


**PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR**

**FACULTAD DE INGENIERÍA**

**ESCUELA DE SISTEMAS**

**DISERTACIÓN PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERA EN SISTEMAS Y COMPUTACIÓN**

# ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE PRODUCTOS Y CITAS DE PROVEEDORES EN UNA EMPRESA DE DISTRIBUCIÓN DE PRODUCTOS MASIVOS. CASO DE ESTUDIO: ASERTIA

**CRISTIAN DAVID REINOSO DE LA CRUZ**

**DIRECTOR: ING. CHARLES ESCOBAR**

**QUITO, 2020**

**RESUMEN**

El presente trabajo de titulación pretende desarrollar e implementar una aplicación web utilizando la metodología ágil SCRUM que le permita gestionar productos y citas de los proveedores de la empresa Asertia. Para desarrollar e implementar la aplicación web primeramente se realizó el levantamiento de requerimiento con los posibles usuarios de la aplicación web mediante historias de usuario, posteriormente se realizó el diseño y selección de las herramientas para construir la aplicación web. Tomando en cuenta los requerimientos, el diseño de la aplicación y las herramientas que se utilizarán, se estableció el modelo de la base de datos en donde se almacenará la información que manejará la aplicación web.

Utilizando varios estándares de programación se procedió a desarrollar, en tres Sprints, los requerimientos previamente planteados de la aplicación web. Finalmente se realizaron las pruebas sobre cada requerimiento desarrollado. Una vez aprobadas las pruebas, se integró la aplicación web a la empresa Asertia.

## DEDICATORIA

*Quiero dedicar el desarrollo de este trabajo de titulación a mis padres, hermanos, familiares y amigos quienes han sido un pilar fundamental en mi vida.*

## AGRADECIMIENTOS

*Quiero agradecer infinitamente a mis padres hermanos y familiares quienes me han brindado su apoyo a lo largo de mi vida.*

*Agradezco a mi director Ing. Charles Escobar y revisores Ing. Edison Mora, Ing. Beatriz Campos quienes me han ayudado a llevar a cabo este trabajo de titulación con sus sabios consejos y observaciones.*

*Agradezco a todos mis profesores a lo largo de este periodo de pregrado quienes han sabido compartir su conocimiento de la mejor manera.*

*Agradezco a mis amigos quienes me han ayudado enormemente a lo largo de la carrera y han sabido darme su consejo incondicional.*

**CONTENIDO**

CONTENIDO ..... 5

ILUSTRACIONES Y TABLAS ..... 9

1 CAPÍTULO 1: INTRODUCCIÓN Y FUNDAMENTOS TEÓRICOS ..... 12

1.1 Tema ..... 12

1.2 Justificación ..... 12

1.3 Planteamiento del Problema ..... 13

1.4 Objetivos ..... 15

1.4.1 Objetivo General ..... 15

1.4.2 Objetivos Específico ..... 15

2 Marco Teórico ..... 16

2.1 Aplicaciones Web ..... 16

2.2 Arquitectura de Sistemas Web ..... 16

2.3 Servicios Web ..... 17

2.4 Frameworks de Desarrollo ..... 18

2.5 Metodologías de Desarrollo ..... 19

2.6 SCRUM ..... 20

2.6.1 Equipo de Scrum y Roles ..... 21

2.6.2 Eventos de Scrum ..... 21

2.6.3 El Sprint ..... 21

2.6.4 Planificación de Sprint (Sprint Planning) ..... 21

2.6.5 Scrum Diario (Daily Scrum) ..... 21

2.6.6 Revisión de Sprint (Sprint Review) ..... 22

2.6.7 Retrospectiva de Sprint (Sprint Retrospective) ..... 22

2.6.8 Artefactos de Scrum ..... 22

2.6.9 Lista de Producto (Product Backlog) ..... 22

2.6.10 Lista de Pendientes del Sprint (Sprint Backlog) ..... 23

2.6.11 Incremento ..... 23

3 CAPITULO 3: ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE PRODUCTOS Y CITAS DE PROVEEDORES PARA LA EMPRESA ASERTIA. ..... 24

3.1 ANÁLISIS DE REQUERIMIENTOS PARA DISEÑAR EL SISTEMA. ..... 24

3.1.1 Historias de Usuario ..... 24

3.1.2 Product Backlog ..... 34

3.1.2.1 Características ..... 34

3.1.3 Descripción del Sistema ..... 45

3.1.3.1 Descripción General ..... 45

3.1.3.2 Funcionalidades Generales del Sistema ..... 46

3.1.3.3 Funciones Enfocadas al Administrador ..... 46

3.1.3.4 Funciones Enfocadas al Proveedor ..... 47

3.1.3.5 Funciones Enfocadas al Comprador ..... 47

3.2 DISEÑO DE LA APLICACIÓN TANTO FRONTEND (SOFTWARE QUE INTERACTÚA CON EL USUARIO) Y BACKEND (MANEJO DE LA LÓGICA DEL NEGOCIO E INTERACCIÓN CON LA BASE DE DATOS). ..... 49

3.2.1 Arquitectura del Sistema ..... 49

3.2.2 Backend ..... 50

3.2.2.1 Modelo de Datos ..... 50

3.2.2.2 Lógica del negocio ..... 50

3.2.2.3 Servicios Web ..... 50

3.2.3 Frontend ..... 51

3.2.3.1 Store ..... 51

3.2.3.2 Reductor ..... 51

3.2.3.3 Acciones ..... 51

3.2.3.4 Vistas ..... 52

3.2.4 Diseño Gráfico del Sistema ..... 52
3.2.4.1 Prototipo del sistema ..... 52
3.3 SELECCIÓN DE HERRAMIENTAS DE DESARROLLO ADECUADAS A SER UTILIZADAS. ..... 69
3.3.1 Base de Datos ..... 69
3.3.2 Backend ..... 72
3.3.2.1 Node.js ..... 73
3.3.2.2 Alojamiento de la aplicación ..... 75
3.3.2.3 Herramientas de Desarrollo ..... 75
3.3.3 Frontend ..... 76
3.3.3.1 React ..... 77
3.3.3.2 Redux ..... 78
3.3.3.3 Alojamiento de la aplicación ..... 79
3.3.3.4 Herramientas de Desarrollo ..... 79
3.4 DEFINICIÓN DEL MODELO DE LA BASE DE DATOS EN DONDE SE ALMACENARÁ LA INFORMACIÓN DEL SISTEMA. ..... 81
3.4.1 Modelo de la Base de Datos ..... 81
3.4.1.1 Usuarios ..... 81
3.4.1.2 Proveedores ..... 82
3.4.1.3 Compradores ..... 83
3.4.1.4 Productos ..... 83
3.4.1.5 Citas ..... 84
3.5 IMPLEMENTACIÓN DE LA APLICACIÓN INCREMENTALMENTE CUMPLIENDO CON LAS FASES DE LA METODOLOGÍA DE DESARROLLO SCRUM. ..... 87
3.5.1 SCRUM ..... 87
3.5.1.1 Especificación de Roles ..... 87

3.5.1.2 Sprint 1 ..... 88
3.5.1.3 Sprint 2 ..... 92
3.5.1.4 Sprint 3 ..... 96
3.6 PRUEBAS E INTEGRACIÓN DE LA APLICACIÓN WEB DESCRITA Y VERIFICACIÓN DEL FUNCIONAMIENTO ..... 102
3.6.1 Pruebas Sprint 1 ..... 102
3.6.2 Pruebas Sprint 2 ..... 110
3.6.3 Pruebas Sprint 3 ..... 119
3.6.4 Integración del Sistema ..... 129
4 CONCLUSIONES Y RECOMENDACIONES ..... 130
4.1 Conclusiones ..... 130
4.2 Recomendaciones ..... 131
5 Bibliografía ..... 133

# ILUSTRACIONES Y TABLAS

Ilustración 1 Prototipo perfil de administrador. Pantalla para entrar al sistema..... 53
Ilustración 2 Prototipo perfil de administrador. Pantalla de compradores ..... 53
Ilustración 3 Prototipo perfil administrador. Pantalla para agregar un comprador ..... 54
Ilustración 4 Prototipo perfil administrador. Pantalla para editar un comprador ..... 54
Ilustración 5 Prototipo perfil de administrador. Pantalla de proveedores ..... 55
Ilustración 6 Prototipo perfil de administrador. Pantalla para editar un proveedor ..... 56
Ilustración 7 Prototipo perfil de administrador. Pantalla para ver productos nuevos..... 56
Ilustración 8 Prototipo perfil de administrador. Pantalla para ver productos revisados..... 57
Ilustración 9 Prototipo perfil de comprador. Pantalla para registrarse ..... 57
Ilustración 10 Prototipo perfil de comprador. Pantalla para cambiar contraseña..... 58
Ilustración 11 Prototipo perfil de comprador. Pantalla para entrar al sistema ..... 58
Ilustración 12 Prototipo perfil de comprador. Pantalla para ver proveedores ..... 59
Ilustración 13 Prototipo perfil de comprador. Pantalla para editar un proveedor ..... 59
Ilustración 14 Prototipo perfil de comprador. Pantalla para ver productos nuevos ..... 60
Ilustración 15 Prototipo perfil de comprador. Pantalla para ver productos revisados..... 61
Ilustración 16 Prototipo perfil de comprador. Tabla de productos..... 61
Ilustración 17 Prototipo perfil de comprador. Pantalla para ver agregar un comentario a productos rechazados..... 62
Ilustración 18 Prototipo perfil de comprador. Pantalla para ver citas ..... 62
Ilustración 19 Prototipo perfil de comprador. Pantalla para agregar una cita ..... 63
Ilustración 20 Prototipo perfil de comprador. Pantalla para editar una cita ..... 63
Ilustración 21 Prototipo perfil de proveedor. Pantalla para registrarse en el sistema ..... 64
Ilustración 22 Prototipo perfil de proveedor. Pantalla para cambiar la contraseña ..... 64
Ilustración 23 Prototipo perfil de proveedor. Pantalla para ingresar al sistema ..... 65
Ilustración 24 Prototipo perfil de proveedor. Pantalla para ver productos nuevos..... 66
Ilustración 25 Prototipo perfil de proveedor. Pantalla para ver productos revisados..... 66
Ilustración 26 Prototipo perfil de proveedor. Pantalla para ver comentarios de productos rechazados ..... 67
Ilustración 27 Prototipo perfil de proveedor. Pantalla para ver citas ..... 67
Ilustración 28 Prototipo perfil de proveedor. Pantalla para reservar una cita ..... 68
Ilustración 29 Roa, A. (2018). Redux [figura]. Recuperado de https://medium.com/@alejandroroa/redux-sin-tanto-dolor-f5c44bc687c ..... 79

Ilustración 30 Modelo de la base de datos..... 85
Ilustración 31 Prueba del sistema. Requerimiento 1 del Sprint 1..... 102
Ilustración 32 Prueba del sistema. Requerimiento 1 mensaje del Sprint 1..... 103
Ilustración 33 Prueba del sistema. Requerimiento 2 del Sprint 1..... 104
Ilustración 34 Prueba del sistema. Requerimiento 3 del Sprint 1..... 104
Ilustración 35 Prueba del sistema. Requerimiento 4 del Sprint 1..... 105
Ilustración 36 Prueba del sistema. Requerimiento 5 del Sprint 1..... 106
Ilustración 37 Prueba del sistema. Requerimiento 6 del Sprint 1..... 107
Ilustración 38 Prueba del sistema. Requerimiento 7 del Sprint 1..... 108
Ilustración 39 Prueba del sistema. Requerimiento 8 del Sprint 1..... 109
Ilustración 40 Prueba del sistema. Requerimiento 9 del Sprint 1..... 109
Ilustración 41 Prueba del sistema. Requerimiento 10 del Sprint 1..... 110
Ilustración 42 Prueba del sistema. Requerimiento 1 carga de productos del Sprint 2 ..... 111
Ilustración 43 Prueba del sistema. Requerimiento 1 del Sprint 2..... 111
Ilustración 44 Prueba del sistema. Requerimiento 2 del Sprint 2..... 112
Ilustración 45 Prueba del sistema. Requerimiento 3 del Sprint 2..... 113
Ilustración 46 Prueba del sistema. Requerimiento 4 del Sprint 2..... 114
Ilustración 47 Prueba del sistema. Requerimiento 5 del Sprint 2..... 115
Ilustración 48 Prueba del sistema. Requerimiento 6 del Sprint 2..... 116
Ilustración 49 Prueba del sistema. Requerimiento 7 del Sprint 2..... 117
Ilustración 50 Prueba del sistema. Requerimiento 8 del Sprint 2..... 117
Ilustración 51 Prueba del sistema. Requerimiento 9 del Sprint 2..... 118
Ilustración 52 Prueba del sistema. Requerimiento 10 del Sprint 2..... 119
Ilustración 53 Prueba del sistema. Requerimiento 1 del Sprint 3..... 119
Ilustración 54 Prueba del sistema. Requerimiento 2 del Sprint 3..... 120
Ilustración 55 Prueba del sistema. Requerimiento 3 del Sprint 3..... 121
Ilustración 56 Prueba del sistema. Requerimiento 4 del Sprint 3..... 121
Ilustración 57 Prueba del sistema. Requerimiento 5 del Sprint 3..... 122
Ilustración 58 Prueba del sistema. Requerimiento 6 del Sprint 3..... 123
Ilustración 59 Prueba del sistema. Requerimiento 7 del Sprint 3..... 124
Ilustración 60 Prueba del sistema. Requerimiento 8 del Sprint 3..... 125
Ilustración 61 Prueba del sistema. Requerimiento 9 del Sprint 3..... 125
Ilustración 62 Prueba del sistema. Requerimiento 10 del Sprint 3..... 126

Ilustración 63  Prueba del sistema. Requerimiento 11 del Sprint 3..... 127
Ilustración 64  Prueba del sistema. Requerimiento 12 del Sprint 3..... 128
Ilustración 65  Prueba del sistema. Requerimiento 13 del Sprint 3..... 128

# 1 CAPÍTULO 1: INTRODUCCIÓN

## 1.1 TEMA

Análisis, diseño e implementación de una aplicación web para la gestión de productos y citas de proveedores en una empresa de distribución de productos masivos. Caso de estudio: Asertia

## 1.2 JUSTIFICACIÓN

Asertia es una empresa líder en operaciones comerciales que se dedica a la distribución de productos masivos con cobertura a nivel nacional. Desde la fundación de Asertia en 1985 ha ido evolucionando de acuerdo con la demanda del mercado manteniéndose como líderes frente a la competencia, uno de los factores de éxito más importantes que ellos reconocen es la tecnología y la innovación como lo mencionan en su misión. Actualmente, Asertia cuenta con alrededor de 200 proveedores y más de 2.000 diferentes productos con los que trabaja a diario y son su principal fuente de ingreso.

La presente investigación busca implementar un sistema web que permita automatizar el proceso para los proveedores de la organización en el cual puedan cargar un archivo en formato CSV con datos de varios productos, el sistema analizara los datos del archivo y validará que tenga el formato establecido por la empresa, dando una retroalimentación al proveedor sobre los productos específicos que no cumplan con el formato y permitirles corregirlos fácilmente la información del producto para que estos puedan ser aprobados por un comprador y estén disponibles para la venta. A su vez reducirá el tiempo al personal del departamento de compras para aceptar o rechazar un producto y seguir con el proceso de registrar productos, tomando en cuenta las necesidades de la empresa.

Además, el sistema permitirá gestionar citas entre compradores y proveedores permitiendo a los compradores publicar un calendario con fecha y hora en los cuales estarán disponibles para que los proveedores puedan agendar citas con la finalidad de realizar negociaciones sobre productos o entregar nueva mercadería, con lo cual facilitará al personal de compras a organizar eficientemente su tiempo y recordarles por medio de envió de correos electrónico al comprador y al proveedor cuando se agende o cancele una reunión.

## 1.3 PLANTEAMIENTO DEL PROBLEMA

Actualmente, el proceso para adquirir productos en la empresa Asertia comienza cuando un proveedor envía un archivo en Excel con la información de los productos que este ofrece a la empresa, el cual es receptado por el personal de compras de la empresa para que ellos lo revisen y verifiquen que todos los productos que constan en el archivo cumplan con el formato de la organización para después tomar la decisión de aceptarlos y estos puedan ser evaluados por un comprador. Una vez aceptado el producto el proveedor y el comprador coordinan una cita para realizar la entrega de mercadería o para realizar nuevas negociaciones sobre ciertos productos. Este proceso es realizado a diario por los compradores de la empresa el cual, en promedio, toma más de 3 horas de trabajo a cada comprador de la empresa. El proceso previamente descrito no es conveniente para la empresa ya que toma más tiempo del que debería y se lo podría automatizar con la ayuda de la tecnología, siendo esta la razón de implementar la aplicación web.

Actualmente no existe un sistema que automatice este proceso, es decir, la empresa no cuenta con un análisis de las funcionalidades que se podría automatizar en el proceso, tampoco cuenta con un diseño técnico del sistema que automatice este proceso ni de las herramientas óptimas para construirlo. Asimismo, se necesita una metodología ágil que permita llevar a

cabo la automatización de dicho proceso generando como resultado un producto final de calidad que cumpla con su funcionalidad de manera eficaz y eficiente.

Del análisis previo se puede identificar como problema principal:

- No se cuenta con una aplicación web para la gestión de productos y citas de proveedores para la empresa Asertia.

Y los siguientes problemas secundarios derivados:

- No se ha realizado un análisis de requerimientos para diseñar el sistema.
- No se ha diseñado una aplicación tanto frontend y backend para la aplicación.
- No se ha definido que herramientas de desarrollo se pueden utilizar para la implementación del sistema.
- No se ha definido un modelo para la base de datos en donde se almacene la información del sistema.
- La implementación no podría desarrollarse con metodologías no ágiles
- No se tienen pruebas sobre la aplicación web que permitan verificar el correcto funcionamiento de esta.
- La empresa no cuenta con un sistema web que se integra a sus aplicaciones y actividades.

De estos problemas se pude plantear la pregunta de investigación de la siguiente manera:

¿Qué aplicación web se puede diseñar e implementar para la gestión de productos y citas de proveedores para la empresa Asertia?

## 1.4 OBJETIVOS

### 1.4.1 Objetivo General

Analizar, diseñar e implementar una aplicación web para la gestión de productos y citas de proveedores para la empresa Asertia.

### 1.4.2 Objetivos Específico

- Realizar el análisis de requerimientos para diseñar el sistema.
- Realizar el diseño de la aplicación tanto frontend (Software que interactúa con el usuario) y backend (Manejo de la lógica del negocio e interacción con la base de datos).
- Seleccionar las herramientas de desarrollo adecuadas a ser utilizadas.
- Establecer el modelo de la base de datos en donde se almacenará la información del sistema.
- Realizar la implementación de la aplicación incrementalmente cumpliendo con las fases de la metodología de desarrollo SCRUM
- Realizar las pruebas e integración sobre la aplicación web descrita y verificar el correcto funcionamiento de esta.

# 2 CAPÍTULO 2: MARCO TEÓRICO

## 2.1 APLICACIONES WEB

Las aplicaciones web son programas alojados en servidores web capaces de tener una interacción sofisticada con el usuario, a los cuales para acceder solo se necesita tener conexión a Internet y contar con un navegador web, independientemente del dispositivo móvil que los usuarios estén utilizando (tablets, teléfonos inteligentes o computadoras) y del sistema operativo de los mismos.

Entre las principales funcionalidades que las aplicaciones web brindan es que se puede acceder al sistema desde cualquier lugar físico a cualquier hora del día, es decir, no es necesario estar conectado a una red especifica. Además, todos los usuarios del sistema pueden tener acceso a la información que les corresponda dependiendo su función dentro del sistema.

El comportamiento general de las aplicaciones web se lo puede describir como el envío de peticiones al servidor web en el cual se encuentra alojada la aplicación. El sistema procesa la petición y de ser necesario obtiene información de la base de datos en donde se encuentra la información que maneja el sistema, la cual puede estar alojada en el mismo servidor web o no. Después que el servidor procesa la petición envía una respuesta a la aplicación la cual es procesada y presentada al usuario final.

## 2.2 ARQUITECTURA DE SISTEMAS WEB

La arquitectura de la mayoría de las aplicaciones web se las puede dividir en tres capas:

- Frontend: se enfoca en la interfaz del usuario y como este interactúa con el sistema.
- Backend: se enfoca en manejar la lógica del negocio.
- Modelo: se enfoca en manejar la información de la base de datos.

"Uno de los patrones que ha demostrado ser fundamental a la hora de diseñar aplicaciones web es el Modelo-Vista-Control (MVC). Este patrón propone la separación en distintos componentes de la interfaz de usuario (vistas), el modelo de negocio y la lógica de control." (Castejón, 2004)

## 2.3 SERVICIOS WEB

El desarrollo de servicios web se ha convertido en un estándar en la programación al momento de compartir información entre diferentes aplicaciones de software, ya que para distribuir sus funcionalidades estos son independientes del hardware, el sistema operativo, el lenguaje de programación y al igual que son independientes para proveer las funcionalidades, son independientes al momento de ejecutarse y desarrollarse para su uso. (Hamad, Saad, & Abed, 2010).

REST (Representational State Transfer) en un tipo de arquitectura orientada a recursos para el desarrollo de servicios web reemplazando especificaciones como SOAP (Pai, M, Asnika, & R, 2004). El desarrollo de servicios web REST evita tener que desarrollar la misma aplicación para diferentes plataformas, ni tampoco depende del lenguaje o del ordenador en el cual se esté utilizando la aplicación web, sino que gracias a la arquitectura de servicios web REST es suficiente con exponer el servicio web y de esta manera poder conectarse al mismo desde cualquier otro tipo de aplicación. Es importante mencionar que REST no está relacionado únicamente con el protocolo HTTP, pero es con el que mayormente se lo asocia.

Entre los principales principios que cumplen los servicios web basados en REST son:

- Utiliza verbos de HTTP de manera explícita los cuales tienen una función en específico:

- GET: obtiene un recurso desde el servidor
- POST: crea un nuevo recurso en el servidor
- PUT: actualiza o cambia el estado de un recurso
- DELETE: elimina un recurso.

- Tiene un escalonamiento frente a la demanda constante mediante balanceador, clúster y servidores con alta disponibilidad con la finalidad de reducir el tiempo de respuesta de una petición.
- La estructura de las URI es fácil de entender y pueden ser utilizadas intuitivamente
- Emiten una respuesta en formato JSON o XML los cuales pueden ser consumidos por cualquier aplicación desarrollada en cualquier lenguaje.

Para conectarse a un servicio web REST "El cliente tiene acceso a los recursos usando una única URI y una representación del recurso es devuelto con un estado de la transferencia de la información." (Pai, M, Asnika, & R, 2004)

## 2.4 FRAMEWORKS DE DESARROLLO

Existen varios frameworks y lenguajes de programación que permiten crear una aplicación web. Sin embargo, Node.js que es una plataforma de ejecución que permite ejecutar código JavaScript de parte del servidor que al combinarse con Express.js que es un framework que funciona sobre Node.js permite generar enrutamiento para trabajar sobre el protocolo HTTP. (Edward, Teresa, Peñaherrera, & Quiña, 2019)

Node.js es un entorno de ejecución orientado a eventos asíncronos de JavaScript diseñado para construir aplicaciones de red escalables (Node.js, 2018). Node.js permanece dormido

si es que no hay trabajo por hacer, hasta que haya una conexión para que se active una llamada al servicio, todo esto es posible ya que se usa JavaScript del lado de servidor. (w3schools.com, 2018).

Node.js cuenta con un gestor de paquete de datos (NPM) el cual nos permite distribuir paquetes, administrar nuestros módulos y agregar dependencias de una manera ágil y sencilla.

Al igual que para el desarrollo backend. Existen varios frameworks para desarrollar la capa del frontend de la aplicación web, entre los más conocidos están: React.js, Vue.js y Angular, de los cuales React.js es la que mejor interactúa con servicios web REST. “React es una biblioteca de JavaScript declarativa, eficiente y flexible para crear interfaces de usuarios. Permite componer interfaces de usuario complejas a partir de pequeñas y aisladas piezas líneas de código llamadas componentes.” (React.Js, 2018)

Otro pilar fundamental dentro del desarrollo de una aplicación web es sin duda, las bases de datos las cuales permiten almacenar información de manera organizada y darle diferentes propósitos y usos. Los datos quedan organizados de manera similar en base a algún factor en común que compartan, de manera que facilita encontrar y utilizar la información que necesitamos en cada momento. (Apser, 2015)

## 2.5 METODOLOGÍAS DE DESARROLLO

Existen varias metodologías para desarrollar aplicaciones web de una manera organizada y consistente. Avison y Fiztgerald definen a la metodología como “una colección de procedimientos, técnicas, herramientas y documentos auxiliares que ayudan a los desarrolladores de software en sus esfuerzos por implementar nuevos sistemas de información. Una metodología está formada por fases, cada una de las cuales se puede dividir en sub-fases, que guiarán a los desarrolladores de sistemas a elegir las técnicas más

apropiadas en cada momento del proyecto y también a planificarlo, gestionarlo, controlarlo y evaluarlo.” (Avison & Fiztgerald, 1998)

Para escoger una metodología de desarrollo de software es importante considerar varios aspectos relacionados al proyecto al cual se le aplicará la metodología de desarrollo. Entre estos aspectos se encuentran: que tan involucrados estará el cliente durante el desarrollo del sistema, el grado de familiaridad que se tenga acerca del problema que se pretende resolver y la probabilidad que existan cambios durante el desarrollo del proyecto, entre otros.

Uno de los tipos de metodologías de desarrollo más actuales y utilizadas para desarrollar aplicaciones web son las metodologías ágiles, las cuales intentan evitar procesos tediosos y burocráticos de las metodologías tradicionales y se enfocan en iteraciones de desarrollo que generan entregables periódicamente durante todo el ciclo de vida del proyecto. Cada iteración de desarrollo del ciclo de vida contempla la planificación, análisis de requerimientos, diseño, desarrollo y pruebas, de esta manera se busca minimizar riesgos o errores que puedan tener un gran impacto en la calidad del producto

Entre las metodologías ágiles más utilizadas para el desarrollo de aplicaciones web se encuentra Scrum que “fue concebido por Jeff Sutherland y su equipo de desarrollo a principios de la década de 1990 y se utiliza para guiar actividades de desarrollo” (Pressman, 2010) , cada desarrollo incluye las actividades de : análisis, diseño, evolución y entrega. Scrum nombra a cada una de estas actividades de desarrollo como Sprint.

