados.

## Certificado de la cadena de farmacias FARMECC



FARMACIAS FARMECC.

Quito, 20 de agosto del 2022

### CERTIFICADO

Yo, Efrain Caza Chiluiza, con Cl. 1722656194, como gerente general de FARMACIAS FARMECC ubicada en Quito – Ecuador.

Por medio de la presente certifico:

Que el Srta. **Diana Katherine Almeida Anchatuña** con cédula de ciudadanía **1727369132**, estudiante de la carrera de Desarrollo de Software, realizó su trabajo de integración curricular de desarrollo de un sistema para la gestión del inventario en FARMECC (sistema *web*), mismo que cumple con los requerimientos y funcionalidades definidas en las reuniones mantenidas.

Es todo en cuanto puedo mencionar en honor a la verdad pudiendo el interesado hacer uso de este documento como estime conveniente.

Atentamente:

EFRAIN CAZA CHILUIZA

LICENCIADO RADIÓLOGO

0967676843





## ANEXO III

El siguiente enlace, permite observar el Manual de Usuario, donde se detalla la información con respecto a las funcionalidades del sistema *web*, así como la participación de los perfiles en la interacción.

<https://youtu.be/v-SxqoxMKGE>

## ANEXO IV

A continuación, se especifica las credenciales de acceso para el sistema *web*, así como el enlace al repositorio en *GitHub* en donde se encuentra el código fuente y en el apartado de README los pasos para realizar la instalación de forma local.

### **Credenciales de acceso para el sistema *web***

Para acceder al sistema *web* en producción, ingresar a la siguiente URL:

<https://web-farmecc.herokuapp.com/>

Credenciales para el perfil administrador:

- **Usuario:** Administrador
- **Contraseña:** admin1234

### **Repositorio del código fuente del sistema *web***

El código fuente de todo el proyecto, se encuentra alojado en el repositorio *GitHub*, el cual se puede acceder a través de la siguiente URL:

[https://github.com/AlmDiana/SistWeb\\_FARMECC.git](https://github.com/AlmDiana/SistWeb_FARMECC.git)