## 2.6 SCRUM

Scrum es una metodología o marco de trabajo que facilita la gestión de desarrollo de productos. Scrum está compuesto por: equipo de Scrum y sus roles, eventos y artefactos. (Sutherland y Schwaber, 2016).

### 2.6.1 Equipo de Scrum y Roles

El equipo de Scrum está conformado por el dueño del producto (Product Owner), el Equipo de desarrollo (Development Team) y el Scrum Master. Sin embargo, es importante considerar a los involucrados del proyecto los cuales son: el cliente, el usuario y todas las personas que tendrán un tipo de relación con el sistema web.

### 2.6.2 Eventos de Scrum

Los eventos de Scrum están definidos por un tiempo determinado y cada uno tienen un objetivo específico con los cuales podremos llevar a cabo el desarrollo de un producto. Los eventos de Scrum son:

### 2.6.3 El Sprint

El Sprint es considerado el corazón de Scrum ya que contiene a los demás eventos, el sprint es un bloque de tiempo de hasta 4 semanas máximo y su objetivo principal es crear un incremento del producto terminado utilizable (Sutherland y Schwaber, 2016).

### 2.6.4 Planificación de Sprint (Sprint Planning)

La planificación del sprint es una reunión de no más de 8 horas en la cual está involucrada todo el equipo de Scrum y su objetivo principal es planificar el trabajo a realizar en el Sprint.

La Planificación de Sprint responde a las siguientes preguntas: ¿Qué puede entregarse en el Incremento resultante del Sprint que comienza? y ¿Cómo se conseguirá hacer el trabajo necesario para entregar el Incremento? (Sutherland y Schwaber, 2016).

### 2.6.5 Scrum Diario (Daily Scrum)

El Scrum diario es una reunión entre el Scrum Master y el equipo de desarrollo la cual se realiza antes de empezar a trabajar en el desarrollo del producto y su tiempo de duración es

de 15 minutos. El objetivo de esta reunión es que el equipo de desarrollo planifique las actividades que realizará para lograr el objetivo del Sprint.

### 2.6.6 Revisión de Sprint (Sprint Review)

La revisión del Sprint es una reunión que se realiza al final de cada Sprint, los involucrados en esta reunión son el equipo Scrum y el cliente. El objetivo principal de esta reunión será socializar al cliente las actividades realizadas en el Sprint y presentar el incremento del producto terminado para que este comience a ser utilizado por los usuarios.

#### 2.6.7 Retrospectiva de Sprint (Sprint Retrospective)

La Retrospectiva de Sprint es una reunión entre el equipo Scrum la cual se realiza después de finalizar la revisión del Sprint. El objetivo de esta reunión es analizar el desempeño del equipo Scrum e identificar lo que se realizó exitosamente y en qué se puede mejorar para el siguiente Sprint.

### 2.6.8 Artefactos de Scrum

Sutherland y Schwaber (2016) afirman que: “Los artefactos definidos por Scrum están diseñados específicamente para maximizar la transparencia de la información clave, necesaria para asegurar que todos tengan el mismo entendimiento del artefacto” (p 14).

#### 2.6.9 Lista de Producto (Product Backlog)

La lista de producto son todos los requerimientos establecidos entre el cliente y el Product Owner. Estos requerimientos son redactados por el Product Owner. Cada requerimiento cuenta con una prioridad en base a su nivel de importancia para ser desarrollada y con una estimación del tiempo de desarrollo.

### 2.6.10Lista de Pendientes del Sprint (Sprint Backlog)

El Sprint Backlog o lista de pendientes del Sprint es una porción de requerimientos obtenidos del Product Backlog que serán desarrollados en el siguiente Sprint

### 2.6.11Incremento

El incremento es una parte del producto que puede ser utilizada por el cliente. El incremento de la aplicación se obtiene al finalizar cada Sprint. Para esto, es necesario cumplir con los requerimientos detallados en el Sprint Backlog.

# 3 CAPITULO 3: ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE PRODUCTOS Y CITAS DE PROVEEDORES PARA LA EMPRESA ASERTIA.

## 3.1 ANÁLISIS DE REQUERIMIENTOS PARA DISEÑAR EL SISTEMA.

El objetivo de esta sección es desarrollar el proceso de levantamiento de requerimientos con el cliente. Posteriormente se realizará un análisis de acuerdo con la metodología de desarrollo ágil SCRUM, gracias a la cual a cada requerimiento identificado se le asignará una estimación de tiempo de desarrollo y una prioridad la cuál define un orden de desarrollo. El producto que se generará al final de este capítulo es el product backlog.

### 3.1.1 Historias de Usuario

Después de haber sostenido una reunión con los involucrados de los procesos de aprobar productos y agendar citas, se logró comprender exitosamente el contexto bajo el cual nace la necesidad de desarrollar el presente sistema, así como la visión que deberá tener. Finalizada la reunión se logró identificar y documentar 27 historias de usuario de distintas personas con los perfiles de: administrador, proveedor y comprador. Las historias de usuario documentadas son descripciones cortas de funcionalidades del sistema narradas desde la perspectiva del usuario que necesita dicha funcionalidad. A continuación, se detallan las historias de usuario en su versión inédita.

1. “Como administrador del sistema quiero que los proveedores se puedan registrar desde cualquier lugar en el sistema para que se les haga más fácil ese proceso.” (J. De La Cruz, comunicación personal, 17 de septiembre de 2019).

2. “Como administrador del sistema quiero que los compradores se puedan registrar desde cualquier lugar en el sistema para que se les haga más fácil ese proceso.” (J. De La Cruz, comunicación personal, 17 de septiembre de 2019).
3. “Como administrador del sistema quiero crear compradores personalmente.” (J. De La Cruz, comunicación personal, 17 de septiembre de 2019).
4. “Como administrador del sistema quiero asignar un comprador a cada proveedor registrado en el sistema para que sean atendidos por sus respectivos compradores.” (J. De La Cruz, comunicación personal, 17 de septiembre de 2019).
5. “Como administrador del sistema quiero ver todos los proveedores registrados en el sistema para saber que proveedores tengo y cuantos.” (J. De La Cruz, comunicación personal, 17 de septiembre de 2019).
6. “Como administrador del sistema quiero ver todos los compradores registrados en el sistema para saber que compradores tengo y cuantos.” (J. De La Cruz, comunicación personal, 17 de septiembre de 2019).
7. “Como proveedor quiero subir un archivo en formato CSV con la información de mis productos para enviar a mi comprador asignado y realizar la venta.” (F. Andino, comunicación personal, 17 de septiembre de 2019).
8. “Como proveedor quiero corregir información de un producto del archivo en formato CSV en el sistema para que sea más rápido editar la información de mis productos.” (F. Andino, comunicación personal, 17 de septiembre de 2019).
9. “Como proveedor quiero buscar un producto por el código interno del producto o por el nombre del producto o por el código de producto proveedor o por la categoría del producto o por la subcategoría del producto o por la familia del producto.” (F. Andino, comunicación personal, 17 de septiembre de 2019).

10. “Como proveedor quiero ordenar los productos para facilitar el análisis de los datos.” (F. Andino, comunicación personal, 17 de septiembre de 2019).
11. “Como proveedor quiero ver los productos revisados y los productos no revisados.” (F. Andino, comunicación personal, 17 de septiembre de 2019).
12. “Como proveedor quiero ver el estado de los productos revisados por un comprador para saber si fueron aprobados o rechazados.” (F. Andino, comunicación personal, 17 de septiembre de 2019).
13. “Como proveedor quiero ver el comentario de porqué fueron rechazados los productos.” (F. Andino, comunicación personal, 17 de septiembre de 2019).
14. “Como proveedor quiero ver el horario disponible de mi comprador asignado para agendar una cita.” (F. Andino, comunicación personal, 17 de septiembre de 2019).
15. “Como proveedor quiero que me llegué un correo cuando un comprador cancela una cita para hablar con él.” (F. Andino, comunicación personal, 17 de septiembre de 2019).
16. “Como comprador quiero publicar los días y horas en las que puedo atender a mis proveedores para recibir mercadería.” (P. Tovar, comunicación personal, 17 de septiembre de 2019).
17. “Como comprador quiero que me llegué un correo cuando un proveedor agenda una cita para tomarla en cuenta.” (P. Tovar, comunicación personal, 17 de septiembre de 2019).
18. “Como comprador quiero que me llegué un correo cuando un proveedor cancela una cita para hablar con él.” (P. Tovar, comunicación personal, 17 de septiembre de 2019).

19. “Como comprador quiero ver todos los productos enviados de mis proveedores para aceptar o rechazar productos.” (P. Tovar, comunicación personal, 17 de septiembre de 2019).
20. “Como comprador quiero editar información de los productos enviados por mis proveedores para agregar información interna de la empresa.” (P. Tovar, comunicación personal, 17 de septiembre de 2019).
21. “Como comprador quiero buscar un producto por el código interno del producto o por el nombre del producto o por el código de producto proveedor o por la categoría del producto o por la subcategoría del producto o por la familia del producto.” (P. Tovar, comunicación personal, 17 de septiembre de 2019).
22. “Como comprador quiero ordenar los productos para facilitar el análisis de los datos.” (P. Tovar, comunicación personal, 17 de septiembre de 2019).
23. “Como comprador quiero ver los productos revisados y los productos no revisados.” (P. Tovar, comunicación personal, 17 de septiembre de 2019).
24. “Como comprador quiero poder aprobar todos los productos o rechazar todos los productos para no tener que hacer este proceso por cada producto por revisar.” (P. Tovar, comunicación personal, 17 de septiembre de 2019).
25. “Como comprador quiero descargarme un archivo en formato CSV con la información de los productos aprobados enviada por los proveedores más la información interna de la empresa.” (P. Tovar, comunicación personal, 17 de septiembre de 2019).
26. “Como comprador quiero poder editar información de los proveedores.” (P. Tovar, comunicación personal, 17 de septiembre de 2019).
27. “Como comprador quiero activar o desactivar un proveedor para restringir su ingreso al sistema.” (P. Tovar, comunicación personal, 17 de septiembre de 2019).

Después de recopilar y analizar las historias de usuario se las documentó en la siguiente tabla con los campos:

- **Número:** Número único que ayuda a identificar a la historia de usuario.
- **Identificación:** Titulo corto que describe a la historia de usuario.
- **Descripción:** Historia de usuario narrada por el usuario.
- **Story Point:** Valoración de la historia de usuario basada en esfuerzo y tiempo de desarrollo categorizado del 1 al 5 siendo 1 bajo esfuerzo y 5 alto esfuerzo.
- **Rol:** Perfil del usuario autor de la historia de usuario
- **Prioridad:** Prioridad de la historia de usuario a ser desarrollada categorizada del 1 al 5 siendo 1 baja prioridad y 5 Alta prioridad.

**Tabla 1**

*Historias de usuario*

| Número | Identificación | Descripción | Story Point | Rol | Prioridad |
|---|---|---|---|---|---|
| 1 | Registro de proveedores | Como administrador del sistema quiero que los proveedores se puedan registrar desde cualquier lugar en el sistema para que se les haga más fácil ese proceso. | 2 | Proveedor | 5 |
| 2 | Registro de compradores | Como administrador del sistema quiero que los compradores se puedan registrar desde cualquier lugar en el sistema para que se les haga más fácil ese proceso. | 2 | Comprador | 5 |
| 3 | Creación de compradores | Como administrador del sistema quiero crear compradores personalmente. | 2 | Administrador | 4 |
| 4 | Asignar comprador a proveedor | Como administrador del sistema quiero asignar un comprador a cada proveedor registrado en el sistema para que sean atendidos por sus respectivos compradores. | 4 | Administrador | 4 |

| Número | Identificación | Descripción | Story Point | Rol | Prioridad |
|---|---|---|---|---|---|
| 5 | Vista de proveedores para administrador | Como administrador del sistema quiero ver todos los proveedores registrados en el sistema para saber que proveedores tengo y cuantos. | 3 | Administrador | 4 |
| 6 | Vista de compradores para administrador | Como administrador del sistema quiero ver todos los compradores registrados en el sistema para saber que compradores tengo y cuantos. | 3 | Administrador | 4 |
| 7 | Cargar archivo de productos proveedor | Como proveedor quiero subir un archivo en formato CSV con la información de mis productos para enviar a mi comprador asignado y realizar la venta. | 5 | Proveedor | 3 |
| 8 | Editar información de productos | Como proveedor quiero corregir información de un producto del archivo en formato CSV en el sistema para que sea más rápido editar la información de mis productos. | 4 | Proveedor | 3 |
| 9 | Buscar productos | Como proveedor quiero buscar un producto por el código interno del producto o por el nombre del producto o por el código de producto proveedor o por la categoría del producto o por la subcategoría del producto o por la familia del producto. | 2 | Proveedor | 4 |

| Número | Identificación | Descripción | Story Point | Rol | Prioridad |
|---|---|---|---|---|---|
| 10 | Ordenar productos | Como proveedor quiero ordenar los productos para facilitar el análisis de los datos. | 2 | Proveedor | 5 |
| 11 | Ver productos revisados | Como proveedor quiero ver los productos revisados y los productos no revisados. | 3 | Proveedor | 3 |
| 12 | Ver estado de productos revisados | Como proveedor quiero ver el estado de los productos revisados por un comprador para saber si fueron aprobados o rechazados. | 2 | Proveedor | 4 |
| 13 | Ver comentario productos rechazados | Como proveedor quiero ver el comentario de porqué fueron rechazados los productos. | 4 | Proveedor | 4 |
| 14 | Ver horario para citas | Como proveedor quiero ver el horario disponible de mi comprador asignado para agendar una cita. | 4 | Proveedor | 3 |
| 15 | Correo cita cancelada para proveedor | Como proveedor quiero que me llegué un correo cuando un comprador cancela una cita para hablar con él. | 3 | Proveedor | 4 |
| 16 | Agregar citas | Como comprador quiero publicar los días y horas en las que puedo atender a mis proveedores para recibir mercadería. | 4 | Comprador | 2 |

| Número | Identificación | Descripción | Story Point | Rol | Prioridad |
|---|---|---|---|---|---|
| 17 | Correo cita Agendada para comprador | Como comprador quiero que me llegué un correo cuando un proveedor agenda una cita para tomarla en cuenta. | 3 | Comprador | 4 |
| 18 | Correo cita cancelada para comprador | Como comprador quiero que me llegué un correo cuando un proveedor cancela una cita para hablar con él. | 3 | Comprador | 4 |
| 19 | Aceptar o rechazar productos. | Como comprador quiero ver todos los productos enviados de mis proveedores para aceptar o rechazar productos. | 4 | Comprador | 2 |
| 20 | Editar información de productos enviados a compradores. | Como comprador quiero editar información de los productos enviados por mis proveedores para agregar información interna de la empresa. | 4 | Comprador | 2 |
| 21 | Buscar productos enviados comprador | Como comprador quiero buscar un producto por el código interno del producto o por el nombre del producto o por el código de producto proveedor o por la categoría del producto o por la subcategoría del producto o por la familia del producto. | 2 | Comprador | 4 |

| Número | Identificación | Descripción | Story Point | Rol | Prioridad |
|---|---|---|---|---|---|
| 22 | Ordenar productos enviados comprador | Como comprador quiero ordenar los productos para facilitar el análisis de los datos. | 2 | Comprador | 4 |
| 23 | Ver productos revisados y no revisados comprador | Como comprador quiero ver los productos revisados y los productos no revisados. | 4 | Comprador | 2 |
| 24 | Aceptar o rechazar todos los productos | Como comprador quiero poder aprobar todos los productos o rechazar todos los productos para no tener que hacer este proceso por cada producto por revisar. | 3 | Comprador | 2 |
| 25 | Descargar archivo con productos aceptados | Como comprador quiero descargarme un archivo en formato CSV con la información de los productos aprobados enviada por los proveedores más la información interna de la empresa. | 4 | Comprador | 2 |
| 26 | Editar productos enviados | Como comprador quiero poder editar información de los proveedores | 3 | Comprador | 2 |
| 27 | Activar o desactivar proveedores | Como comprador quiero activar o desactivar un proveedor para restringir su ingreso al sistema. | 2 | Comprador | 2 |

### 3.1.2 Product Backlog

El product backlog es la lista principal de las funcionalidades que el sistema deberá cumplir, estas funcionalidades nacen a partir de historias de usuario previamente documentadas. Cada una de las funcionalidades debe ser escrita en lenguaje que sea comprensible tanto para el cliente como para el equipo de desarrollo.

#### 3.1.2.1 Características

Los requerimientos que componen el product backlog debe cumplir con las siguientes características:

- Suficientemente detallado de tal manera que el equipo de desarrollo entienda con exactitud qué es lo que esa funcionalidad debe cumplir.
- Deben ser estimadas, es decir categorizadas en tiempo y esfuerzo que requiera el equipo de desarrollo para resolver esta funcionalidad.
- Variable debido a que las funcionalidades pueden variar o eliminarse dependiendo de las necesidades.
- Priorizadas, es decir, se establece un orden de acuerdo con que funcionalidades deberán ser desarrolladas antes que otras.

Después de analizar las historias de usuario se encontraron varias funcionalidades expresadas en la siguiente tabla con los campos:

- **Número:** Número único que ayuda a identificar un requerimiento.
- **Descripción:** Descripción del requerimiento.
- **Story Point:** Valoración del requerimiento basado en esfuerzo y tiempo de desarrollo categorizado del 1 al 5 siendo 1 bajo esfuerzo y 5 alto esfuerzo.
- **Rol:** Perfil del usuario que utilizara el requerimiento.

- **Prioridad:** Prioridad del requerimiento a ser desarrollado categorizada del 1 al 5 siendo 1 baja prioridad y 5 Alta prioridad.
- **Sprint:** Número de sprint en el cual la funcionalidad será desarrollada.
- **Tiempo:** Estimación en horas para desarrollar esa funcionalidad.
- **Criterio de aceptación:** Descripción del criterio que deberá cumplir el requerimiento para ser aceptado.

***Tabla 2***

*Product Backlog*

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| 1 | Crear una pantalla de registro para los proveedores. | 2 | Proveedor | 5 | 1 | 4 horas | El usuario con el perfil de proveedor podrá registrarse en el sistema desde cualquier dispositivo electrónico con navegador. |
| 2 | Crear una pantalla de registro para los compradores. | 2 | Comprador | 5 | 1 | 4 horas | El usuario con el perfil de comprador podrá registrarse en el sistema desde cualquier dispositivo electrónico con navegador. |
| 3 | Crear un cuadro de dialogo en la pantalla de compradores para que el administrador pueda agregar compradores | 2 | Administrador | 4 | 1 | 4 horas | El usuario con el perfil de administrador podrá crear compradores. |
| 4 | Crear un componente de selección por cada proveedor en donde el administrador pueda seleccionar el comprador que desea asignarle a ese proveedor. | 4 | Administrador | 4 | 1 | 8 horas | El usuario con el perfil de administrador podrá asignarle un comprador a cada proveedor. |

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| 5 | Implementar en la pantalla de proveedores una tabla, con paginación, en la que se encuentren todos los proveedores registrados en el sistema | 3 | Administrador | 4 | 1 | 6 horas | El usuario con el perfil de administrador podrá ver en una tabla todos los proveedores registrados en el sistema. |
| 6 | Implementar en la pantalla de compradores una tabla, con paginación, en la que se encuentren todos los compradores registrados en el sistema. | 3 | Administrador | 4 | 1 | 6 horas | El usuario con el perfil de administrador podrá ver en una tabla todos los compradores registrados en el sistema. |
| 7 | Implementar una barra de búsqueda en la tabla de proveedores que encuentren un proveedor por su nombre, email, ruc, nombre de la empresa o celular. | 2 | Administrador | 4 | 1 | 4 horas | El usuario con el perfil de administrador podrá buscar un proveedor por su nombre, email, ruc, nombre de la empresa o celular |
| 8 | Implementar un ordenamiento en la tabla de proveedores por: nombre, email, ruc, estado, nombre de la empresa y fecha de creación. | 2 | Administrador | 4 | 1 | 4 horas | El usuario con el perfil de administrador podrá ordenar proveedores por su nombre, email, ruc, estado, nombre de la empresa y fecha de creación. |

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| 9 | Implementar una barra de búsqueda en la tabla de compradores que encuentren un comprador por su nombre, email o celular. | 2 | Administrador | 4 | 1 | 4 horas | El usuario con el perfil de administrador podrá buscar un comprador por su nombre, email o celular. |
| 10 | Implementar un ordenamiento en la tabla de compradores por: nombre, email, estado y rol | 2 | Administrador | 4 | 1 | 4 horas | El usuario con el perfil de administrador podrá ordenar compradores por su nombre, email, estado y rol. |
| 11 | Implementar un botón que permita cargar un archivo en formato CSV y que ponga en una tabla la información de los productos encontrados en el archivo. | 3 | Proveedor | 3 | 2 | 6 horas | El usuario con el perfil de proveedor podrá cargar un archivo con información de productos y ver la información en una tabla. |
| 12 | Marcar en rojo los campos de los productos incorrectos o vacíos | 2 | Proveedor | 3 | 2 | 4 horas | El usuario con perfil de proveedor podrá visualizar los campos de los productos que no cumplen con los estándares. |
| 13 | Evitar que el usuario guarde productos para ser revisados que no cumplan con los estándares. | 1 | Proveedor | 3 | 2 | 2 horas | El usuario con perfil de proveedor no podrá guardar productos para ser revisados si no cumplen con los estándares. |

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| 14 | Implementar una barra de búsqueda en la tabla de productos que encuentren un producto por su nombre, código interno, código de producto proveedor, categoría, subcategoría o familia del producto. | 2 | Proveedor | 4 | 2 | 4 horas | El usuario con el perfil de proveedor podrá buscar un producto por su nombre, código interno, código de producto proveedor, categoría, subcategoría o familia del producto. |
| 15 | Implementar un ordenamiento en la tabla de productos por: código interno, código producto del proveedor, código de barras, nombre, categoría subcategoría, familia, rentabilidad, costo con IVA, precio marcado, precio sugerido, unidad por caja, peso por caja, alto caja, ancho caja, largo caja, cajas por paleta, cajas por manto, manto por paleta, volumen, clase de caducidad y estado | 2 | Proveedor | 4 | 2 | 4 horas | El usuario con el perfil de proveedor podrá ordenar productos por su código interno, código producto del proveedor, código de barras, nombre, categoría subcategoría, familia, rentabilidad, costo con IVA, precio marcado, precio sugerido, unidad por caja, peso por caja, alto caja, ancho caja, largo caja, cajas por paleta, cajas por manto, manto por paleta, volumen, clase de caducidad y estado. |
| 16 | Implementar dos pestañas en la pantalla de productos en los cuales se pueda ver una tabla con los productos revisados y productos no revisados | 3 | Proveedor | 3 | 2 | 6 horas | El usuario con perfil de proveedor podrá ver los productos revisados y productos no revisados. |

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| 17 | Agregar un campo con nombre estado a cada producto revisado para saber si el producto fue aceptado o rechazado | 2 | Proveedor | 4 | 2 | 4 horas | El usuario con perfil de proveedor podrá ver que productos fueron aceptados y que productos fueron rechazados por su comprador asignado. |
| 18 | Implementar un dialogo en el cual se pueda ver el comentario de un producto rechazado | 2 | Proveedor | 4 | 2 | 4 horas | El usuario con perfil de proveedor podrá ver la razón por la que un producto fue rechazado |
| 19 | Implementar un calendario en el cual se pueda ver las citas disponibles del comprador asignado a un proveedor | 4 | Proveedor | 3 | 2 | 8 horas | El usuario con perfil de proveedor podrá visualizar en un calendario las citas disponibles de su comprador asignado. |
| 20 | Enviar un correo electrónico al proveedor cuando su comprador asignado cancele una cita ya agendada. | 3 | Proveedor | 4 | 2 | 6 horas | El usuario con perfil de proveedor recibirá un correo electrónico cuando un comprador cancele una cita agendada. |
| 21 | En la pantalla de citas de un comprador, implementar un | 3 | Comprador | 2 | 3 | 4 horas | El usuario con perfil de comprador podrá agregar una cita. |

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| | dialogo que permita agregar una cita | | | | | | |
| 22 | Agregar citas desde las 10 de la mañana hasta las 5 de la tarda a todos los compradores con una duración de 1 hora por cada cita, exceptuando la hora de almuerzo de 1 de la tarde a 2 de la tarde. | 2 | Comprador | 2 | 3 | 4 horas | Los compradores por defecto tendrán citas creadas de lunes a viernes desde las 10 am hasta las 5 pm con tiempo de duración de una hora. |
| 23 | Enviar un correo electrónico al comprador cuando un proveedor agende una cita. | 3 | Comprador | 4 | 3 | 4 horas | El usuario con perfil de comprador recibirá un correo electrónico cuando un proveedor agende una cita. |
| 24 | Enviar un correo electrónico al comprador cuando un proveedor cancele una cita. | 3 | Comprador | 4 | 3 | 4 horas | El usuario con perfil de comprador recibirá un correo electrónico cuando un proveedor cancele una cita. |
| 25 | Implementar una tabla en la cual se pueda ver los productos enviados por un proveedor con dos iconos al final de cada producto que permitan aprobar o rechazar el producto | 4 | Comprador | 2 | 3 | 6 horas | El usuario con perfil de Comprador podrá aprobar o rechazar productos. |

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| 26 | Implementar, en la misma tabla de productos, entradas de texto donde su pueda editar de manera similar a una tabla de Excel, la información de los productos | 4 | Comprador | 2 | 3 | 6 horas | El usuario con perfil de comprador podrá editar en la tabla la información de los productos enviados por el proveedor. |
| 27 | Implementar una barra de búsqueda en la tabla de productos que permita buscar un producto por el código interno del producto o por el nombre del producto o por el código de producto proveedor o por la categoría del producto o por la subcategoría del producto o por la familia del producto. | 2 | Comprador | 4 | 3 | 4 horas | El usuario con perfil de comprador podrá buscar un producto por el código interno del producto o por el nombre del producto o por el código de producto proveedor o por la categoría del producto o por la subcategoría del producto o por la familia del producto. |
| 28 | Implementar un ordenamiento en la tabla de productos por: código interno, código producto del proveedor, código de barras, nombre, categoría subcategoría, familia, rentabilidad, costo con IVA, precio marcado, precio sugerido, unidad por caja, peso | 2 | Comprador | 4 | 3 | 4 horas | El usuario con el perfil de compradores podrá ordenar productos por su código interno, código producto del proveedor, código de barras, nombre, categoría subcategoría, familia, rentabilidad, costo con IVA, precio marcado, precio sugerido, unidad |

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| | por caja, alto caja, ancho caja, largo caja, cajas por paleta, cajas por manto, manto por paleta, volumen, clase de caducidad y estado | | | | | | por caja, peso por caja, alto caja, ancho caja, largo caja, cajas por paleta, cajas por manto, manto por paleta, volumen, clase de caducidad y estado. |
| 29 | Implementar dos pestañas en la pantalla de productos en los cuales se pueda ver una tabla con los productos revisados y productos no revisados | 4 | Comprador | 2 | 3 | 6 horas | El usuario con perfil comprador podrá ver los productos revisados y productos no revisados. |
| 30 | Implementar un botón que permita aprobar todos los productos y otro botón que permita rechazar todos los productos. | 3 | Comprador | 2 | 3 | 6 horas | El usuario con perfil de comprador podrá aprobar todos los productos o rechazar todos los productos. |
| 31 | Implementar un botón que permita descargarse los productos aprobados en un archivo en formato CSV con la información de los productos enviada por los proveedores más la información interna de la empresa. | 4 | Comprador | 2 | 3 | 5 horas | El usuario con perfil comprador podrá descargarse un archivo en formato CSV con la información de los productos aprobados. |

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| 32 | Implementar un icono para editar en la pantalla de proveedores que despliegue un dialogo que permita editar la información de un proveedor | 3 | Comprador | 2 | 3 | 6 horas | El usuario con perfil de comprador podrá editar la información de los proveedores que tiene asignado. |
| 33 | Implementare un componente de selección donde el comprador pueda activar o desactivar a un proveedor. | 2 | Comprador | 2 | 3 | 2 horas | El usuario con perfil de comprador podrá activar o desactivar a un proveedor. |

Después de documentar cada requerimiento en el product backlog podemos estimar que el tiempo de desarrollo del sistema, para que este sea funcional, tomará 157 horas las cuales se las dividirá en 3 Sprint dependiendo su prioridad comenzando con las tareas con prioridad más alta hasta finalizar con las tareas de menor prioridad como se especifica en el product backlog.

### 3.1.3 Descripción del Sistema

#### 3.1.3.1 Descripción General

De acuerdo con las necesidades y funcionalidades que deberá cumplir este sistema, se decidió categorizar a los usuarios en 3 perfiles con diferentes privilegios y funciones. Dependiendo al rol que desempeña cada usuario para la empresa Asertia:

- **Administrador**: El usuario con el perfil de administrador es el encargado en asignar un usuario con el perfil de comprador a cada usuario con el perfil de proveedor para que el comprador sea el responsable de atender a un proveedor. Además, el administrador puede gestionar a todos los proveedores y compradores registrados en el sistema.
- **Proveedor**: El usuario con perfil de proveedor es el encargado en subir archivos en formato CSV con información de productos que desean vender a la empresa Asertia, para que esta información sea revisada y analizada por el comprador que tiene asignado. Además, podrá agendar citas publicadas por su comprador asignado.
- **Comprador**: El usuario con el perfil de comprador es el encargado en revisar y editar la información de los productos enviados por los diferentes proveedores que tiene asignado y es el responsable en aprobar o rechazar productos para que estos entren a un proceso de compra por la empresa. Además, puede publicar citas dependiendo su disponibilidad de tiempo para reunirse con sus proveedores.

#### 3.1.3.2 Funcionalidades Generales del Sistema

Al finalizar el desarrollo de este producto los miembros del departamento de compras y proveedores de la empresa Asertia contarán con un sistema web el cual será accesible desde cualquier dispositivo electrónico con acceso a internet y a un navegador. El cual está enfocado para mejorar la atención y comunicación entre los proveedores y el departamento de compras de Asertia, permitiendo a los proveedores enviar de una manera más sencilla información de sus productos y al mismo tiempo minimizando los errores de formato en la información de sus productos, facilitando el análisis de la información al departamento de compras y optimizando el tiempo en determinar que productos serán aprobados para que ingresen al proceso de compra.

El sistema también brinda la facilidad a los proveedores de agendar citas a través de un calendario con información de la disponibilidad de tiempo del personal del departamento de compras de Asertia, permitiéndoles realizar una mejor planificación de las actividades diarias tanto a los proveedores como al personal de compras.

#### 3.1.3.3 Funciones Enfocadas al Administrador

El sistema le presenta dos pestañas por las que el usuario con perfil de administrador puede navegar:

**Compradores:** El usuario visualizará una tabla con la información del personal de compras que se ha registrado en el sistema, la cual podrá editar según considere oportuno. Además, en esta pantalla tendrá la facilidad de crear compradores.

**Proveedores:** En esta pantalla el usuario visualizara en una tabla la información de todos los proveedores registrados en el sistema, la cual puede ser editada. El sistema también permitirá al usuario visualizar los productos enviados y revisados por cada proveedor.

#### 3.1.3.4 Funciones Enfocadas al Proveedor

El sistema le presenta dos pestañas por las que el usuario con perfil de proveedor puede navegar:

**Productos:** En esta pantalla el sistema le permitirá visualizar al usuario los productos que ya fueron revisados por un comprador, en donde podrá leer los comentarios de los productos que fueron rechazados. También podrá visualizar los productos que fueron enviados, pero todavía no son revisados los cuales podrán ser editados. Además, el usuario podrá subir, en cualquier momento, un archivo en formato CSV con los datos de los productos que desee enviar a un comprador. El sistema indicará al usuario los productos que no cumplan con los estándares de Asertia y no le permitirá enviar los productos hasta que estos no sean modificados y cumplan con los estándares establecidos.

**Citas:** En esta pantalla el sistema presentará un calendario con las citas disponibles que el usuario puede agendar para reunirse con un agente de compras. También le permitirá agregar una descripción a la cita para que tengan más información sobre el motivo de la reunión.

#### 3.1.3.5 Funciones Enfocadas al Comprador

El sistema le presenta una pestaña dependiendo de los permisos que el usuario con el perfil de comprador posea esta pestaña puede ser:

**Proveedores:** El usuario visualizará una tabla con la información de los proveedores que le fueron asignados, esta información puede ser modificada por el usuario. El sistema le permitirá revisar los productos enviados por cada proveedor los cuales podrá aprobar o rechazar. Finalmente, el sistema le permitirá cargar un archivo en formato CSV con información de productos al usuario con perfil de comprador. Esta pestaña se le presentará a los compradores que no sean catalogados como administradores de bodega.

**Citas:** En esta pantalla el sistema presentará un calendario con las citas disponibles y también las citas que ya han sido agendadas por alguno de sus proveedores. Esta pestaña se le presentará a los compradores que sean catalogados como administradores de bodega.

## 3.2 DISEÑO DE LA APLICACIÓN TANTO FRONTEND (SOFTWARE QUE INTERACTÚA CON EL USUARIO) Y BACKEND (MANEJO DE LA LÓGICA DEL NEGOCIO E INTERACCIÓN CON LA BASE DE DATOS).

En esta sección primeramente definiremos la arquitectura que tendrá el sistema web, así como la forma en la que se transferirán los datos entre el backend y el frontend Posteriormente definiremos el diseño de la aplicación del backend y finalmente definiremos también el diseño de la aplicación del frontend

### 3.2.1 Arquitectura del Sistema

La arquitectura bajo la cual se construirá este sistema web será de tres capas principales las cuales serán: la capa del modelo de datos, la capa de la lógica del negocio y la capa de la vista o interfaz de usuario. Esta arquitectura se la conoce como modelo vista controlador (MVC).

Para llevar a cabo el sistema web se desarrollarán dos aplicaciones con la finalidad de dividir las capas previamente mencionadas y tener una mejor estructura y organización de las capas y funcionalidades del sistema. Como se mencionó en capítulos anteriores, las aplicaciones que conformarán este sistema web serán el backend y el frontend.

La forma en la cual estas dos aplicaciones se comunicarán será mediante API´s también conocidos como servicios web los cuales utilizarán el protocolo REST ya que mediante este protocolo la transferencia de los datos es en formato JSON lo cual facilita la comunicación entre estas dos aplicaciones ya que ambas utilizan el mismo formato para manejar los datos.

### 3.2.2 Backend

La aplicación del backend será la encargada en manejar la capa del modelo de datos y la capa de la lógica del negocio. Además, en esta aplicación se añadirá una capa de servicios web los cuales serán la conexión ente el backend y el frontend.

#### 3.2.2.1 Modelo de Datos

En esta capa se realizará la conexión a la base de datos, así como la gestión de los datos que utilizará el sistema. La función principal de esta capa es recuperar la información de la base de datos y modelar los datos. También, se establecerán reglas que necesita la base de datos para mantener datos consistentes.

#### 3.2.2.2 Lógica del negocio

La capa de la lógica del negocio o controlador es la encargada resolver y solucionar los requerimientos del sistema, mediante funciones las cuales realizan una acción y devuelven información la cual será expuesta gracias a los servicios web. Otra funcionalidad de esta capa es la de conectarse con la capa del modelo de datos con la finalidad de insertar, modificar y obtener datos.

#### 3.2.2.3 Servicios Web

Esta capa se encarga principalmente de exponer o publicar servicios mediante el protocolo REST que se basa en la transferencia de datos en formato JSON. Estos servicios web serán accesible mediante una URL única y utilizando los métodos:

GET: utilizado para obtener un recurso del servidor.

POST: utilizado para insertar un recurso en el servidor

PUT: utilizado para actualizar o modificar un recurso en el servidor.

DELETE: utilizado para eliminar un recurso en el servidor.

### 3.2.3 Frontend

La aplicación del frontend está dedicada principalmente pata desarrollar la interfaz gráfica del sistema con la cual interactúa el usuario, es decir, la lógica que requiere la interfaz usuario para ser amigable y que el usuario tenga una experiencia agradable. Otra función importante de esta aplicación es conectarse al backend por medio de las API's publicadas, con las cuales obtendrán los datos que serán presentados al usuario a través de las acciones que realice el usuario.

El diseño de esta aplicación se basa principalmente en los cuatro elementos que propone la arquitectura Redux para realizar el flujo de datos en el frontend, los cuales son:

#### 3.2.3.1 Store

El store es la única fuente de datos verdadera de la cual la interfaz de usuario se nutrirá de información y estará conformada por las cinco entidades encontradas de acuerdo con los requerimientos del sistema las cuales son: Usuarios, Proveedores, Compradores, Productos y Citas, de esta manera tendremos la información estructurada por entidades lo cual será de mucha utilidad al momento de desarrollar la interfaz de usuario.

#### 3.2.3.2 Reductor

El reductor son funciones las cuales son llamadas por las acciones y son las únicas que podrán modificar el Store. Al igual que el store, el reductor estará dividido en las 5 entidades encontradas en el sistema con la finalidad de mantener un orden en el código fuente del programa, haciéndolo mucho más entendible y legible para cualquier otro programador.

#### 3.2.3.3 Acciones

Las acciones son invocadas por la interacción del usuario con el sistema y su función principal es consumir los servicios web publicados en el backend. Estas acciones son las encargadas de comunicarse con el reductor. De igual manera las acciones estarán divididas

por las entidades del sistema, manteniendo el orden y estructura del flujo de información en el frontend

#### 3.2.3.4 Vistas

Las vistas o interfaz de usuario no son más que un conjunto de componentes de React, las cuales presentan los datos encontrados en el store de la aplicación e interactuará con el usuario a través de las acciones que este realice.

### 3.2.4 Diseño Gráfico del Sistema

Para diseñar el sistema de una manera gráfica y que el equipo de desarrollo tenga una idea más clara de los requerimientos establecidos en el product backlog, se decidió realizar un prototipo de la interfaz gráfica y las funcionalidades del sistema web previo al desarrollo con la finalidad de socializar este prototipo con el cliente para que tenga una idea más clara del producto final.

#### 3.2.4.1 Prototipo del sistema

El prototipo de esta aplicación se realizó con la ayuda de la aplicación Lunacy 5.0 el cual es un programa gratuito de diseño gráfico que posee varias herramientas que son de gran utilidad para diseñar un prototipo de manera fácil y rápida.

De acuerdo con los requerimientos y funcionalidades del sistema, se realizaron tres prototipos con comportamiento y funcionalidades diferentes dependiendo el rol del usuario que lo va a usar:

##### 3.2.4.1.1 Administrador:

El administrador podrá ingresar al sistema utilizando su correo electrónico y su contraseña mediante la siguiente pantalla:




*Ilustración 1 Prototipo perfil de administrador. Pantalla para entrar al sistema*

Al ingresar al sistema se le presentará dos pestañas en el lado izquierdo correspondientes a compradores y proveedores por las cuales podrá navegar libremente. En la pestaña de compradores podrá visualizar una tabla con la información de los compradores registrados en el sistema. En la parte superior de esta tabla tendrá un botón para agregar compradores y una barra de búsqueda para que pueda buscar un comprador de manera rápida.

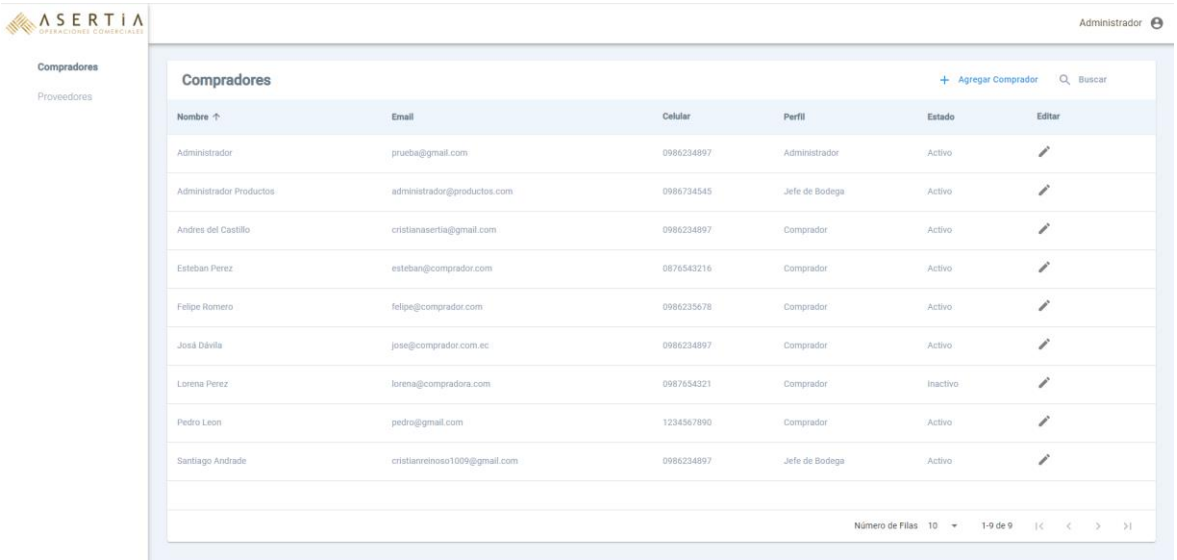



*Ilustración 2 Prototipo perfil de administrador. Pantalla de compradores*

Al presionar el botón para agregar un comprador ubicado en la parte superior derecha de la tabla, se presentará un dialogo para que el usuario complete los datos del comprador que desea agregar

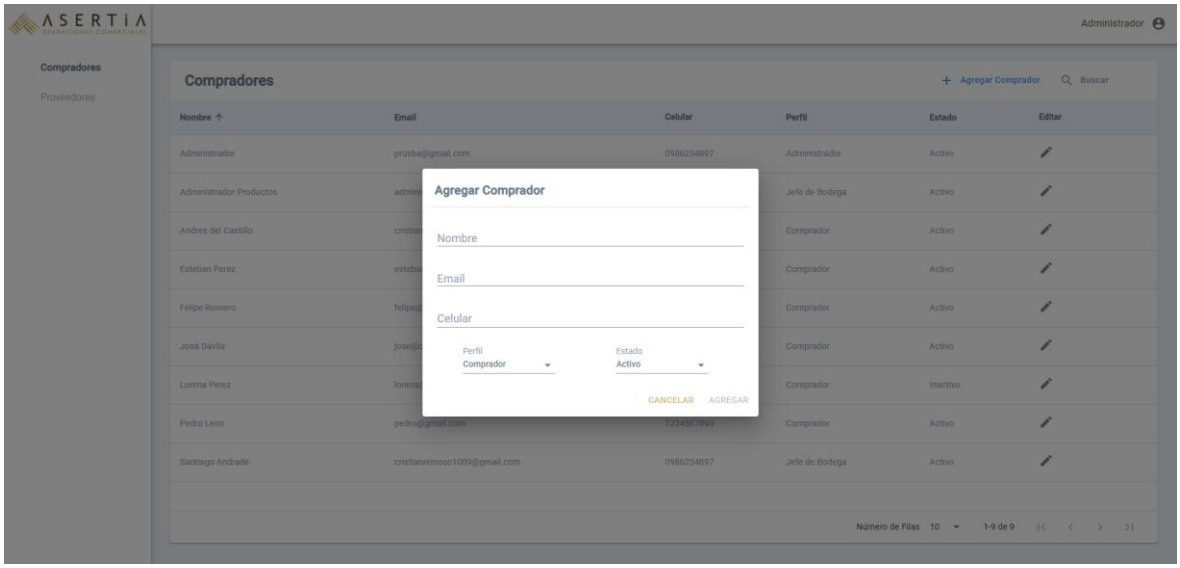



*Ilustración 3 Prototipo perfil administrador. Pantalla para agregar un comprador*

El usuario también será capaz de editar la información de un comprador en cualquier momento presionando el icono de "editar" ubicado en cada fila de la tabla, desplegando un dialogo con la información del comprador seleccionado.

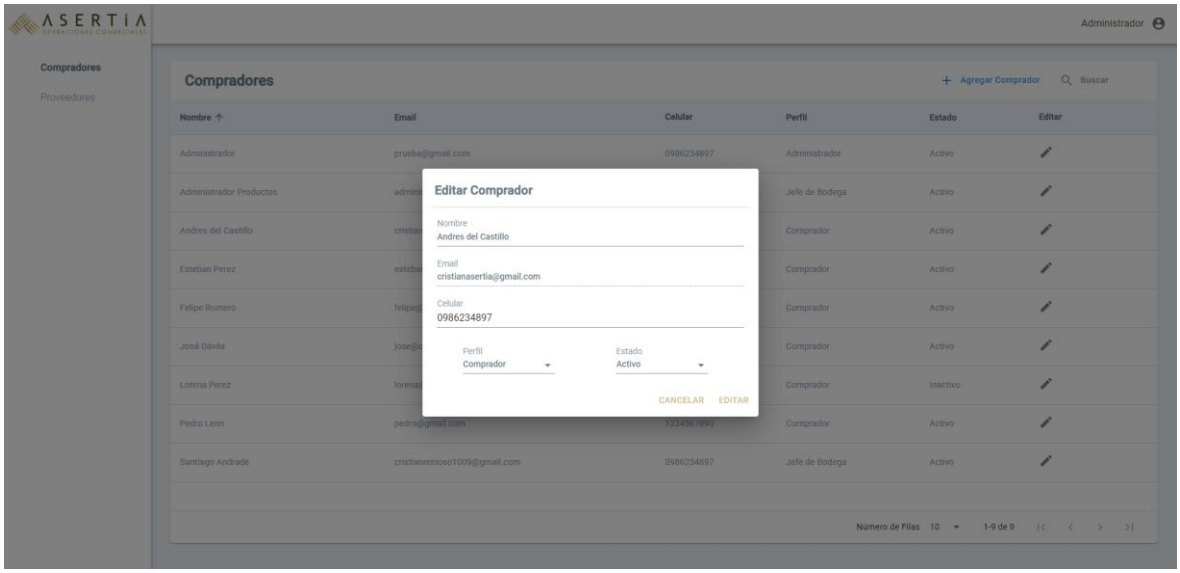



*Ilustración 4 Prototipo perfil administrador. Pantalla para editar un comprador*

En la pestaña de proveedores el usuario visualizará una tabla con la información de todos los proveedores registrados, en la parte superior derecha de la tabla tendrá una barra de búsqueda similar a la tabla presentada anteriormente con la final de buscar un proveedor rápidamente.

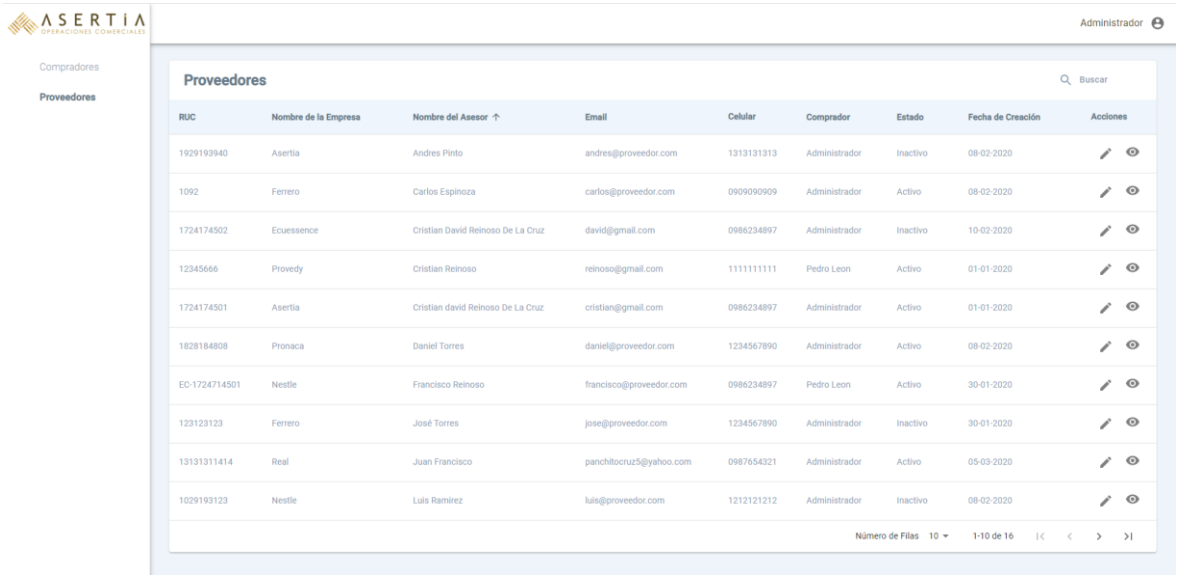



*Ilustración 5 Prototipo perfil de administrador. Pantalla de proveedores*

En esta tabla el usuario también podrá editar la información de los proveedores presionando el icono de “editar” ubicado en la parte derecha de cada fila lo cual presentará un dialogo con la información del proveedor seleccionado para que pueda ser editada.

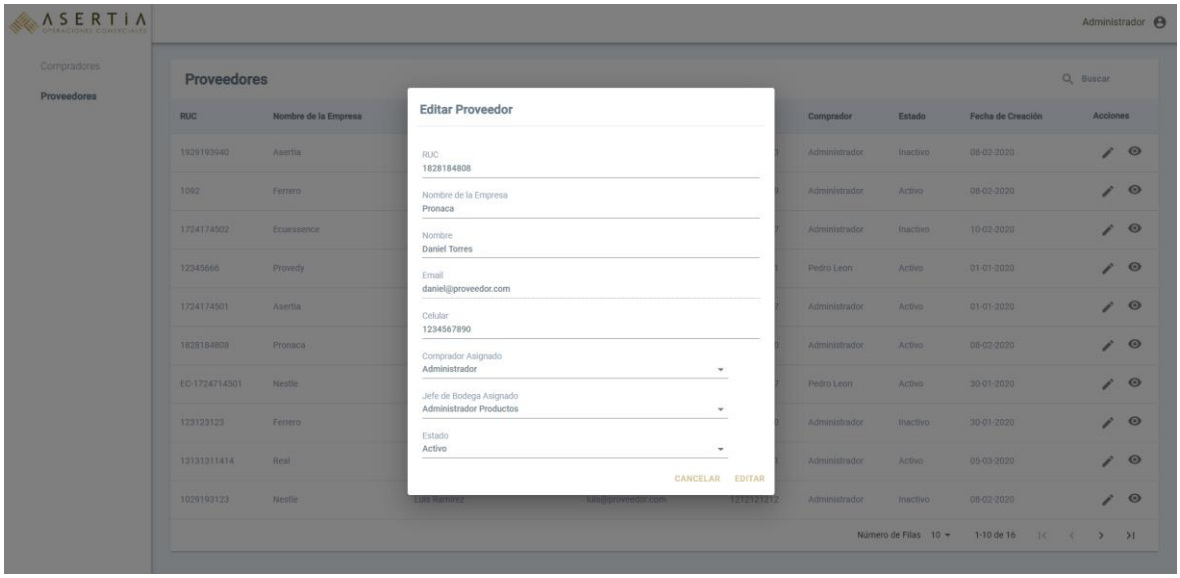



*Ilustración 6 Prototipo perfil de administrador. Pantalla para editar un proveedor*

El administrador podrá revisar los productos de cada proveedor presionando en el icono "ver" ubicado en la parte derecha de cada fila, junto al icono de "editar". Los productos serán presentados en una tabla y están divididos en productos nuevos y productos revisados.

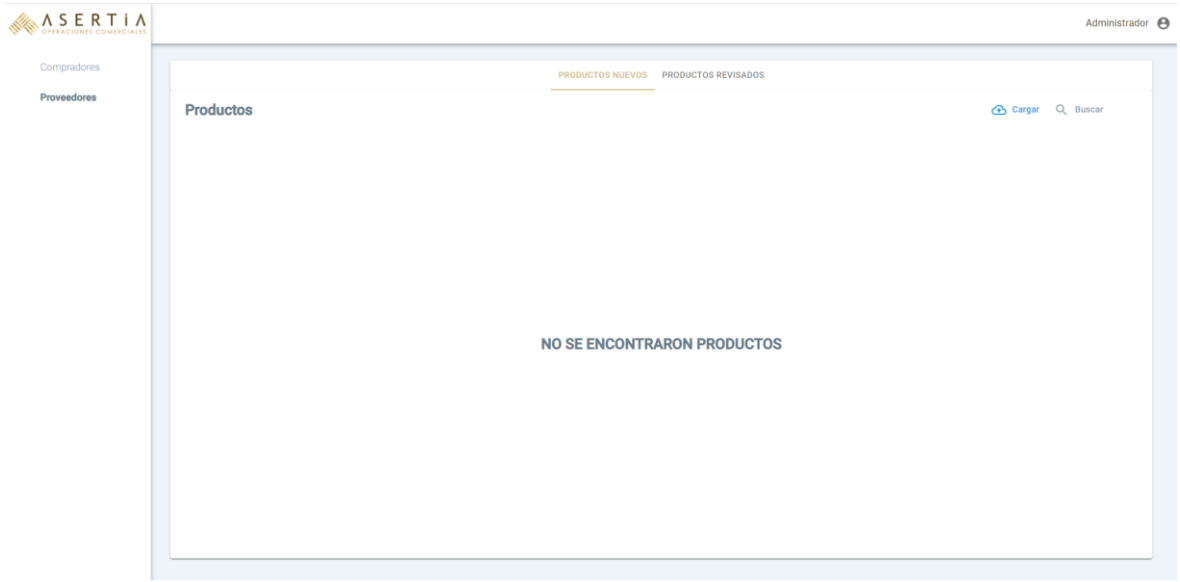



*Ilustración 7 Prototipo perfil de administrador. Pantalla para ver productos nuevos*

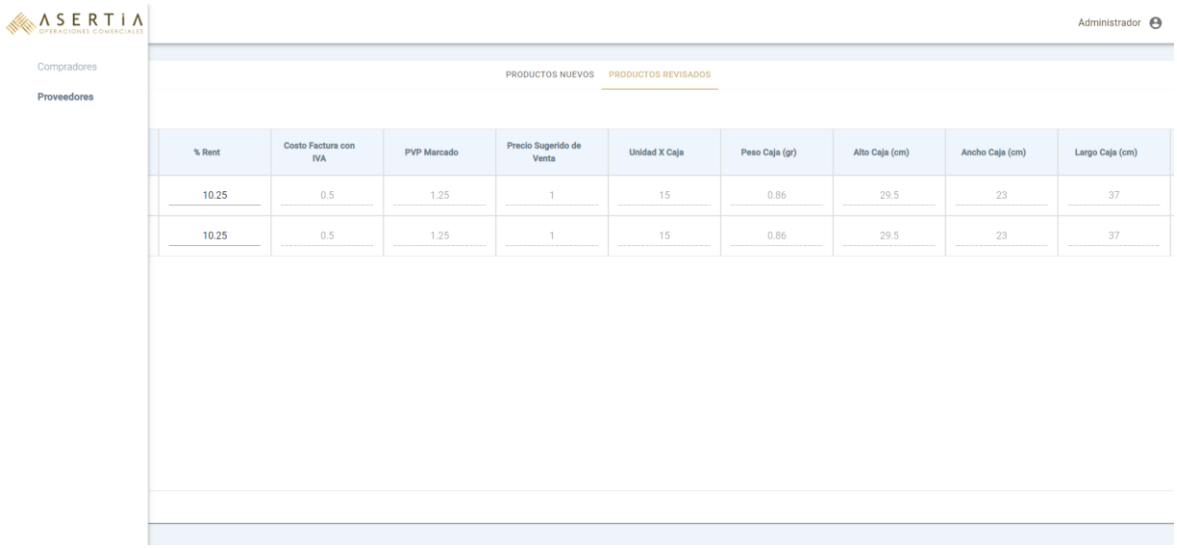



*Ilustración 8 Prototipo perfil de administrador. Pantalla para ver productos revisados*

#### 3.2.4.1.2 Comprador:

El usuario con perfil de comprador podrá registrarse en el sistema ingresando sus datos en la siguiente pantalla. Posteriormente se le enviará un correo electrónico para verificar que sea el propietario del correo electrónico ingresado.




*Ilustración 9 Prototipo perfil de comprador. Pantalla para registrarse*

En el correo electrónico podrá encontrar un botón que lo redirigirá a la siguiente pantalla en donde podrá completar su perfil ingresando la clave que utilizará para ingresar al sistema

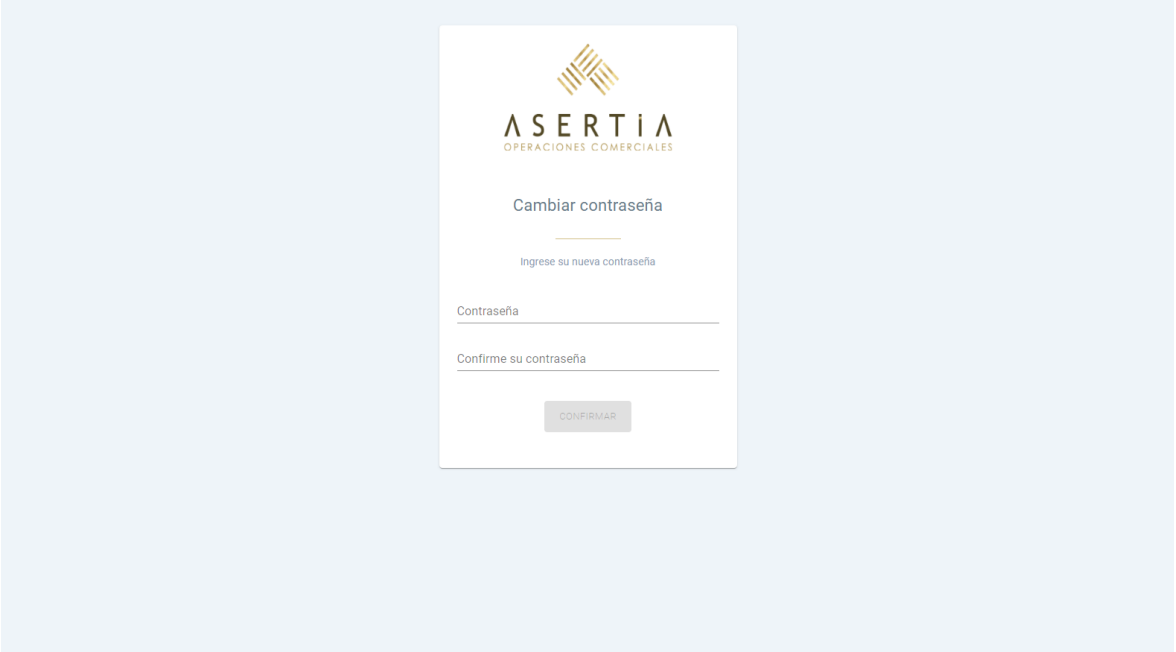



*Ilustración 10 Prototipo perfil de comprador. Pantalla para cambiar contraseña*

Finalizado el proceso de registro y verificación, el usuario con perfil de comprador podrá ingresar al sistema utilizando su correo electrónico y su contraseña mediante la siguiente pantalla:

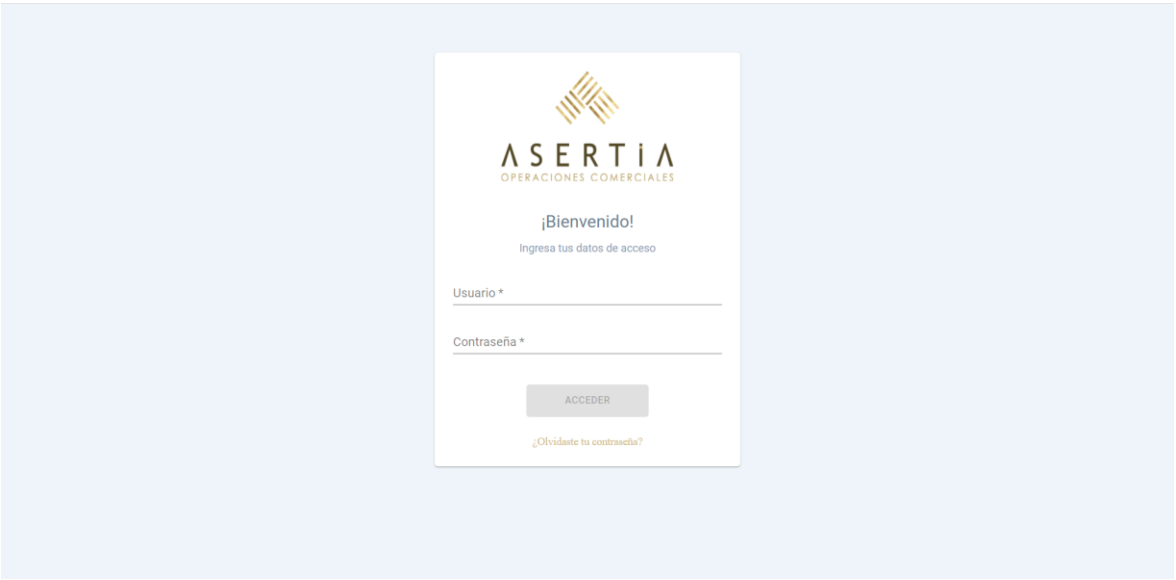



*Ilustración 11 Prototipo perfil de comprador. Pantalla para entrar al sistema*

Al ingresar al sistema se le presentará una pestaña, dependiendo los permisos con los que cuenta, la cual puede ser:

**Proveedores:** El usuario visualizará en una tabla con la información de los proveedores que le fueron asignados.

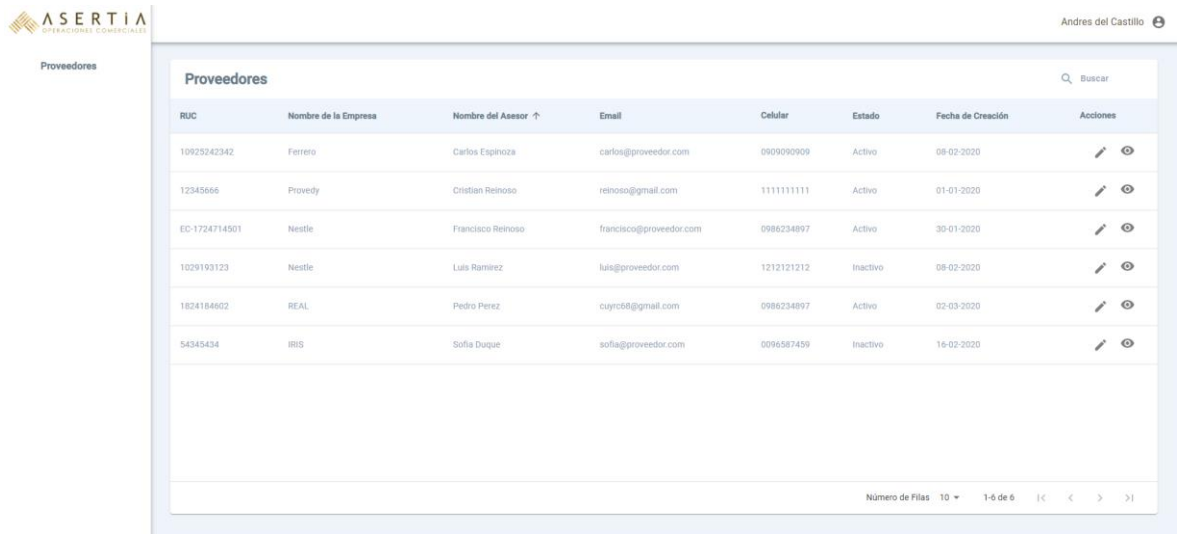



*Ilustración 12 Prototipo perfil de comprador. Pantalla para ver proveedores*

El usuario también estará en capacidad de editar la información de los proveedores presionando el icono de “editar” ubicado en la parte derecha de cada fila lo cual presentará un dialogo con la información del proveedor seleccionado para que pueda ser editada.

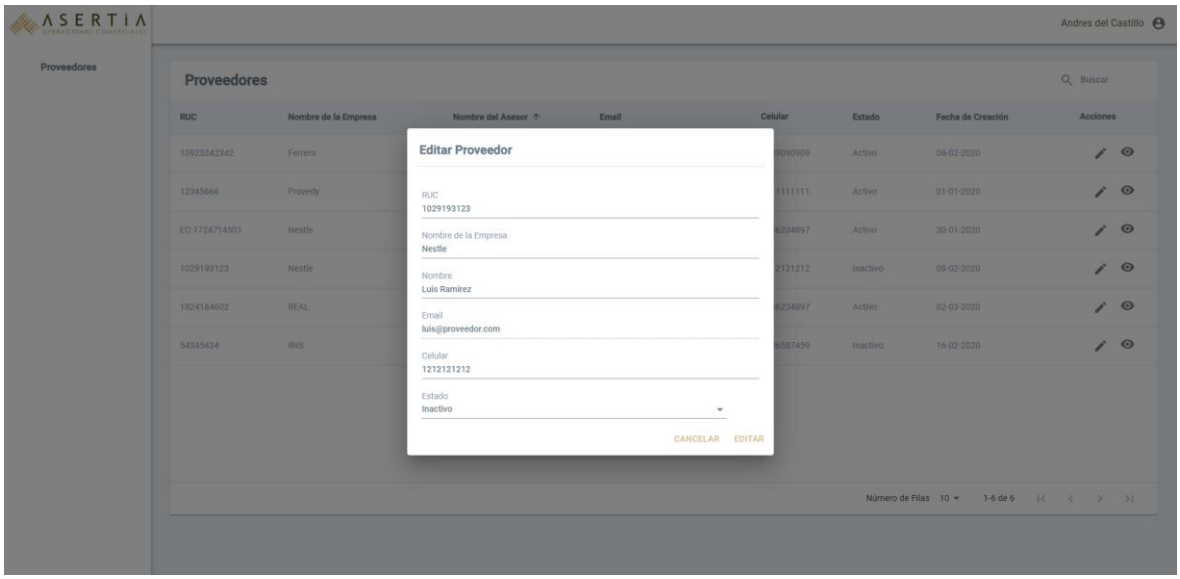



*Ilustración 13 Prototipo perfil de comprador. Pantalla para editar un proveedor*

El Comprador podrá revisar los productos de cada proveedor presionando en el icono “ver” ubicado en la parte derecha de cada fila, junto al icono de “editar”. Los productos serán presentados en una tabla y están divididos en productos nuevos y productos revisados.

**Productos Nuevos:** Se consideran productos nuevos a aquellos productos que han sido cargados por un proveedor, pero no han sido revisados por el comprador.




*Ilustración 14 Prototipo perfil de comprador. Pantalla para ver productos nuevos*

**Productos Revisados:** Se consideran productos revisados a aquellos productos que han sido aprobados o rechazados por un comprador.

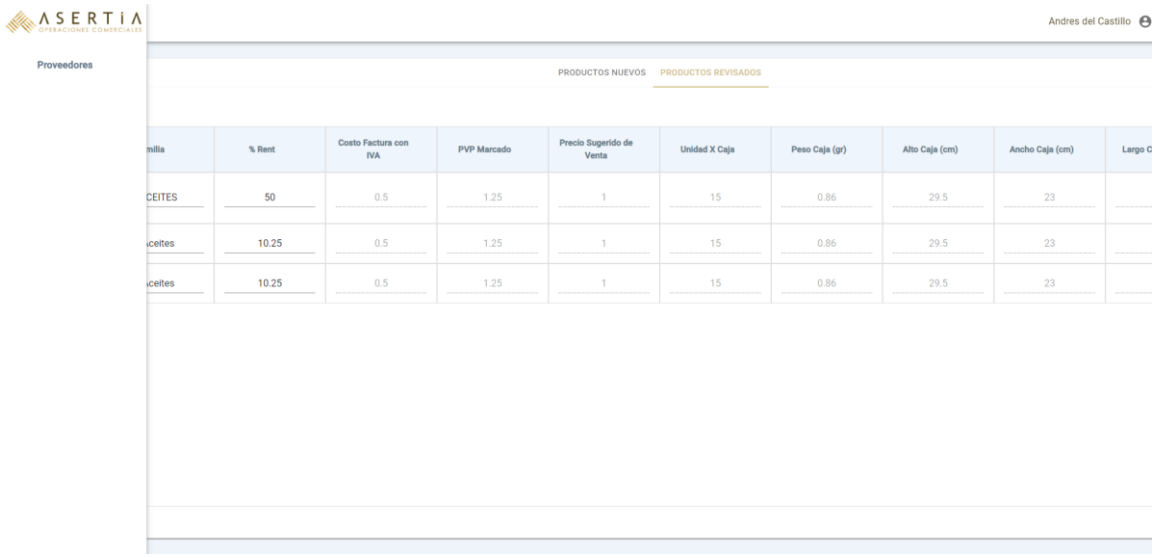


*Ilustración 15 Prototipo perfil de comprador. Pantalla para ver productos revisados*

En la parte superior derecha de la tabla estarán ubicados los botones para: cargar nuevos productos, descargar los productos aprobados, actualizar la información de los productos, aceptar todos los productos, rechazar todos los productos y una barra de búsqueda.

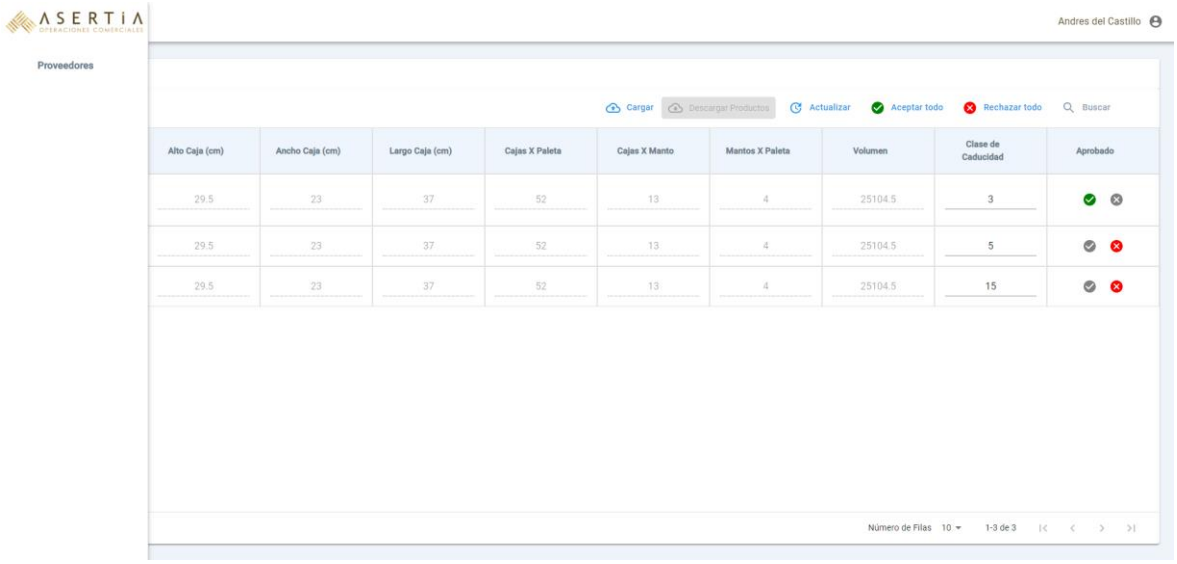



*Ilustración 16 Prototipo perfil de comprador. Tabla de productos*

Al final de cada fila correspondiente a un producto se encontrarán dos iconos con los que el comprador aceptará o rechazará productos. En el caso de que el comprador decida rechazar un producto o todos los productos se le presentará un cuadro de dialogo para que escriba el motivo por el que fue rechazado el o los productos, de esta manera el proveedor tendrá una mejor retroalimentación.

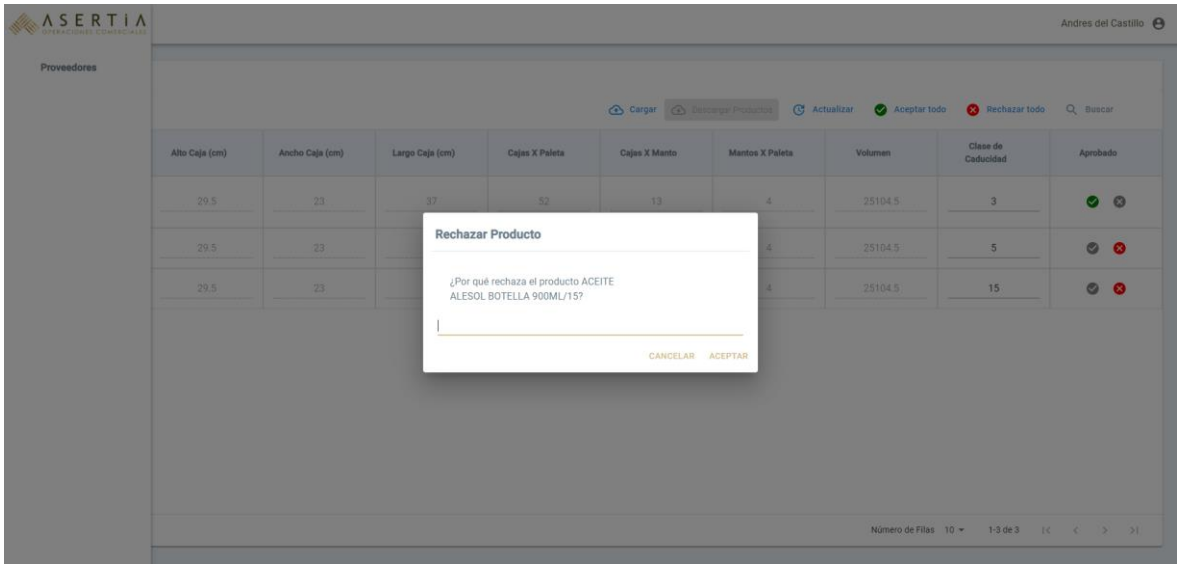



*Ilustración 17 Prototipo perfil de comprador. Pantalla para ver agregar un comentario a productos rechazados*

**Citas:** El sistema presentará un calendario donde se encontrará la información de las citas publicadas y las citas agendadas por algún proveedor. Las citas disponibles se las representará de color verde mientras que las citas agendadas se las representará de color gris

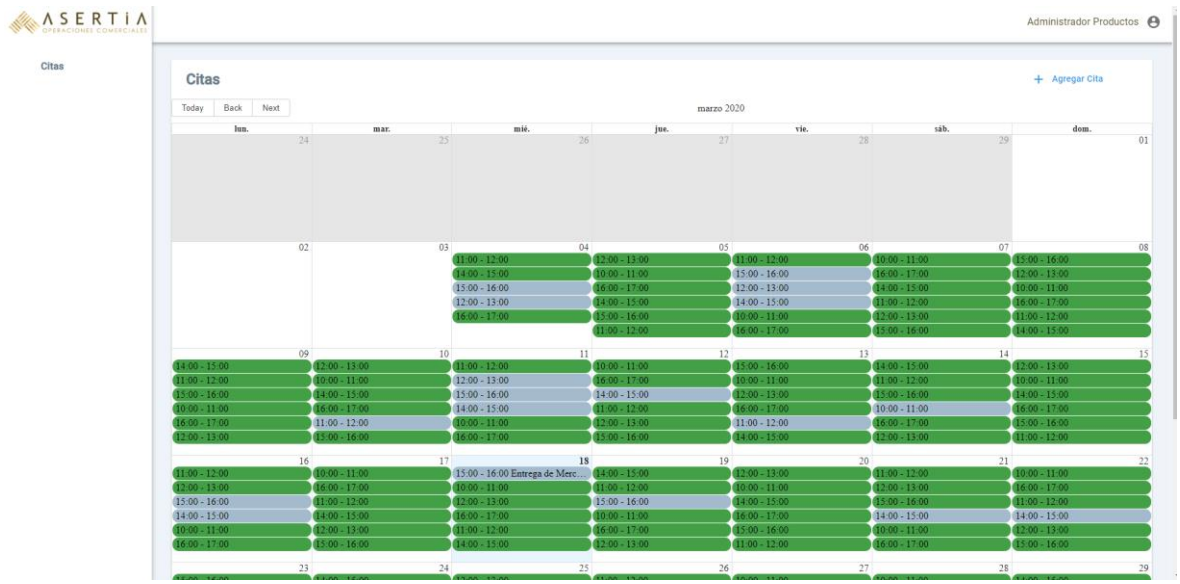



*Ilustración 18 Prototipo perfil de comprador. Pantalla para ver citas*

El usuario será de capaz de agregar citas nuevas presionando el botón ubicado en la parte superior derecha el cual le presentará un cuadro de dialogo al usuario con los datos de la cita para que sean completados.

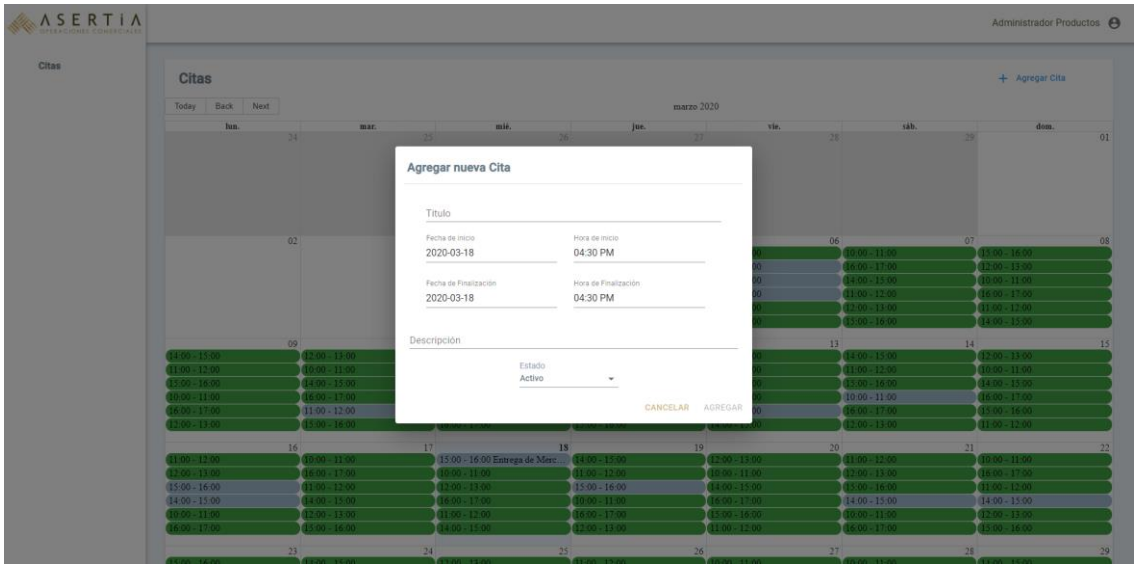



*Ilustración 19 Prototipo perfil de comprador. Pantalla para agregar una cita*

De igual manera el usuario puede modificar los datos de citas publicadas presionando en la cita que desea modificar. Siempre y cuando la cita no haya sido reservada por algún proveedor a no ser que el usuario con perfil de comprador desee cancelar la cita.

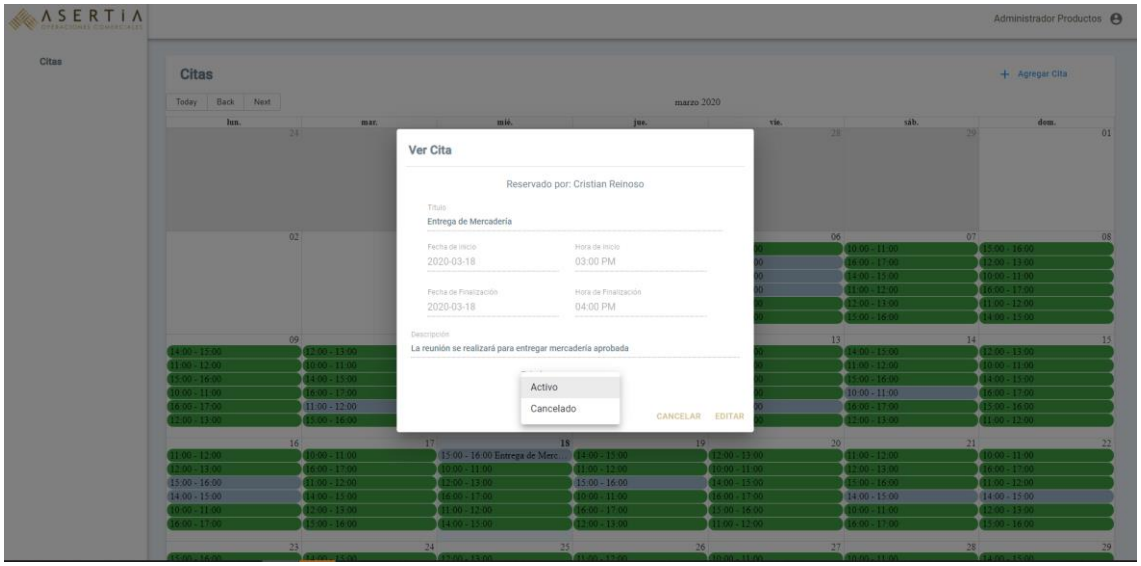



*Ilustración 20 Prototipo perfil de comprador. Pantalla para editar una cita*

#### 3.2.4.1.3 Proveedor:

El usuario con perfil de proveedor podrá registrarse en el sistema ingresando sus datos en la siguiente pantalla. Posteriormente se le enviará un correo electrónico para verificar que sea el propietario del correo electrónico ingresado.

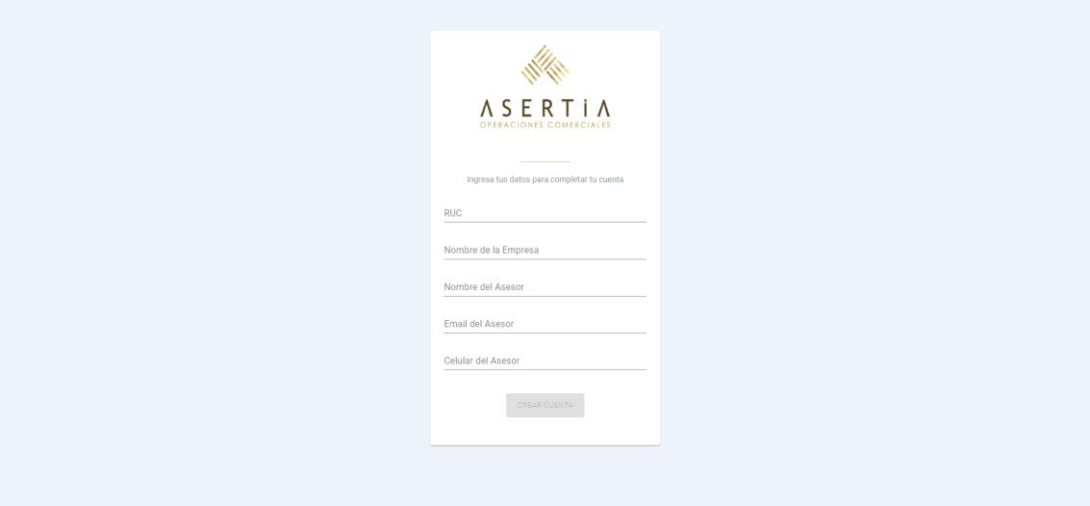



*Ilustración 21 Prototipo perfil de proveedor. Pantalla para registrarse en el sistema*

En el correo electrónico podrá encontrar un botón que lo redirigirá a la siguiente pantalla en donde podrá completar su perfil ingresando la clave que utilizará para ingresar al sistema

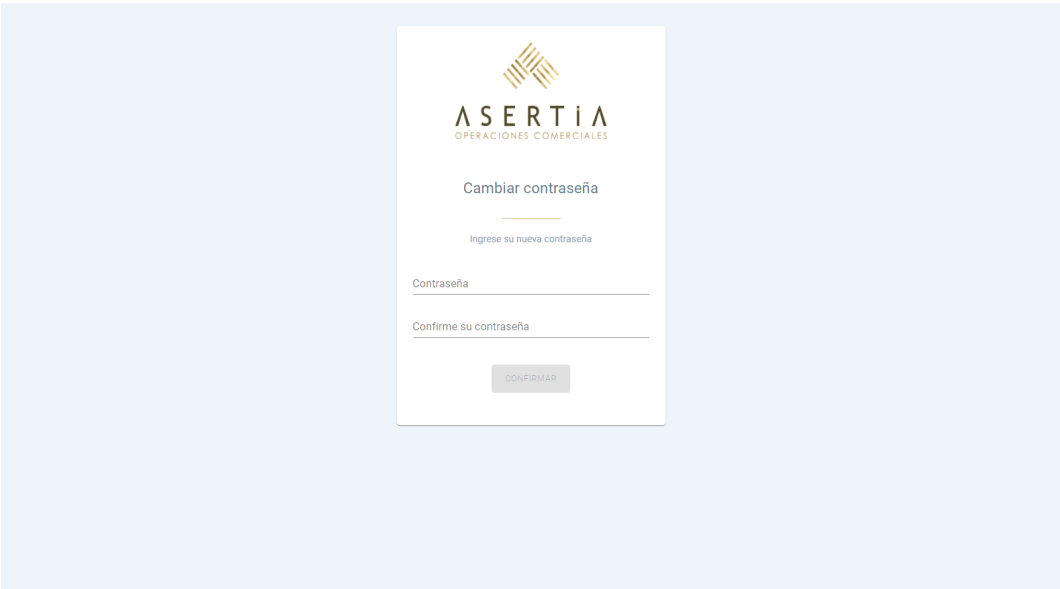



*Ilustración 22 Prototipo perfil de proveedor. Pantalla para cambiar la contraseña*

Finalizado el proceso de registro y verificación, el usuario con perfil de comprador podrá ingresar al sistema utilizando su correo electrónico y su contraseña mediante la siguiente pantalla:




*Ilustración 23 Prototipo perfil de proveedor. Pantalla para ingresar al sistema*

Al ingresar al sistema se le presentará dos pestañas en el lado izquierdo correspondientes a productos y citas por las cuales podrá navegar libremente. Además, podrá visualizar los datos de su comprador asignado, quien es la persona responsable de aprobar o rechazar los productos enviados por el proveedor, y de su bodeguero asignado quien es el encargado de reunirse con el proveedor en las citas reservadas.

**Productos:** En la pestaña de productos podrá visualizar dos pestañas correspondientes a productos nuevos y productos revisados.

**Productos Nuevos:** Se consideran productos nuevos a aquellos productos que han sido cargados, pero no han sido revisados por el comprador, es decir, no han sido aceptados ni rechazados por un comprador. En esta pantalla el usuario podrá cargar nuevos productos presionando en el botón de “cargar” ubicado en la parte superior derecha y seleccionando el archivo en formato CSV.




*Ilustración 24  Prototipo perfil de proveedor. Pantalla para ver productos nuevos*

**Productos Revisados:** Se consideran productos revisados a aquellos productos que han sido aprobados o rechazados por un comprador.

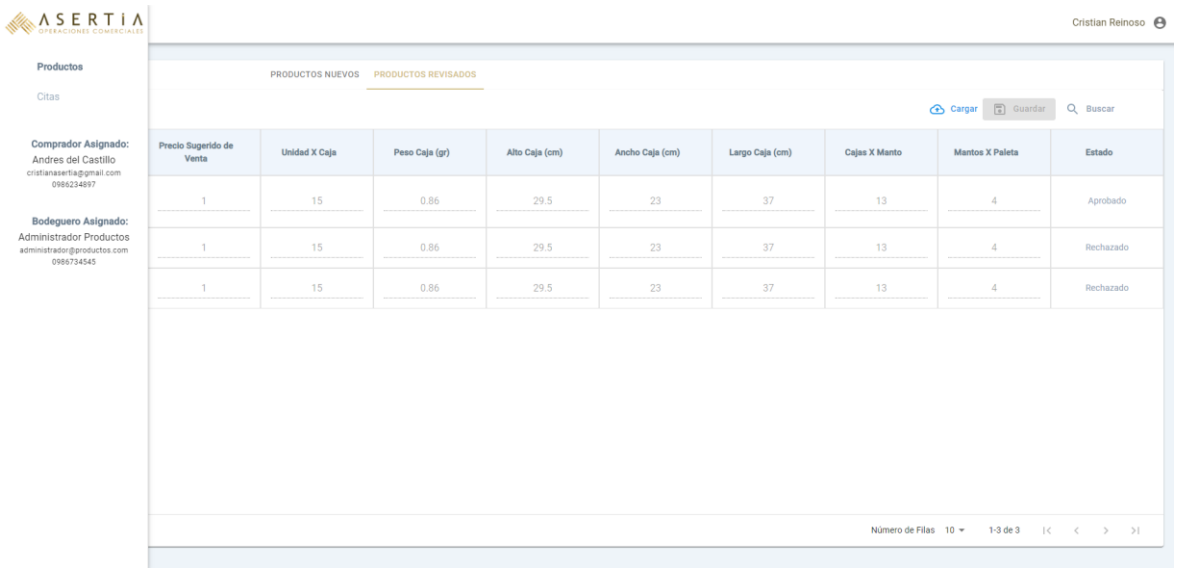



*Ilustración 25  Prototipo perfil de proveedor. Pantalla para ver productos revisados*

En la parte superior derecha de la tabla estarán ubicados los botones para: cargar nuevos productos, y una barra de búsqueda. El usuario podrá visualizar el estado de cada producto en la parte final de cada fila de la tabla.  Los comentarios de los productos rechazados podrán

ser visualizados presionando la palabra “rechazado” ubicada en la columna de estado al final de la tabla.

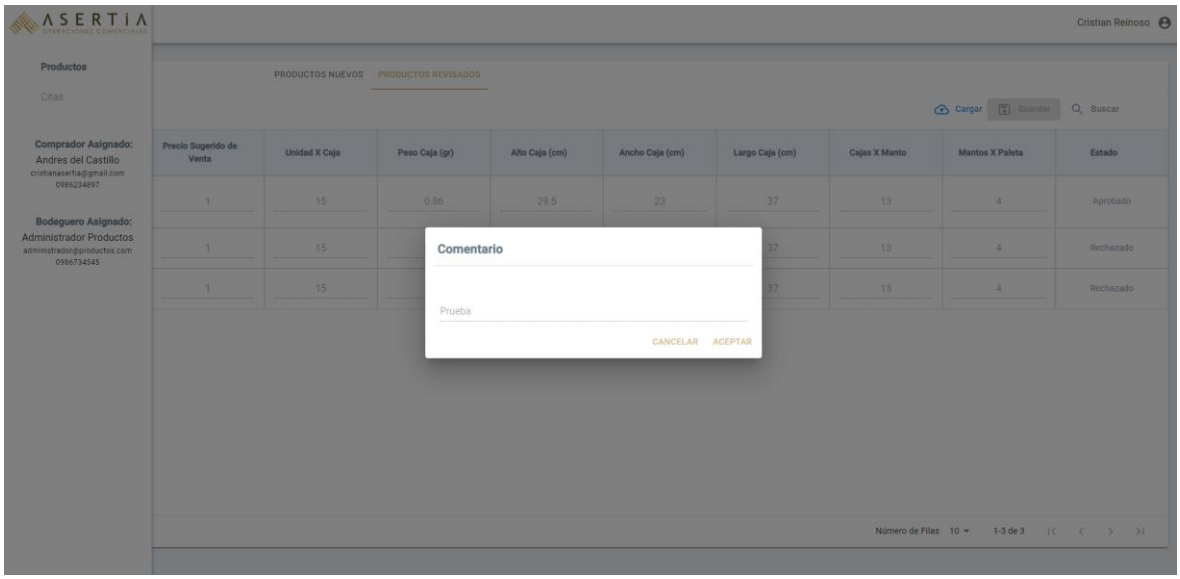



*Ilustración 26  Prototipo perfil de proveedor. Pantalla para ver comentarios de productos rechazados*

**Citas:** El sistema presentará un calendario donde se encontrará la información de las citas publicadas por su comprador asignado las cuales son de color verde y las citas ya agendadas las cuales son de color gris.

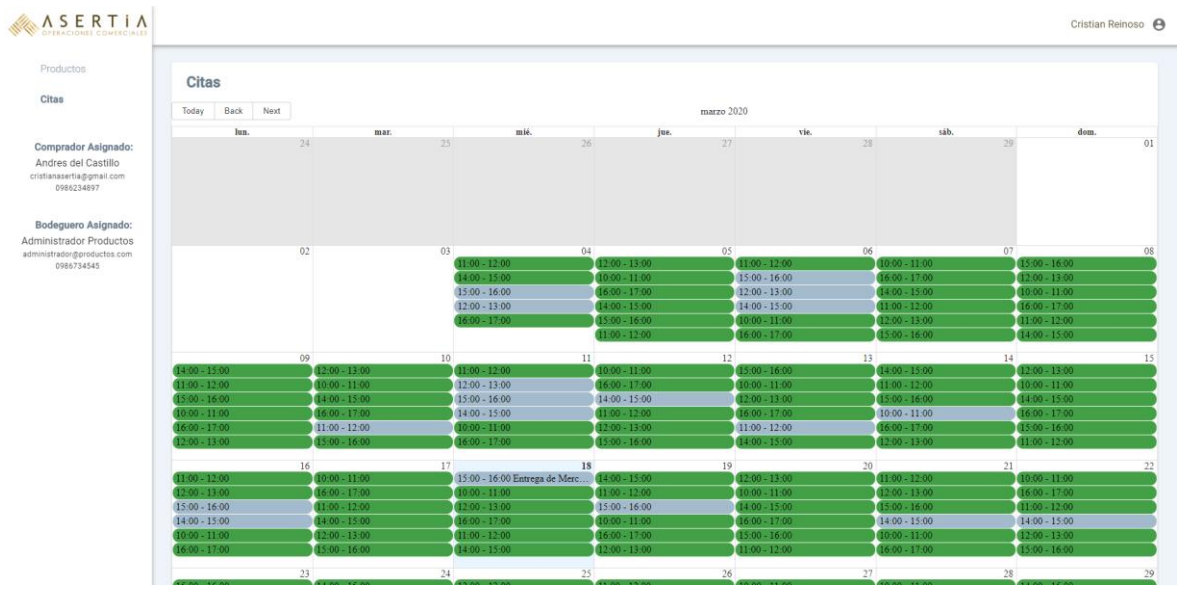



*Ilustración 27  Prototipo perfil de proveedor. Pantalla para ver citas*

El usuario podrá reservar una cita presionando en la cita que desea agendar, a continuación de le presentará un dialogo con los datos de la cita seleccionada para que confirme su reserva.

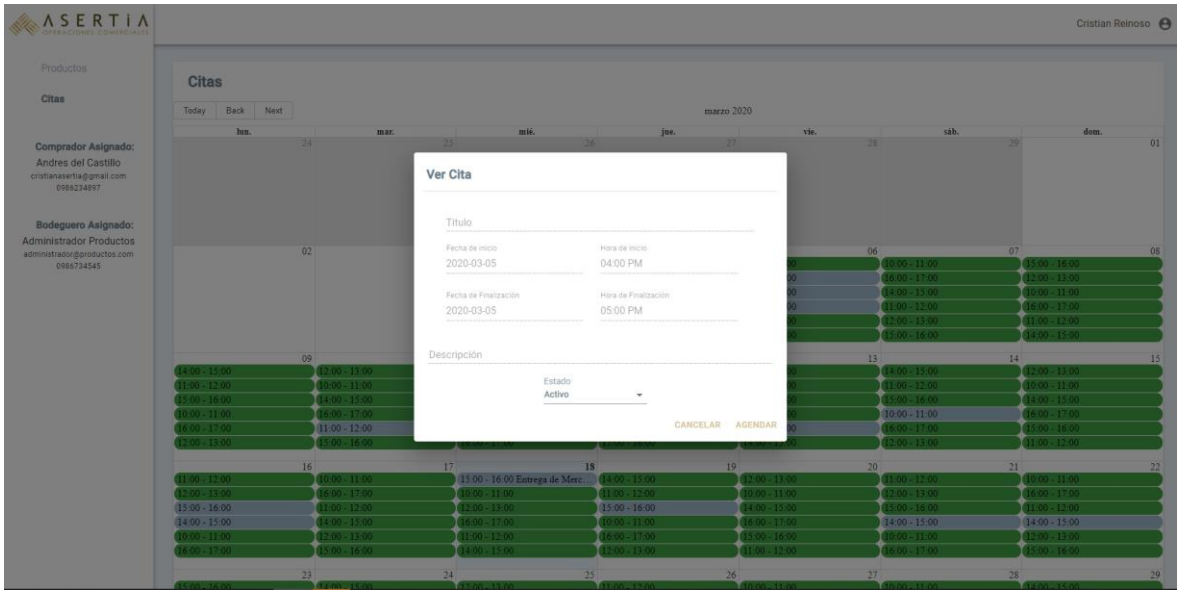



*Ilustración 28  Prototipo perfil de proveedor. Pantalla para reservar una cita*

## 3.3 SELECCIÓN DE HERRAMIENTAS DE DESARROLLO ADECUADAS A SER UTILIZADAS.

En esta sección se seleccionará la base de datos que utilizará el sistema web para almacenar la información. Además, se seleccionará el entorno de trabajo, librerías y herramientas que se utilizaran para desarrollar la aplicación del backend y frontend que componen el sistema web. Por último, se establecerán las plataformas en la nube en las que estarán publicadas estas aplicaciones.

### 3.3.1 Base de Datos

En la actualidad existen dos maneras de almacenar datos en un sistema informático. Por un lado, están las bases de datos relacionales las cuales son bases de datos que representan los datos y la relación que tienen mediante varias tablas. Por otro lado, están las bases de datos no relacionales, también conocidas como NoSQL (no solo SQL) termino que se dio a todas las tecnologías de almacenamiento estructurado que no cumplen el esquema relacional. Este tipo de bases de datos surgen por la necesidad de manejar masivos volúmenes de datos.

Gracia del Busto y Yanes Enríquez (2012) afirman: “Aunque esta tecnología surgió de unas necesidades muy concretas, su difusión y algunos proyectos para encapsular sus funcionalidades y hacerlas más amigables a desarrolladores acostumbrados a SQL está provocando que también se usen en proyectos de pequeño tamaño, con lo que todo indica que a medio plazo convivirán con las bases de datos tradicionales independientemente del volumen de datos a gestionar.” (p 24)

Una de las categorías de las bases de datos no relacionales es el almacenamiento de documentos la cual está orientada a almacenar documentos o datos semiestructurados. Esto representa una ventaja enorme en el rendimiento y escalabilidad de los sistemas que lo

implementen ya que, al almacenar los datos como objetos, facilita a los gestores de datos la recuperación de información ya que evitan hacer operaciones complejas.

Otra ventaja de este tipo de almacenamiento de dato es que los objetos o documentos que se almacenan tienen la notación de objetos de JavaScript (JSON) lo cual facilita su manipulación al momento de programar.

Tomando en cuenta que el sistema que se desarrollará en el presente trabajo de disertación está orientado a la web y de acuerdo con los requerimientos previamente planteados. Se decidió que la base de datos que se utilizará será una base de datos no relacionales debido a la velocidad de respuesta de las consultas y al manejo de la información como documento ya que esto facilita la programación del sistema.

Para seleccionar la herramienta más adecuada compararemos 3 bases de datos no relacionales las cuales son: MongoDB, Amazon DynamoDB y Azure Cosmos DB. Estas tres herramientas serán comparadas a alto nivel bajo los siguientes criterios:

- Facilidad de conexión.
- Rendimiento
- Costo
- Popularidad
- Conocimiento de la herramienta

En la siguiente tabla estableceremos una valoración a las herramientas bajo cada criterio tomando como base la información expuesta por cada herramienta en sus sitios web oficiales. La valoración a cada criterio estará entre el rango del 1 al 5 siendo 1 bajo y 5 alto.

**Tabla 3**

*Comparación entre bases de datos.*

| | **MongoDB** | **DynamoDB** | **Cosmos DB** |
|---|---|---|---|
| Facilidad de conexión | 5 | 4 | 3 |
| Rendimiento | 5 | 5 | 5 |
| Costo | 5 | 2 | 2 |
| Popularidad | 5 | 3 | 4 |
| Conocimiento de la herramienta | 5 | 3 | 3 |
| **TOTAL** | **25** | **17** | **17** |

Como se evidencia en la tabla anterior, la mejor opción de base de datos es MongoDB con una puntuación de 25. La ventaja de MongoDB frente a las demás herramientas es su facilidad de conexión con los frameworks de desarrollo. Además, MongoDB cuenta con una versión gratuita para ser utilizada mientras que el resto de las herramientas cobran un costo mensual por utilizar sus servicios de almacenamientos.

Para el presente sistema a desarrollar se utilizará MongoDB la cual es una base de datos no relacional basada en documentos de código abierto. Además, tiene un servicio de base de datos en la nube llamado MongoDB Atlas el cual es administrado por Amazon web Services, Azure o Google Cloud Plataform.

MongoDB Atlas ofrece una base de datos gratuita la cual será utilizada para el desarrollo del sistema. Este servicio de almacenamiento cuenta con 512 MB de almacenamiento, memoria RAM compartida y un conjunto de réplicas de alta disponibilidad, cifrado de extremo a extremo, parches automatizados, API REST.

### 3.3.2 Backend

El backend es una aplicación que tiene tres funciones principales: manejar la conexión con la base de datos, manejar la lógica del programa y, por último, exponer servicios para que puedan ser consumidos por otras aplicaciones.

Para seleccionar la herramienta más adecuada con la cual desarrollaremos la aplicación del backend, realizaremos una pequeña comparación entre dos de los más populares y utilizados frameworks para el desarrollo del backend como son: Node.js y Laravel.

Es importante tomar en cuenta que el código del presente sistema web posiblemente será utilizado e incrementado por otros desarrolladores, con la finalidad de escalar a un sistema web con más funcionalidades. Es por esto que, los criterios bajo los cuales compararemos estos dos frameworks son los siguientes:

- Facilidad de aprendizaje.
- Popularidad.
- Facilidad de conexión con la base de datos seleccionada.
- Conocimiento de la herramienta.

En la siguiente tabla evaluaremos cada uno de estos criterios de acuerdo con la documentación oficial de las distintas herramientas y de acuerdo con mi experiencia como desarrollador. La valoración estará en el rango de 1 a 5 siendo 1 bajo y 5 alto.

**Tabla 4**

*Comparación entre frameworks para el backend*

| | **Node.js** | **Laravel** |
|---|---|---|
| Facilidad de aprendizaje | 4 | 3 |
| Popularidad | 4 | 3 |
| Facilidad de conexión con la base de datos | 5 | 2 |
| Conocimiento de la herramienta | 5 | 3 |
| **TOTAL** | **18** | **11** |

Podemos concluir que la herramienta más adecuada para desarrollar la aplicación del backend para el presente sistema web es Node.js ya que la curva de aprendizaje es mucho más baja a comparación de Laravel. Además, la conexión a la base de datos es más sencillo y se necesita menos líneas de código con Node.js.

#### 3.3.2.1 Node.js

Node.js es un framework o entorno de trabajo que tiene como principal característica el motor V8 de Google Chrome y que facilita el desarrollo de aplicaciones que realicen las funciones del backend ya que este framework permite manejar servidores utilizando JavaScript. Node.js Está diseñado para construir aplicaciones web de extrema escalabilidad gracias a una ingeniosa combinación de: JavaScript del lado del servidor, programación asincrónica, construido alrededor de funciones anónimas de JavaScript y una arquitectura de ejecución de eventos de subproceso único (Herron, 2016),

La comunidad de Node.js ha desarrollado varios frameworks para desarrollar aplicaciones web entre los cuales podemos encontrar a Express.js, el cual es Infraestructura web rápida, minimalista y flexible para Node.js (Express.js, 2019). Gracias a Express.js se pueden crear servicios HTTP de forma sencilla y rápida facilitando el desarrollo de la aplicación.

Entre las principales razones por las que se decidió utilizar Node.js y Express como entorno de trabajo para desarrollar el backend de la aplicación web están:

**Popularidad:** Node.js como entorno de trabajo es utilizado por varias empresas reconocidas mundialmente como, por ejemplo: la plataforma en línea de compra de Walmart, PayPal, LinkedIn, y Ebay. Tal es la popularidad de Node.js que la empresa PayPal remplazo su sistema basado en Java por un sistema desarrollado con Node.js.

**Programación basada en eventos asíncronos:** La programación asíncrona es muy útil ya que reduce el tiempo de respuesta de un proceso o petición ya que estos no esperan a que el evento anterior termine para empezar, brindando una ventaja enorme sobre otras arquitecturas.

**Publicar servicios:** Node.js complementado con Express.js bridan una manera muy sencilla de exponer servicios utilizando el protocolo de transferencia de datos HTTP. Esto sin duda es una de las principales ventajas ya que facilita la programación de los servicios web y ayudan a mantener una estructura más organizada el desarrollo de los servicios web que serán expuestos para que otras aplicaciones los consuman.

**Compatibilidad con la base de datos:** Node.js cuenta con varias librerías o paquetes que facilitan la conexión con la base de datos, en este caso MongoDB. Una de ellas es Mongoose la cual permite realizar un modelamiento de datos para ser almacenados mediante esquemas. Mediante Mongoose la conexión a la base de datos y la manipulación de los datos es realmente simple ya que se requiere de poco código para realizarlo y nos ayuda a mantener un orden de los datos mediante los esquemas.

#### 3.3.2.2 Alojamiento de la aplicación

Con la finalidad de que la aplicación se encuentre disponible en todo momento, se decidió que la aplicación del backend de este sistema web se encontrará alojada en una plataforma como servicio en la nube la cual es Heroku. Heroku permite alojar aplicaciones desarrolladas en Node.js, PHP, Ruby, Java, Python, Go, Scala, Clojur, entre otras. Además, cuenta con una opción gratuita para que los usuarios de esta plataforma puedan utilizarla. Con la ayuda de esta plataforma tendremos nuestra aplicación del backend de este sistema web desarrollada en Node.js en la nube, esto quiere decir que podrá ser asequible desde cualquier lugar, y de forma gratuita.

#### 3.3.2.3 Herramientas de Desarrollo

##### 3.3.2.3.1 Visual Studio Code

Como principal herramienta de desarrollo se utilizará el editor de código fuente llamado Visual Studio Code, el cual nos ayudará a codificar el código de nuestra aplicación. Visual Studio Code cuenta con varios plugins que facilitan la codificación de aplicaciones dándole una ventaja significativa sobre otros editores de texto.

##### 3.3.2.3.2 ESLint

Con la finalidad de mantener una buena calidad en el código fuente del programa se decidió utilizar la herramienta ESLint, la cual permite encontrar errores de sintaxis u otros errores sin necesidad de ejecutar el código. ESLint se basa en patrones o reglas de buenas prácticas de programación las cuales permiten mantener un estándar en todo el código de la aplicación de tal manera que otro desarrollador no tenga problemas en leer e interpretar el código. Sin embargo, ESLint te permite desactivar reglas para que estas no sean tomadas en cuenta.

ESLint es una utilidad de código abierto de JavaScript creada originalmente por Nicholas C. Zakas en junio de 2013. El “linting code” es un tipo de análisis estático que se usa con

frecuencia para encontrar patrones o códigos problemáticos que no se adhieren a ciertas pautas de estilo (ESLint, 2019).

Entre las principales empresas que utilizan ESLint para mantener un estándar en el código de sus aplicaciones están: Netflix, PayPal, Airbnb, Microsoft, MongoDB, Facebook, Angular, entre otras.

### 3.3.3 Frontend

El frontend es una aplicación que se encarga principalmente en presentar la información al usuario e interactuar con el de una manera amigable. La aplicación del frontend también es la encargada de conectarse con la aplicación del backend.

Para seleccionar la herramienta más adecuada con la cual desarrollaremos la aplicación del frontend, realizaremos una pequeña comparación entre dos de los más populares y utilizados frameworks para el desarrollo del frontend, del presente sistema, como son: React y Angular

Es importante tomar en cuenta que el código del presente sistema web posiblemente será utilizado e incrementado por otros desarrolladores, con la finalidad de escalar a un sistema web con más funcionalidades. Es por esto que, los criterios bajo los cuales compararemos estos dos frameworks son los siguientes:

- Facilidad de aprendizaje.
- Popularidad.
- Facilidad de conexión con el backend.
- Conocimiento de la herramienta.

En la siguiente tabla evaluaremos cada uno de estos criterios de acuerdo con la documentación oficial de las distintas herramientas y de acuerdo con mi experiencia como desarrollador. La valoración estará en el rango de 1 a 5 siendo 1 bajo y 5 alto.

**Tabla 5**

*Comparación entre frameworks para el frontend*

| | **React** | **Angular** |
|---|---|---|
| Facilidad de aprendizaje | 5 | 3 |
| Popularidad | 5 | 3 |
| Facilidad de conexión con el backend | 4 | 4 |
| Conocimiento de la herramienta | 5 | 2 |
| **TOTAL** | **19** | **12** |

Podemos concluir que la herramienta más adecuada para desarrollar la aplicación del frontend para el presente sistema web es React ya que la curva de aprendizaje es mucho más baja a comparación de Laravel. Además, el conocimiento de la herramienta es más alto con React lo cual facilitaría el desarrollo y se reducirá el tiempo de entrega del producto final.

#### 3.3.3.1 React

React es una librería de JavaScript de código abierto liberada por Facebook en 2003 enfocada en construir interfaces de usuario complejas (React.Js, 2018). React permite la construcción de complejas interfaces de usuario mediante componentes los cuales son pequeñas e independientes líneas de código.

Los componentes que utiliza React están escritos en JSX que es una extensión de JavaScript que, al igual que React, fue creada por Facebook el cual sirve como procesador y transforma el código de generado con React a JavaScript.

Una de las principales ventajas de React es que los componentes pueden ser creados por los desarrolladores y pueden ser reutilizados en varias interfaces. Además, existen varias librerías de componentes ya creados los cuales puedes importar y utilizarlos de manera fácil.

MaterialUI es una de estas librerías que ofrece varios componentes de React que te dan la libertar de poderlos utilizarlos y adaptarlos según la interfaz que desees crear. MaterialUI es una librería muy popular y reconocida ya que es utilizada por varias organizaciones como, por ejemplo: La Nasa, Netflix, Amazon, Unity, entre otras.

#### 3.3.3.2 Redux

Redux es una librería en JavaScript que permite mantener un flujo consistente de información en aplicaciones de frontend, Redux se basa en tres principios fundamentales:

1. **Solamente existe una única fuente de la verdad**

   En todas las interfaces de usuarios creadas con React, los datos se obtendrán del Store la cual según Redux, es la única fuente de la verdad de esta manera aseguramos que la información en toda la aplicación sea consistente.

2. **El estado de la aplicación es solo de lectura**

   Para cambiar el estado de la aplicación se deben utilizar acciones las cuales llaman al reductor el cual se encarga en modificar el estado que es la fuente de información de la aplicación. La interfaz de usuario reflejara estos cambios de manera automática ya que la información que se presenta al usuario es la obtenida del store. Este flujo permite tener un mayor control de los datos que se modifican y no permite a cualquier componente realizar cambios directos en el store sin antes tener que pasar por el reductor.

3. **Los cambios se realizan con funciones puras**

   Como lo mencionamos anteriormente los cambios que se realizan en el Store son hechos por el reductor el cual no es más que funciones puras, es decir, funciones que dados los mismos argumentos devuelve siempre el mismo resultado y que no afecta ni tiene relación con estados o variables fuera de su ámbito de ejecución.

En la siguiente imagen podemos apreciad de una manera más gráfica de cómo es el flujo de información utilizando la librería Redux.

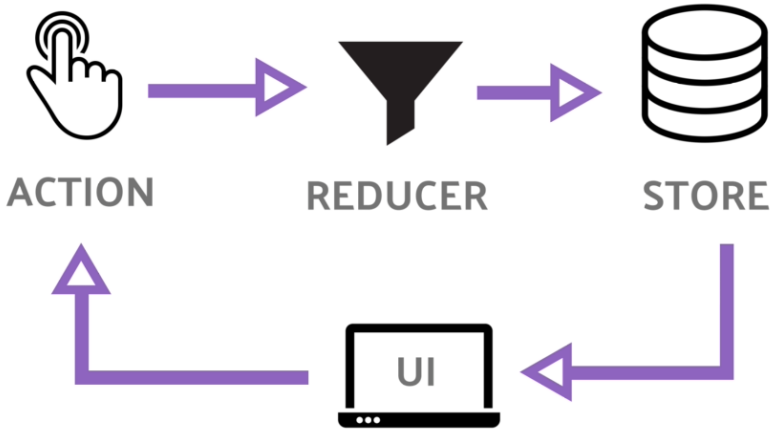



*Ilustración 29 Roa, A. (2018). Redux [figura]. Recuperado de https://medium.com/@alejandroroa/redux-sin-tanto-dolor-f5c44bc687c*

#### 3.3.3.3 Alojamiento de la aplicación

Con la finalidad de que la aplicación se encuentre disponible en todo momento, se decidió que la aplicación del frontend de este sistema web se encontrará alojada en una plataforma como servicio en la nube la cual es Netlify. Netlify caracteriza por permitir publicar, en la red, aplicaciones enfocadas al desarrollo de interfaces de usuario (frontend). Netlify soporta aplicaciones que utilizan los frameworks como: React, Vue.js, Next.js, Ruby, PHP, Angular, entre otros.

#### 3.3.3.4 Herramientas de Desarrollo

Las herramientas para desarrollar la aplicación del frontend de este sistema web será el editor de texto Visual Studio Code, al igual que para el desarrollo del backend. Aprovechando que la aplicación del backend como la aplicación del frontend de este sistema web serán

desarrolladas en el mismo lenguaje de programación, JavaScript, se utilizará la herramienta ESLint para mantener el mismo estándar de programación tanto en el backend como en el frontend.

## 3.4 DEFINICIÓN DEL MODELO DE LA BASE DE DATOS EN DONDE SE ALMACENARÁ LA INFORMACIÓN DEL SISTEMA.

En esta sección se establecerá el modelo de la base de datos que utilizará el sistema web a desarrollar. Se definirán los esquemas necesarios, según el contexto y los requerimientos previamente establecidos, con sus atributos respectivos. Además, se definirán los tipos de datos a almacenarse en cada atributo.

### 3.4.1 Modelo de la Base de Datos

Como se especificó en el anterior capítulo, la base de datos que se utilizará para almacenar la información en este sistema web será no relacional. Los datos se almacenarán de una manera semiestructurada, es decir, en forma de objetos o documentos.

En MongoDB los documentos son objetos en formato JSON que cumplen con un esquema preestablecido. Los esquemas vendrían a ser, en las bases de datos no relacionales. Lo que en las bases relacionales se los conoce como tablas. Los esquemas definen la forma y el contenido de los documentos.

De acuerdo con el contexto del negocio y en base a los requerimientos del sistema web, se lograron identificar los siguientes esquemas:

#### 3.4.1.1 Usuarios

El esquema de usuarios está dedicado para almacenar la información de todos los usuarios registrados en el sistema independientemente del rol con el que se registraron. Este esquema se utilizará principalmente para controlar el ingreso al sistema. Las propiedades o atributos de este usuario son las siguientes:

- ID

- Email
- Clave encriptada del usuario.
- Rol del usuario.
- Activo
- Fecha de creación
- Token para cambiar clave

#### 3.4.1.2 Proveedores

El esquema de proveedores está dedicado para almacenar la información propia de los de los usuarios con el perfil de proveedores. Los atributos o propiedades que se almacenarán serán las siguientes:

- ID
- Nombre:
- RUC:
- Nombre de la Empresa:
- Celular:
- Activo
- Email
- Fecha de Creación
- ID de Usuario
- ID del comprador asignado
- ID del Administrador de bodega asignado

#### 3.4.1.3 Compradores

El esquema de compradores está dedicado para almacenar la información propia de los usuarios con el perfil de compradores. Los atributos o propiedades que se almacenarán serán las siguientes:

- ID
- Nombre
- Celular
- Email
- Activo
- Administrador de bodega
- ID de Usuario

#### 3.4.1.4 Productos

El esquema de productos está dedicado para almacenar la información de los productos que se registren en el sistema. Los atributos o propiedades que se almacenarán serán las siguientes:

- ID
- ID del proveedor que creo el producto
- Código interno del producto
- Código del proveedor
- Código de barras
- Nombre del producto
- Categoría
- Subcategoría

- Familia
- Rentabilidad
- Costo con IVA
- Precio marcado
- Precio sugerido
- Unidad por caja
- Peso por caja
- Altura por caja
- Ancho por caja
- Longitud por caja
- Paleta por caja
- Manto por caja
- Manto por paleta
- Volumen
- Clase de expiración
- Comentario
- Aprobado
- Revisado
- Fecha de creación

#### 3.4.1.5 Citas

El esquema de Citas está dedicado para almacenar la información de todas las citas creadas en el sistema. Los atributos o propiedades que se almacenarán serán las siguientes:

- ID
- ID de comprador

- ID de proveedor
- Fecha y hora de Inicio
- Fecha y hora de finalización de la cita.
- Titulo
- Descripción
- Activa

Para una representación más grafico se decidió realizar un diagrama de clases del modelo de la base de datos del presente sistema web, ya que la información será almacenada en forma de objetos. El diagrama se realizó con la ayuda de la herramienta Power Designer

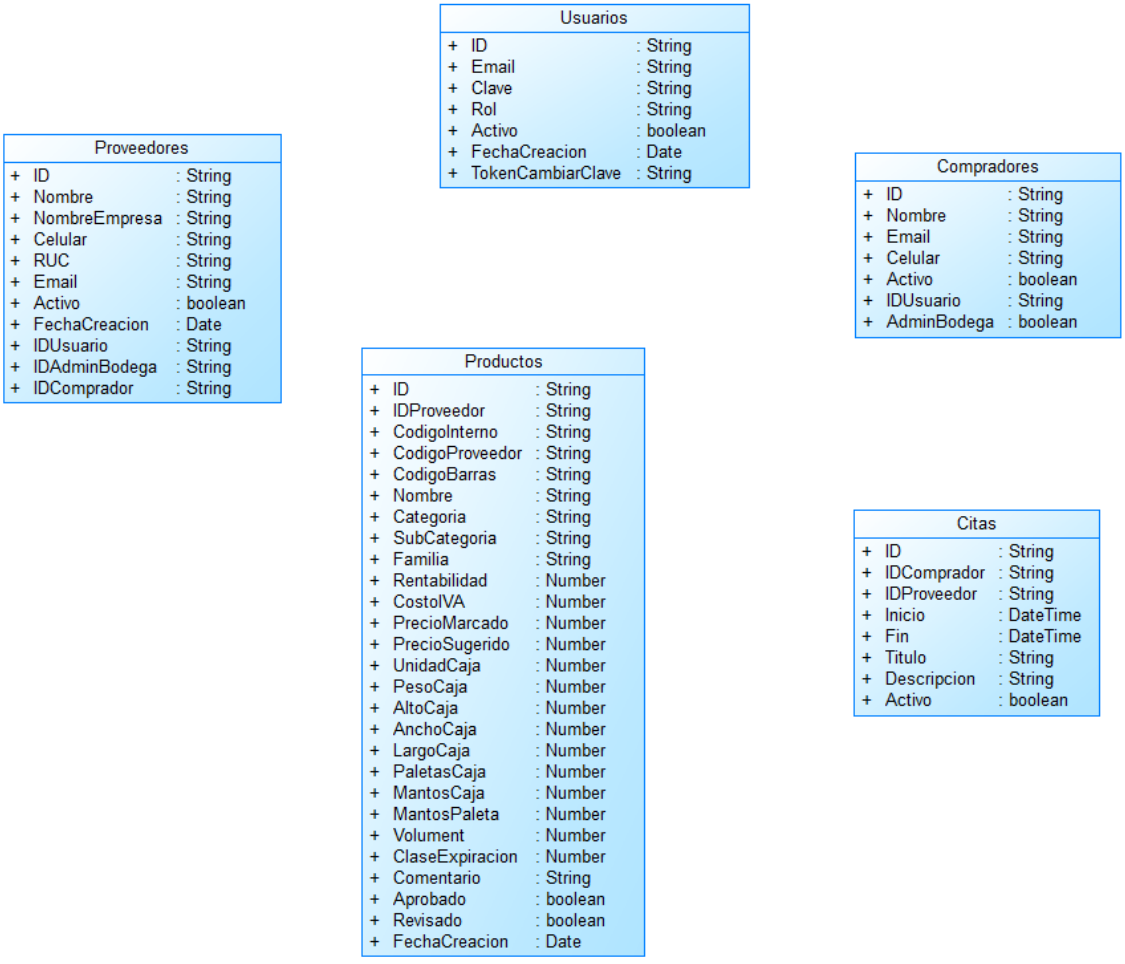



*Ilustración 30 Modelo de la base de datos*

## 3.5 IMPLEMENTACIÓN DE LA APLICACIÓN INCREMENTALMENTE CUMPLIENDO CON LAS FASES DE LA METODOLOGÍA DE DESARROLLO SCRUM.

En esta sección se realizará el desarrollo del sistema web incrementalmente utilizando la metodología de desarrollo Scrum. El desarrollo del sistema web se ha dividido en 3 Sprints de acuerdo con las funcionalidades más urgentes para el cliente.

### 3.5.1 SCRUM

Para el desarrollo de este sistema web se ha optado por seleccionar a Scrum como metodología de desarrollo ya que el cliente requiere un desarrollo incremental del presente sistema web con la finalidad de perfeccionarlo a lo largo del desarrollo de esta. Esto quiere decir que, el cliente estará permanentemente involucrado en el desarrollo de la aplicación y los requerimientos inicialmente planteados serán variables. Tomando en cuenta esto, Scrum es una metodología de desarrollo que se adapta favorablemente al contexto en el cual se desarrollará este sistema web.

#### 3.5.1.1 Especificación de Roles

A continuación, se detallarán los roles y responsables que conformarán el equipo Scrum para el desarrollo del sistema web.

***Tabla 6***

*Roles del equipo Scrum*

| ROL | RESPONSABLE |
|---|---|
| Product Owner | Cristian Reinoso |
| Scrum Master | Cristian Reinoso |
| Development Team | Cristian Reinoso |
| Cliente | Juan Francisco de la Cruz |

#### 3.5.1.2 Sprint 1

##### 3.5.1.2.1 Planificación del Sprint

Este Sprint está enfocado en desarrollar las funcionalidades del usuario con perfil de administrador ya que estas funcionalidades tienen mayor prioridad. En este Sprint también se desarrollarán las pantallas de registro para todos los usuarios del sistema, así como la pantalla de inicio de sesión. De esta forma, al final del Sprint tendremos un incremento utilizable para todos los usuarios ya que los usuarios con el perfil de comprador y proveedor podrán registrarse en el sistema mientras que el usuario con el perfil de administrador podrá realizar todas las funcionalidades planteadas en los requerimientos.

##### 3.5.1.2.2 Sprint Backlog

El Sprint Backlog o lista de pendientes correspondiente a este Sprint contempla los requerimientos con mayor prioridad asignados en el Product Backlog. A continuación, se detallan los requerimientos que conforman la lista de pendientes de este Sprint.

**Tabla 7**

*Sprint Backlog 1*

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| 1 | Crear una pantalla de registro para los proveedores. | 2 | Proveedor | 5 | 1 | 4 horas | El usuario con el perfil de proveedor podrá registrarse en el sistema desde cualquier dispositivo electrónico con navegador. |
| 2 | Crear una pantalla de registro para los compradores. | 2 | Comprador | 5 | 1 | 4 horas | El usuario con el perfil de comprador podrá registrarse en el sistema desde cualquier dispositivo electrónico con navegador. |
| 3 | Crear un cuadro de dialogo en la pantalla de compradores para que el administrador pueda agregar compradores | 2 | Administrador | 4 | 1 | 4 horas | El usuario con el perfil de administrador podrá crear compradores. |
| 4 | Crear un componente de selección por cada proveedor en donde el administrador pueda seleccionar el comprador que desea asignarle a ese proveedor. | 4 | Administrador | 4 | 1 | 8 horas | El usuario con el perfil de administrador podrá asignarle un comprador a cada proveedor. |

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| 5 | Implementar en la pantalla de proveedores una tabla, con paginación, en la que se encuentren todos los proveedores registrados en el sistema | 3 | Administrador | 4 | 1 | 6 horas | El usuario con el perfil de administrador podrá ver en una tabla todos los proveedores registrados en el sistema. |
| 6 | Implementar en la pantalla de compradores una tabla, con paginación, en la que se encuentren todos los compradores registrados en el sistema. | 3 | Administrador | 4 | 1 | 6 horas | El usuario con el perfil de administrador podrá ver en una tabla todos los compradores registrados en el sistema. |
| 7 | Implementar una barra de búsqueda en la tabla de proveedores que encuentren un proveedor por su nombre, email, ruc, nombre de la empresa o celular. | 2 | Administrador | 4 | 1 | 4 horas | El usuario con el perfil de administrador podrá buscar un proveedor por su nombre, email, ruc, nombre de la empresa o celular |
| 8 | Implementar un ordenamiento en la tabla de proveedores por: nombre, email, ruc, estado, nombre de la empresa y fecha de creación. | 2 | Administrador | 4 | 1 | 4 horas | El usuario con el perfil de administrador podrá ordenar proveedores por su nombre, email, ruc, estado, nombre de la empresa y fecha de creación. |

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| 9 | Implementar una barra de búsqueda en la tabla de compradores que encuentren un comprador por su nombre, email o celular. | 2 | Administrador | 4 | 1 | 4 horas | El usuario con el perfil de administrador podrá buscar un comprador por su nombre, email o celular. |
| 10 | Implementar un ordenamiento en la tabla de compradores por: nombre, email, estado y rol | 2 | Administrador | 4 | 1 | 4 horas | El usuario con el perfil de administrador podrá ordenar compradores por su nombre, email, estado y rol. |

#### 3.5.1.2.3 Duración del Sprint

De acuerdo con la estimación de tiempo de todos los requerimientos correspondientes a este Sprint, el tiempo de desarrollo de este Sprint será de 48 horas, es decir 6 días de 8 horas de trabajo cada uno.

#### 3.5.1.2.4 Revisión del Sprint

Después de finalizar el desarrollo de los requerimientos se presentó el primer incremento del sistema web al cliente. Se realizó la presentación de todas las funcionalidades desarrolladas, principalmente los formularios de registro para los usuarios con diferentes perfiles, así como las funcionalidades que podrá realizar el usuario con perfil de administrador. Finalmente, el cliente expresó su satisfacción con el resultado de este Sprint

### 3.5.1.3 Sprint 2

#### 3.5.1.3.1 Planificación del Sprint

Este Sprint está enfocado en desarrollar las funcionalidades del usuario con perfil de proveedor ya que, de las funcionalidades restantes, estas funcionalidades tienen mayor prioridad. De esta forma, al final del Sprint los usuarios con el perfil de proveedor podrán realizar todas las funcionalidades planteadas en los requerimientos.

#### 3.5.1.3.2 Sprint Backlog

El Sprint Backlog o lista de pendientes correspondiente a este Sprint contempla los requerimientos pendientes con mayor prioridad en el Product Backlog. A continuación, se detallan los requerimientos que conforman la lista de pendientes de este Sprint.

***Tabla 8***

*Sprint Backlog 2*

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| 1 | Implementar un botón que permita cargar un archivo en formato CSV y que ponga en una tabla la información de los productos encontrados en el archivo. | 3 | Proveedor | 3 | 2 | 6 horas | El usuario con el perfil de proveedor podrá cargar un archivo con información de productos y ver la información en una tabla. |
| 2 | Marcar en rojo los campos de los productos incorrectos o vacíos | 2 | Proveedor | 3 | 2 | 4 horas | El usuario con perfil de proveedor podrá visualizar los campos de los productos que no cumplen con los estándares. |
| 3 | Evitar que el usuario guarde productos para ser revisados que no cumplan con los estándares. | 1 | Proveedor | 3 | 2 | 2 horas | El usuario con perfil de proveedor no podrá guardar productos para ser revisados si no cumplen con los estándares. |
| 4 | Implementar una barra de búsqueda en la tabla de productos que encuentren un producto por su nombre, código interno, código de producto proveedor, | 2 | Proveedor | 4 | 2 | 4 horas | El usuario con el perfil de proveedor podrá buscar un producto por su nombre, código interno, código de producto |

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| | categoría, subcategoría o familia del producto. | | | | | | proveedor, categoría, subcategoría o familia del producto. |
| 5 | Implementar un ordenamiento en la tabla de productos por: código interno, código producto del proveedor, código de barras, nombre, categoría subcategoría, familia, rentabilidad, costo con IVA, precio marcado, precio sugerido, unidad por caja, peso por caja, alto caja, ancho caja, largo caja, cajas por paleta, cajas por manto, manto por paleta, volumen, clase de caducidad y estado | 2 | Proveedor | 4 | 2 | 4 horas | El usuario con el perfil de proveedor podrá ordenar productos por su código interno, código producto del proveedor, código de barras, nombre, categoría subcategoría, familia, rentabilidad, costo con IVA, precio marcado, precio sugerido, unidad por caja, peso por caja, alto caja, ancho caja, largo caja, cajas por paleta, cajas por manto, manto por paleta, volumen, clase de caducidad y estado. |
| 6 | Implementar dos pestañas en la pantalla de productos en los cuales se pueda ver una tabla con los productos revisados y productos no revisados | 3 | Proveedor | 3 | 2 | 6 horas | El usuario con perfil de proveedor podrá ver los productos revisados y productos no revisados. |
| 7 | Agregar un campo con nombre estado a cada producto revisado | 2 | Proveedor | 4 | 2 | 4 horas | El usuario con perfil de proveedor podrá ver que productos fueron aceptados y que productos fueron |

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| | para saber si el producto fue aceptado o rechazado | | | | | | rechazados por su comprador asignado. |
| 8 | Implementar un dialogo en el cual se pueda ver el comentario de un producto rechazado | 2 | Proveedor | 4 | 2 | 4 horas | El usuario con perfil de proveedor podrá ver la razón por la que un producto fue rechazado |
| 9 | Implementar un calendario en el cual se pueda ver las citas disponibles del comprador asignado a un proveedor | 4 | Proveedor | 3 | 2 | 8 horas | El usuario con perfil de proveedor podrá visualizar en un calendario las citas disponibles de su comprador asignado. |
| 10 | Enviar un correo electrónico al proveedor cuando su comprador asignado cancele una cita ya agendada. | 3 | Proveedor | 4 | 2 | 6 horas | El usuario con perfil de proveedor recibirá un correo electrónico cuando un comprador cancele una cita agendada. |

#### 3.5.1.3.3 Duración del Sprint

De acuerdo con la estimación de tiempo de todos los requerimientos correspondientes a este Sprint, el tiempo de desarrollo de este Sprint será de 48 horas, es decir 6 días de 8 horas de trabajo cada uno al igual que el Sprint anterior.

#### 3.5.1.3.4 Revisión del Sprint

Después de finalizar el desarrollo de los requerimientos se presentó el segundo incremento del sistema web al cliente. Se realizó la presentación de todas las funcionalidades desarrolladas las cuales se enfocan en las acciones que podrá realizar el usuario con perfil de proveedor. Finalmente, el cliente expresó su satisfacción con el resultado de este Sprint

### 3.5.1.4 Sprint 3

#### 3.5.1.4.1 Planificación del Sprint

Este Sprint está enfocado en desarrollar las funcionalidades del usuario con perfil de comprador ya que estas funcionalidades son las restantes para completar todos los requerimientos del Product Backlog. De esta forma, al final del Sprint los usuarios con el perfil de comprador podrán realizar todas las funcionalidades planteadas en los requerimientos.

#### 3.5.1.4.2 Sprint Backlog

El Sprint Backlog o lista de pendientes correspondiente a este Sprint contempla los requerimientos restantes del Product Backlog. A continuación, se detallan los requerimientos que conforman la lista de pendientes de este Sprint.

**Tabla 9**

*Sprint Backlog 3*

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| 1 | En la pantalla de citas de un comprador, implementar un dialogo que permita agregar una cita | 3 | Comprador | 2 | 3 | 4 horas | El usuario con perfil de comprador podrá agregar una cita. |
| 2 | Agregar citas desde las 10 de la mañana hasta las 5 de la tarda a todos los compradores con una duración de 1 hora por cada cita, exceptuando la hora de almuerzo de 1 de la tarde a 2 de la tarde. | 2 | Comprador | 2 | 3 | 4 horas | Los compradores por defecto tendrán citas creadas de lunes a viernes desde las 10 am hasta las 5 pm con tiempo de duración de una hora. |
| 3 | Enviar un correo electrónico al comprador cuando un proveedor agende una cita. | 3 | Comprador | 4 | 3 | 4 horas | El usuario con perfil de comprador recibirá un correo electrónico cuando un proveedor agende una cita. |
| 4 | Enviar un correo electrónico al comprador cuando un proveedor cancele una cita. | 3 | Comprador | 4 | 3 | 4 horas | El usuario con perfil de comprador recibirá un correo electrónico cuando un proveedor cancele una cita. |

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| 5 | Implementar una tabla en la cual se pueda ver los productos enviados por un proveedor con dos iconos al final de cada producto que permitan aprobar o rechazar el producto | 4 | Comprador | 2 | 3 | 6 horas | El usuario con perfil de Comprador podrá aprobar o rechazar productos. |
| 6 | Implementar, en la misma tabla de productos, entradas de texto donde su pueda editar de manera similar a una tabla de Excel, la información de los productos | 4 | Comprador | 2 | 3 | 6 horas | El usuario con perfil de comprador podrá editar en la tabla la información de los productos enviados por el proveedor. |
| 7 | Implementar una barra de búsqueda en la tabla de productos que permita buscar un producto por el código interno del producto o por el nombre del producto o por el código de producto proveedor o por la categoría del producto o por la subcategoría del producto o por la familia del producto. | 2 | Comprador | 4 | 3 | 4 horas | El usuario con perfil de comprador podrá buscar un producto por el código interno del producto o por el nombre del producto o por el código de producto proveedor o por la categoría del producto o por la subcategoría del producto o por la familia del producto. |

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| 8 | Implementar un ordenamiento en la tabla de productos por: código interno, código producto del proveedor, código de barras, nombre, categoría subcategoría, familia, rentabilidad, costo con IVA, precio marcado, precio sugerido, unidad por caja, peso por caja, alto caja, ancho caja, largo caja, cajas por paleta, cajas por manto, manto por paleta, volumen, clase de caducidad y estado | 2 | Comprador | 4 | 3 | 4 horas | El usuario con el perfil de compradores podrá ordenar productos por su código interno, código producto del proveedor, código de barras, nombre, categoría subcategoría, familia, rentabilidad, costo con IVA, precio marcado, precio sugerido, unidad por caja, peso por caja, alto caja, ancho caja, largo caja, cajas por paleta, cajas por manto, manto por paleta, volumen, clase de caducidad y estado. |
| 9 | Implementar dos pestañas en la pantalla de productos en los cuales se pueda ver una tabla con los productos revisados y productos no revisados | 4 | Comprador | 2 | 3 | 6 horas | El usuario con perfil comprador podrá ver los productos revisados y productos no revisados. |
| 10 | Implementar un botón que permita aprobar todos los productos y otro botón que permita rechazar todos los productos. | 3 | Comprador | 2 | 3 | 6 horas | El usuario con perfil de comprador podrá aprobar todos los productos o rechazar todos los productos. |

| Número | Descripción | Story Point | Rol | Prioridad | Sprint | Tiempo | Criterio de Aceptación |
|---|---|---|---|---|---|---|---|
| 11 | Implementar un botón que permita descargarse los productos aprobados en un archivo en formato CSV con la información de los productos enviada por los proveedores más la información interna de la empresa. | 4 | Comprador | 2 | 3 | 5 horas | El usuario con perfil comprador podrá descargarse un archivo en formato CSV con la información de los productos aprobados. |
| 12 | Implementar un icono para editar en la pantalla de proveedores que despliegue un dialogo que permita editar la información de un proveedor | 3 | Comprador | 2 | 3 | 6 horas | El usuario con perfil de comprador podrá editar la información de los proveedores que tiene asignado. |
| 13 | Implementare un componente de selección donde el comprador pueda activar o desactivar a un proveedor. | 2 | Comprador | 2 | 3 | 2 horas | El usuario con perfil de comprador podrá activar o desactivar a un proveedor. |

#### 3.5.1.4.3 Duración del Sprint

De acuerdo con la estimación de tiempo de todos los requerimientos correspondientes a este Sprint, el tiempo de desarrollo de este Sprint será de 61 horas, es decir 7 días de 8 horas de trabajo cada uno más un día de 5 horas, siendo este el Sprint más largo de todos.

#### 3.5.1.4.4 Revisión del Sprint

Después de finalizar el desarrollo de los requerimientos se presentó al cliente el tercer y último incremento del sistema web, siendo este el producto final. Se realizó la presentación de todas las funcionalidades desarrolladas en este Sprint las cuales se enfocan en las acciones que podrá realizar el usuario con perfil de comprador. Finalmente, el cliente expresó su satisfacción con el resultado final del desarrollo de este sistema web ya que este cumple con las expectativas del cliente.

## 3.6 PRUEBAS E INTEGRACIÓN DE LA APLICACIÓN WEB DESCRITA Y VERIFICACIÓN DEL FUNCIONAMIENTO

En esta sección se realizarán las pruebas al sistema basándonos en los criterios de aceptación de cada requerimiento establecido en el Product Backlog del sistema Web. Las pruebas del Sistema se realizaron por cada requerimiento desarrollado en cada Sprint y se utilizaron datos históricos reales proporcionados por los gestores de los procesos de Asertia.

### 3.6.1 Pruebas Sprint 1

**Requerimiento 1:** Crear una pantalla de registro para los proveedores.

**Criterio de aceptación:** El usuario con el perfil de proveedor podrá registrarse en el sistema desde cualquier dispositivo electrónico con navegador.

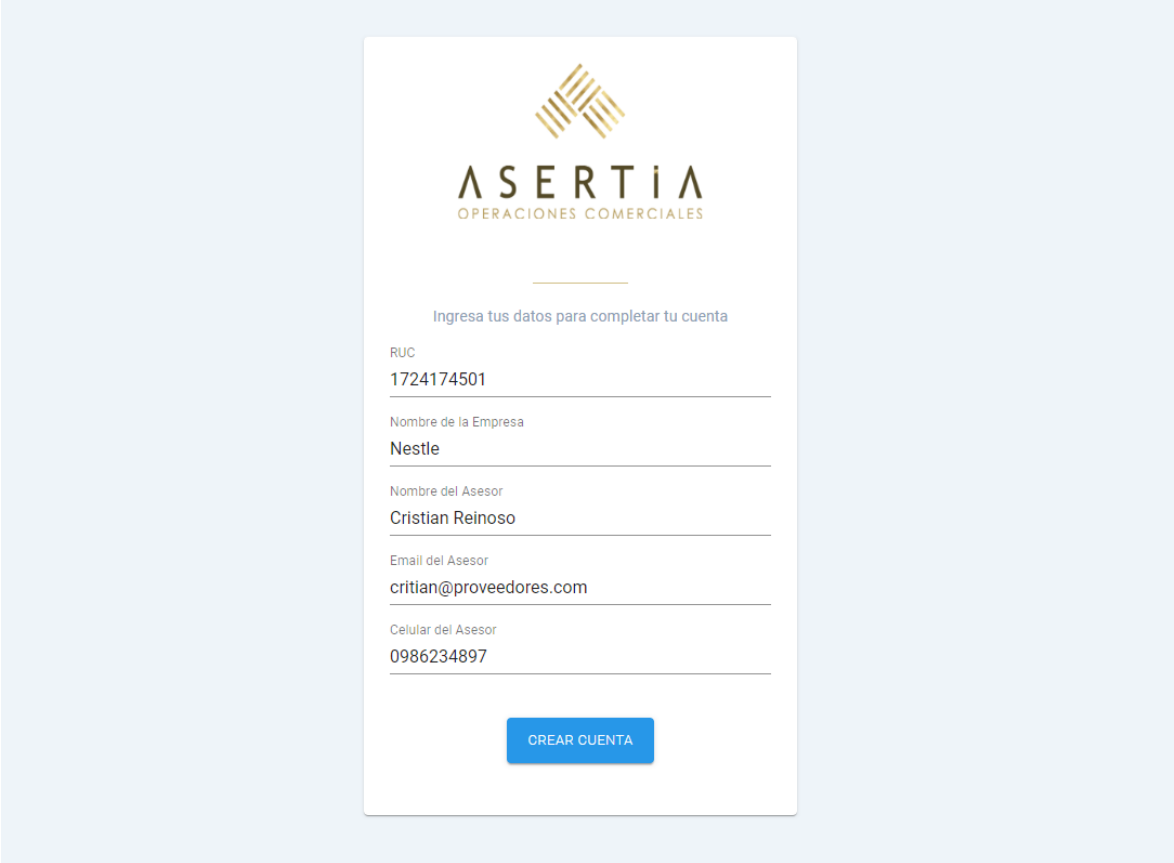



*Ilustración 31 Prueba del sistema. Requerimiento 1 del Sprint 1*

**Descripción de prueba:** En esta pantalla se demuestra que el usuario con perfil de proveedor podrá registrarse siempre y cuando llene todos los campos del formulario. Después de completar el registro, el sistema presenta el siguiente mensaje.




*Ilustración 32 Prueba del sistema. Requerimiento 1 mensaje del Sprint 1*

**Requerimiento 2:** Crear una pantalla de registro para los compradores.

**Criterio de aceptación:** El usuario con el perfil de comprador podrá registrarse en el sistema desde cualquier dispositivo electrónico con navegador.

**Descripción de prueba:** En la siguiente pantalla se demuestra que el usuario con perfil de comprador podrá registrarse siempre y cuando llene todos los campos del formulario.

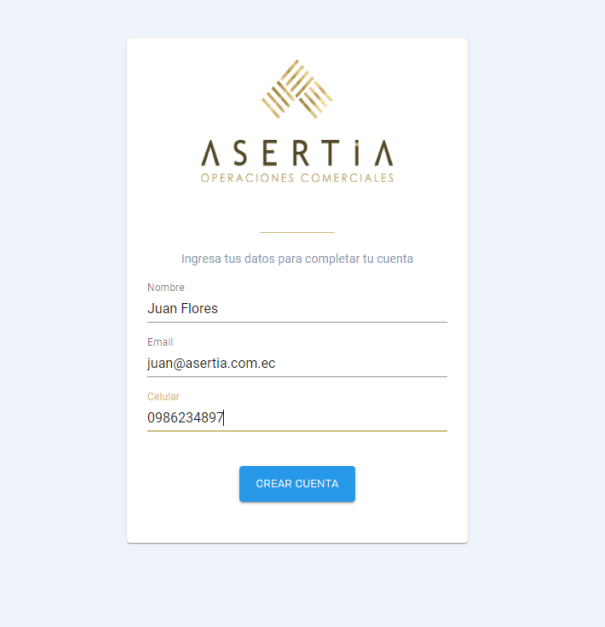



*Ilustración 33 Prueba del sistema. Requerimiento 2 del Sprint 1*

**Requerimiento 3:** Crear un cuadro de dialogo en la pantalla de compradores para que el administrador pueda agregar compradores.

**Criterio de aceptación:** El usuario con el perfil de Administrador podrá crear compradores.

**Descripción de prueba:** En esta pantalla se demuestra que el usuario con el perfil de administrador crea un comprador desde la pantalla de compradores.

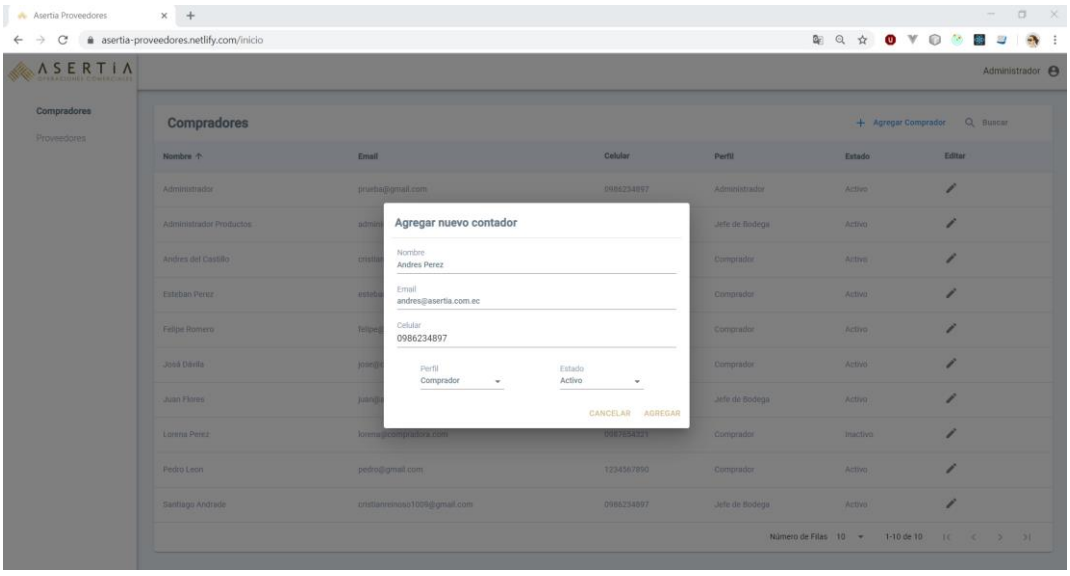



*Ilustración 34 Prueba del sistema. Requerimiento 3 del Sprint 1*

**Requerimiento 4:** Crear un componente de selección por cada proveedor en donde el administrador pueda seleccionar el comprador que desea asignarle a ese proveedor.

**Criterio de aceptación:** El usuario con el perfil de Administrador podrá asignarle un comprador a cada proveedor.

**Descripción de prueba:** En la siguiente imagen se demuestra que el usuario con el perfil de administrador podrá asignar un comprador a cada proveedor.

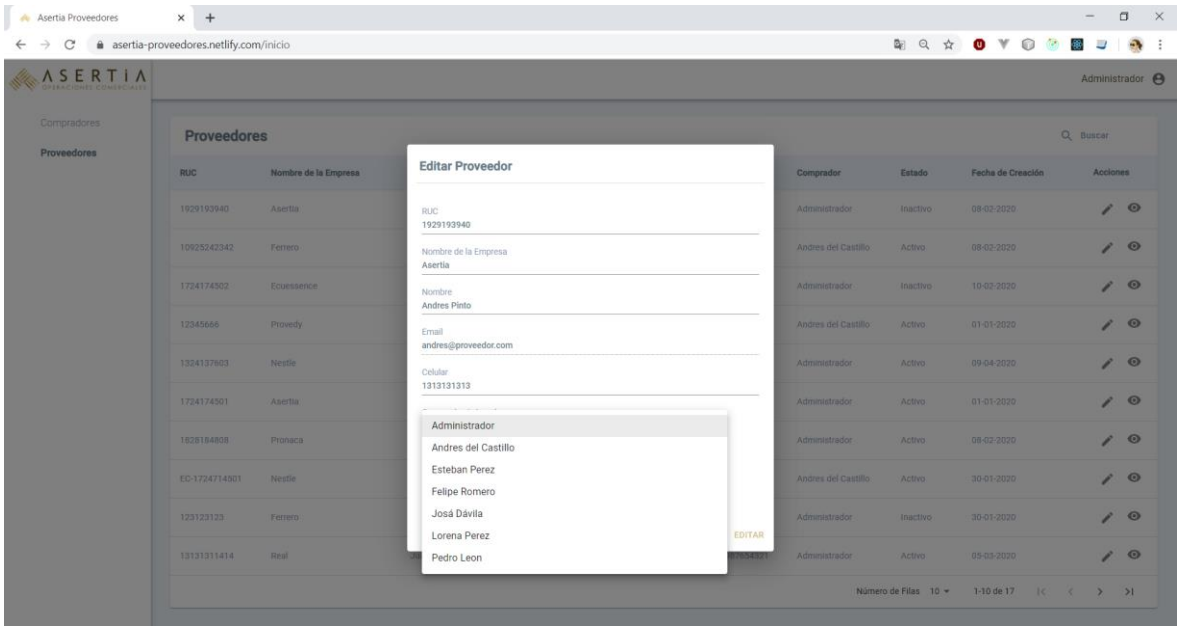



*Ilustración 35 Prueba del sistema. Requerimiento 4 del Sprint 1*

**Requerimiento 5:** Implementar en la pantalla de proveedores una tabla, con paginación, en la que se encuentren todos los proveedores registrados en el sistema

**Criterio de aceptación:** El usuario con el perfil de Administrador podrá ver en una tabla todos los proveedores registrados en el sistema.

**Descripción de prueba:** En la siguiente pantalla se muestra la tabla en la cual el usuario con perfil de administrador puede visualizar a todos los proveedores registrados en el sistema

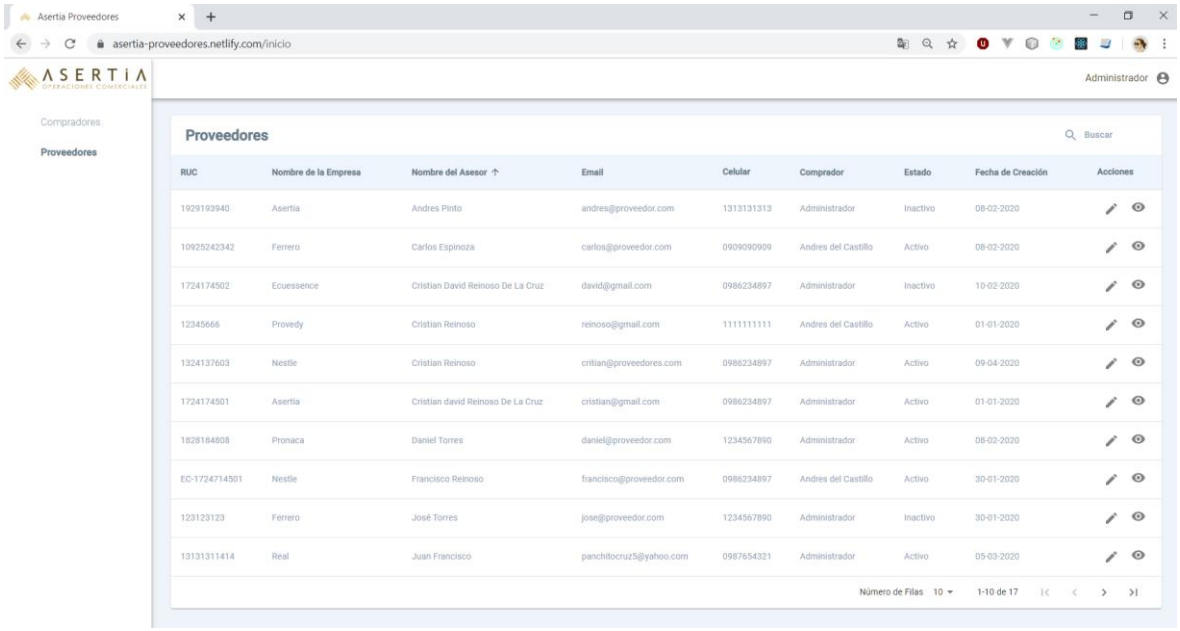



*Ilustración 36 Prueba del sistema. Requerimiento 5 del Sprint 1*

**Requerimiento 6:** Implementar en la pantalla de compradores una tabla, con paginación, en la que se encuentren todos los compradores registrados en el sistema.

**Criterio de aceptación:** El usuario con el perfil de Administrador podrá ver en una tabla todos los compradores registrados en el sistema.

**Descripción de prueba:** En la siguiente pantalla se muestra la tabla en la cual el usuario con perfil de administrador puede visualizar a todos los compradores registrados en el sistema

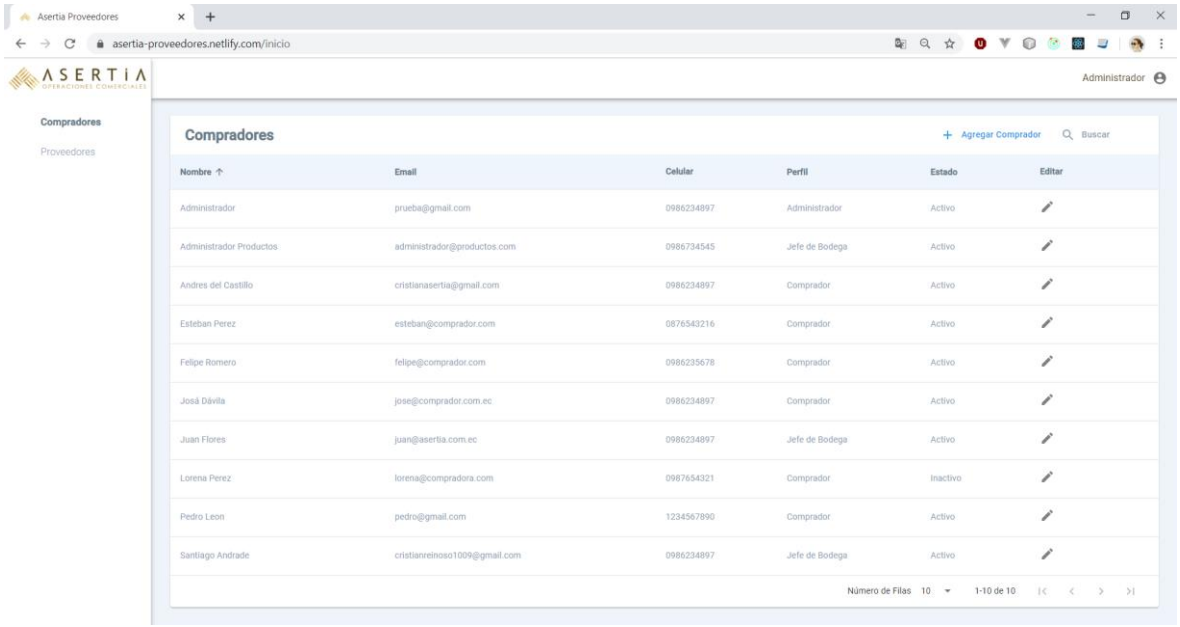



*Ilustración 37 Prueba del sistema. Requerimiento 6 del Sprint 1*

**Requerimiento 7:** Implementar una barra de búsqueda en la tabla de proveedores que encuentren un proveedor por su nombre, email, ruc, nombre de la empresa o celular.

**Criterio de aceptación:** El usuario con el perfil de Administrador podrá buscar un proveedor por su nombre, email, ruc, nombre de la empresa o celular.

**Descripción de prueba:** En la siguiente pantalla se muestra que el usuario con el perfil de administrador podrá buscar un proveedor por su nombre, email, ruc, nombre de la empresa o celular.

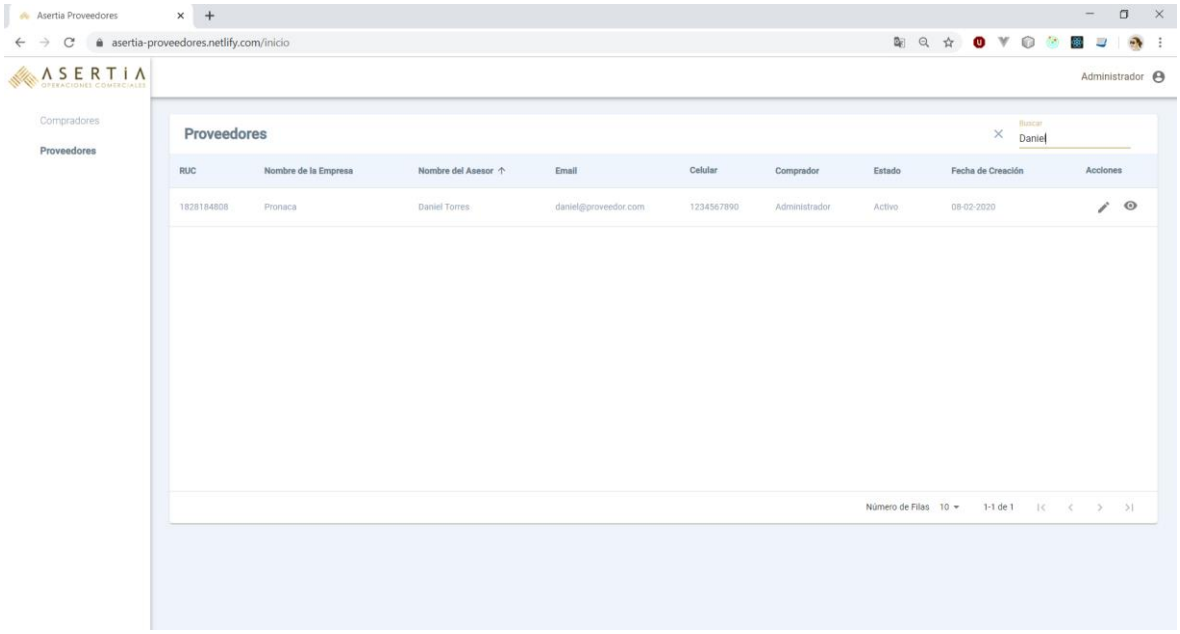



*Ilustración 38 Prueba del sistema. Requerimiento 7 del Sprint 1*

**Requerimiento 8:** Implementar un ordenamiento en la tabla de proveedores por: nombre, email, ruc, estado, nombre de la empresa y fecha de creación.

**Criterio de aceptación:** El usuario con el perfil de Administrador podrá ordenar proveedores por su nombre, email, ruc, estado, nombre de la empresa y fecha de creación.

**Descripción de prueba:** En la siguiente pantalla se muestra que el usuario con el perfil de administrador podrá ordenar un proveedor por su nombre, email, ruc, nombre de la empresa o celular.

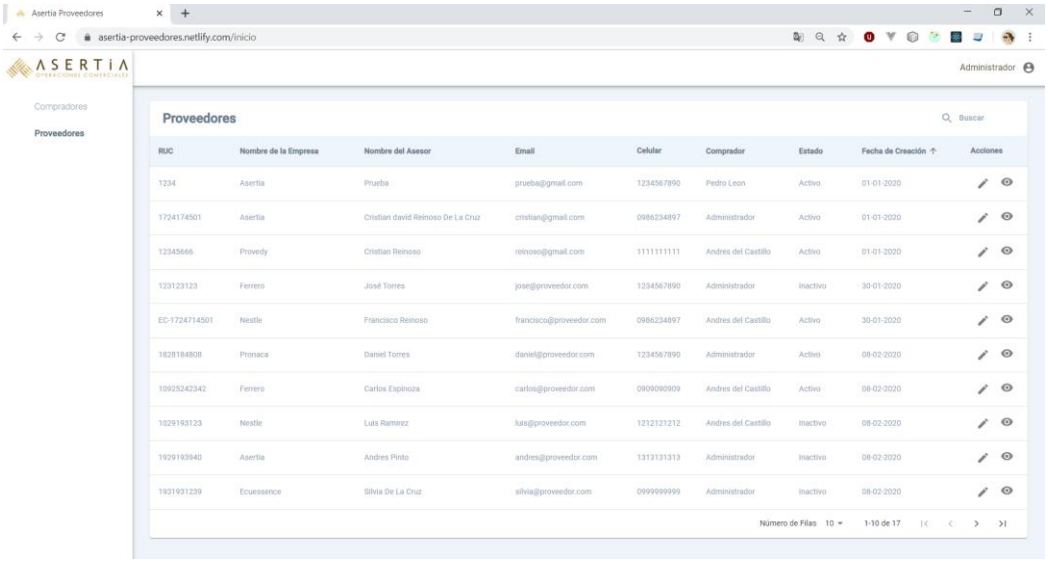



*Ilustración 39 Prueba del sistema. Requerimiento 8 del Sprint 1*

**Requerimiento 9:** Implementar una barra de búsqueda en la tabla de compradores que encuentren un comprador por su nombre, email o celular.

**Criterio de aceptación:** El usuario con el perfil de Administrador podrá buscar un comprador por su nombre, email o celular.

**Descripción de prueba:** En la siguiente pantalla se muestra que el usuario con el perfil de administrador podrá buscar un comprador por su nombre, email o celular.

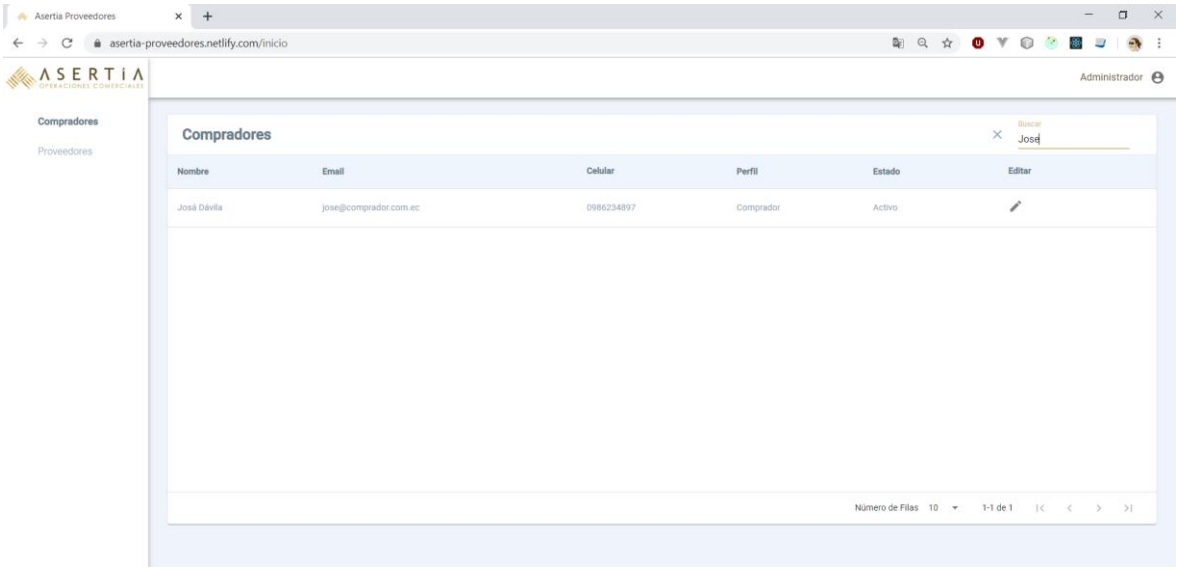



*Ilustración 40 Prueba del sistema. Requerimiento 9 del Sprint 1*

**Requerimiento 10:** Implementar un ordenamiento en la tabla de compradores por: nombre, email, estado y rol

**Criterio de aceptación:** El usuario con el perfil de Administrador podrá ordenar compradores por su nombre, email, estado y rol.

**Descripción de prueba:** En la siguiente pantalla se muestra que el usuario con el perfil de administrador podrá ordenar un comprador por su nombre, email o celular.

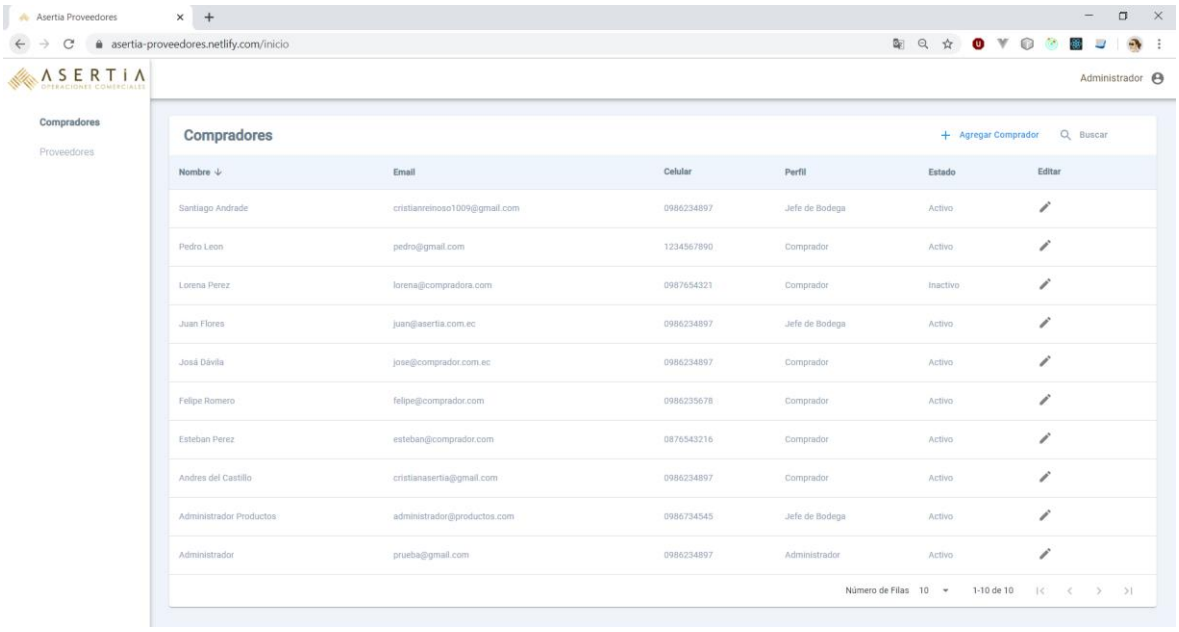



*Ilustración 41 Prueba del sistema. Requerimiento 10 del Sprint 1*

### 3.6.2 Pruebas Sprint 2

**Requerimiento 1:** Implementar un botón que permita cargar un archivo en formato CSV y que ponga en una tabla la información de los productos encontrados en el archivo.

**Criterio de aceptación:** El usuario con el perfil de Proveedor podrá cargar un archivo con información de productos y ver la información en una tabla.

**Descripción de prueba:** En la siguiente imagen se demuestra que el usuario con perfil de administrador puede subir un archivo en formato CSV con la información de los productos

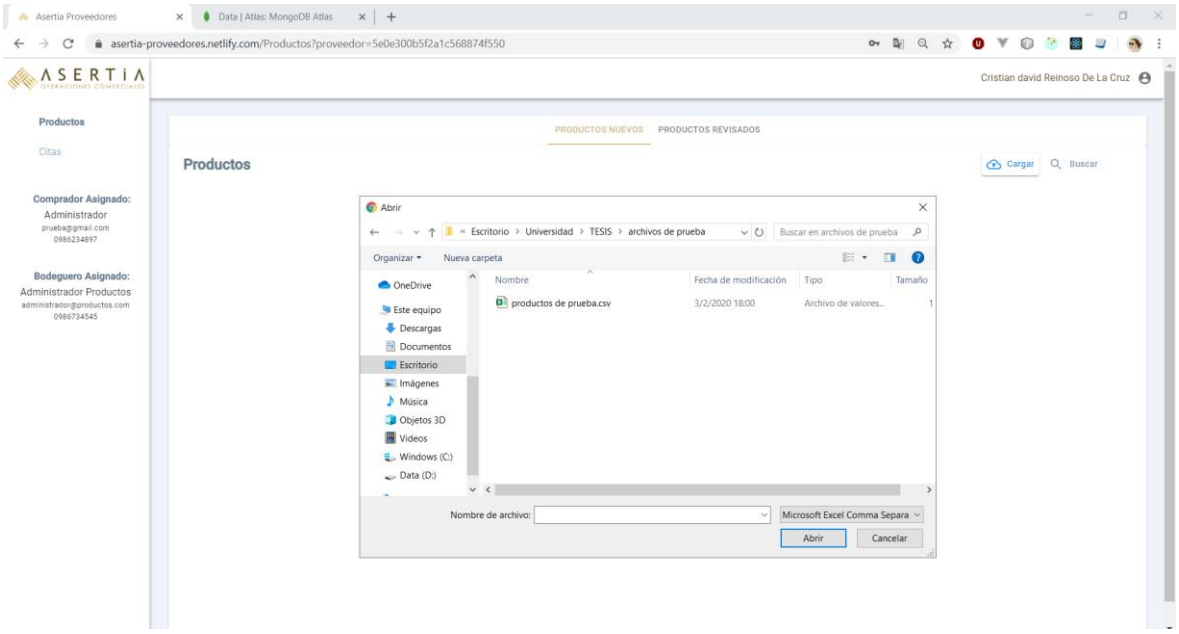



*Ilustración 42 Prueba del sistema. Requerimiento 1 carga de productos del Sprint 2*

Después de seleccionar el archivo la información de los productos se presentará en una tabla

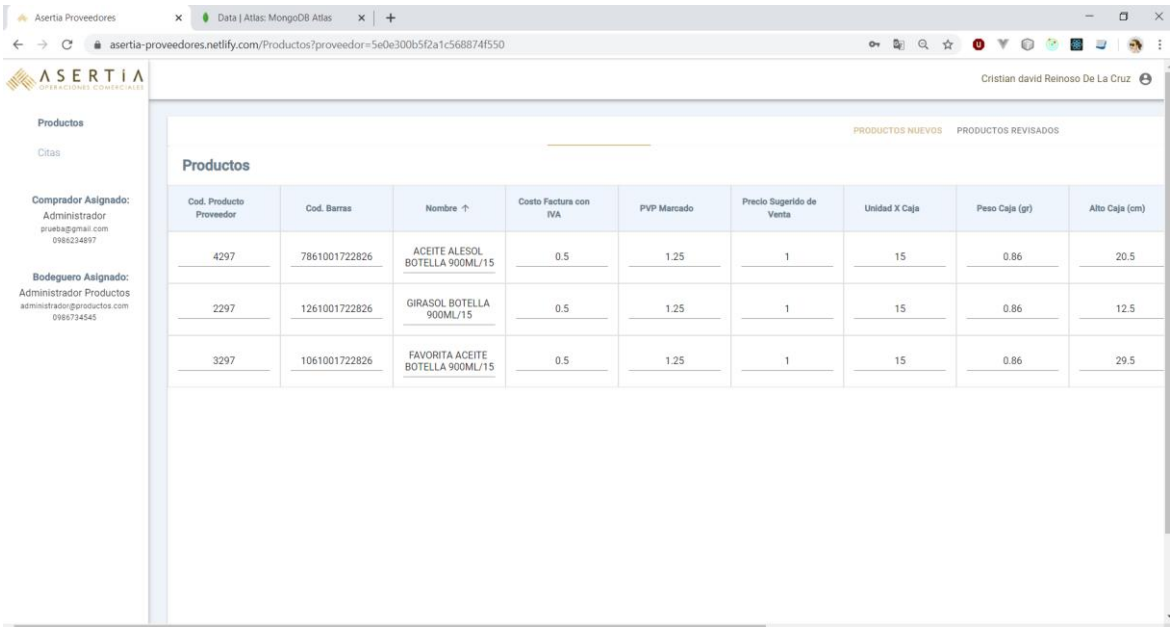



*Ilustración 43 Prueba del sistema. Requerimiento 1 del Sprint 2*

**Requerimiento 2:** Marcar en rojo los campos de los productos incorrectos o vacíos

**Criterio de aceptación:** El usuario con perfil de Proveedor podrá visualizar los campos de los productos que no cumplen con los estándares.

**Descripción de prueba:** En la siguiente imagen se demuestra que los productos que tengan campos vacíos o que no cumplan con el formato establecido son marcados con rojo.

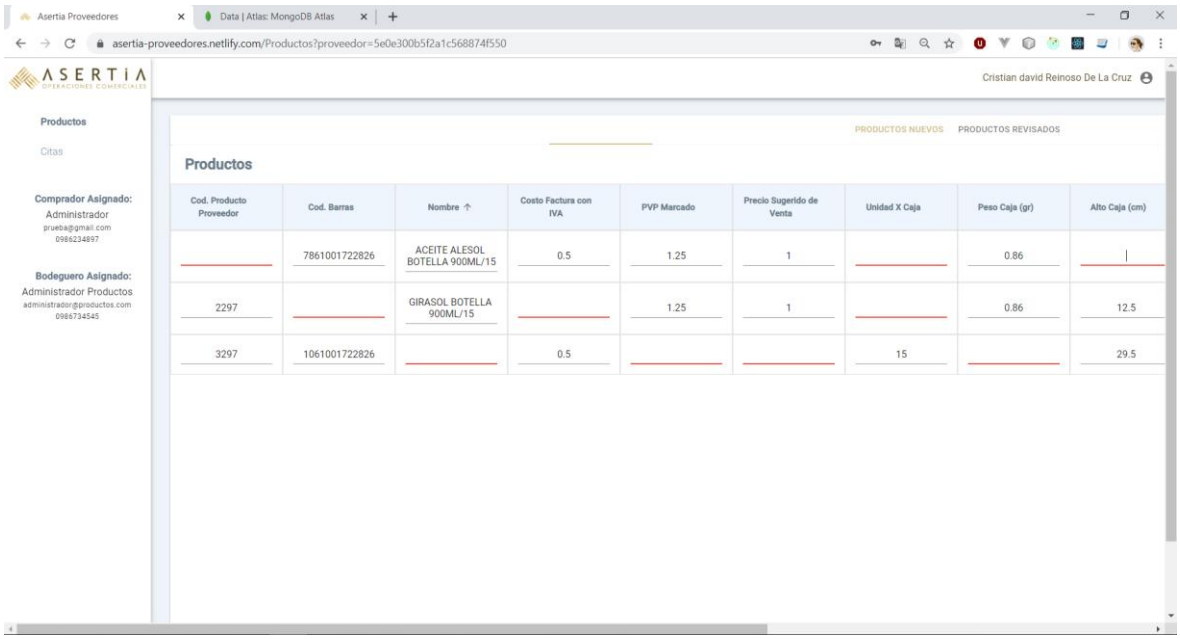



*Ilustración 44 Prueba del sistema. Requerimiento 2 del Sprint 2*

**Requerimiento 3:** Evitar que el usuario guarde productos para ser revisados que no cumplan con los estándares.

**Criterio de aceptación:** El usuario con perfil de proveedor no podrá guardar productos para ser revisados si no cumplen con los estándares.

**Descripción de prueba:** En la siguiente imagen se demuestra que mientras los productos no cumplan con el formato establecido, el usuario con perfil de proveedor no puede guardar la información.

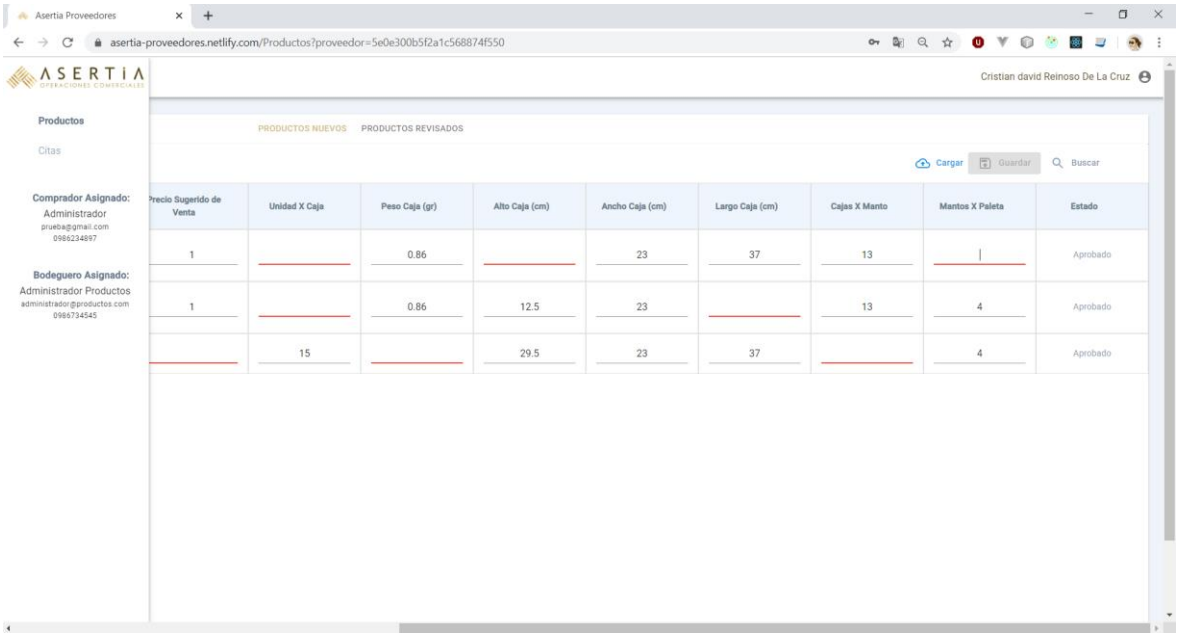



*Ilustración 45 Prueba del sistema. Requerimiento 3 del Sprint 2*

**Requerimiento 4:** Implementar una barra de búsqueda en la tabla de productos que encuentren un producto por su nombre, código interno, código de producto proveedor, categoría, subcategoría o familia del producto.

**Criterio de aceptación:** El usuario con el perfil de Proveedor podrá buscar un producto por su nombre, código interno, código de producto proveedor, categoría, subcategoría o familia del producto.

**Descripción de prueba:** En la siguiente imagen se demuestra que el usuario con perfil de proveedor puede buscar un producto mediante una barra de búsqueda.

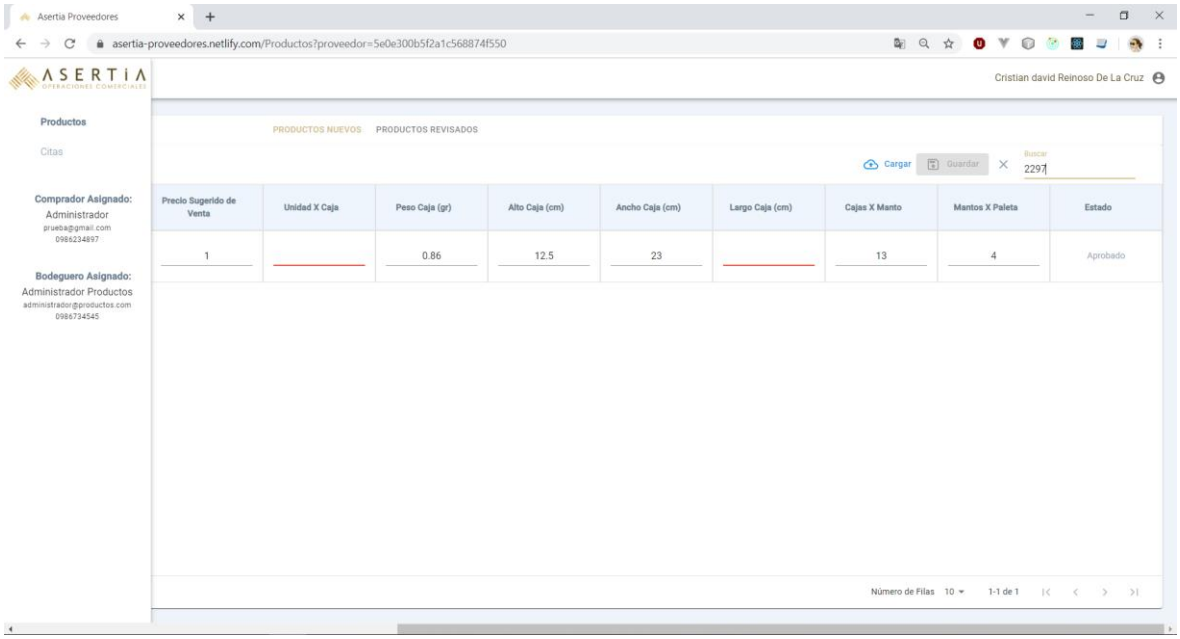



*Ilustración 46 Prueba del sistema. Requerimiento 4 del Sprint 2*

**Requerimiento 5:** Implementar un ordenamiento en la tabla de productos por: código interno, código producto del proveedor, código de barras, nombre, categoría subcategoría, familia, rentabilidad, costo con IVA, precio marcado, precio sugerido, unidad por caja, peso por caja, alto caja, ancho caja, largo caja, cajas por paleta, cajas por manto, manto por paleta, volumen, clase de caducidad y estado

**Criterio de aceptación:** El usuario con el perfil de Proveedor podrá ordenar productos por su código interno, código producto del proveedor, código de barras, nombre, categoría subcategoría, familia, rentabilidad, costo con IVA, precio marcado, precio sugerido, unidad por caja, peso por caja, alto caja, ancho caja, largo caja, cajas por paleta, cajas por manto, manto por paleta, volumen, clase de caducidad y estado.

En la siguiente imagen se demuestra que los productos pueden ser ordenados.

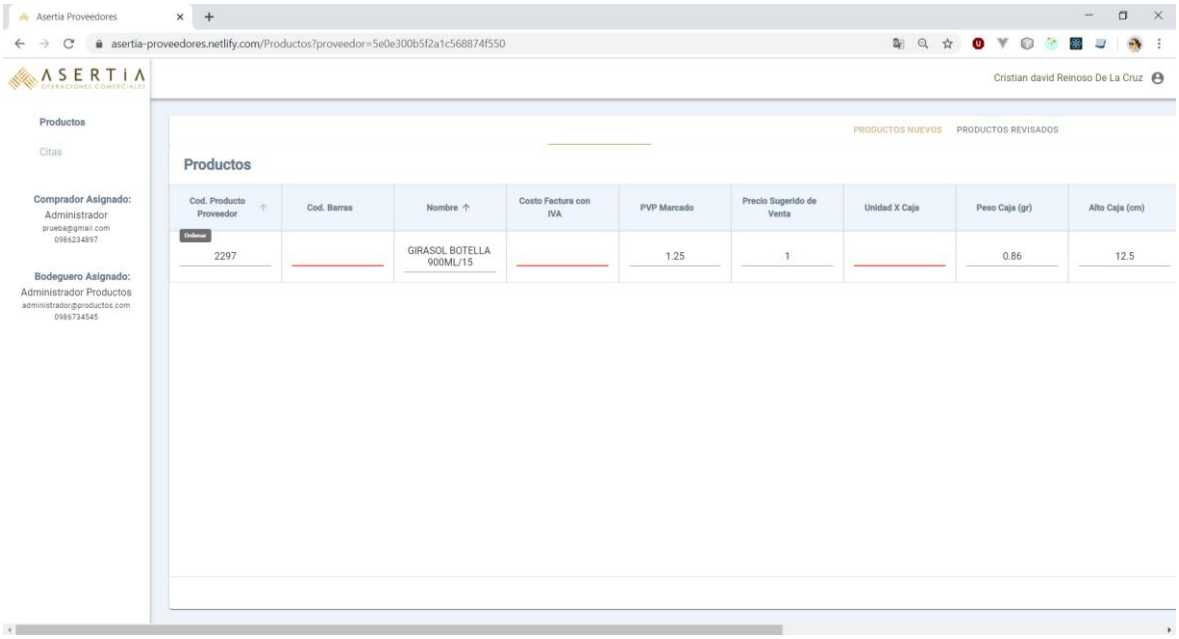



*Ilustración 47 Prueba del sistema. Requerimiento 5 del Sprint 2*

**Requerimiento 6:** Implementar dos pestañas en la pantalla de productos en los cuales se pueda ver una tabla con los productos revisados y productos no revisados.

**Criterio de aceptación:** El usuario con perfil proveedor podrá ver los productos revisados y productos no revisados.

**Descripción de prueba:** En la siguiente imagen se demuestra que el usuario puede ver los productos revisados y no revisados por medio de pestañas ubicadas en la parte superior de la tabla.

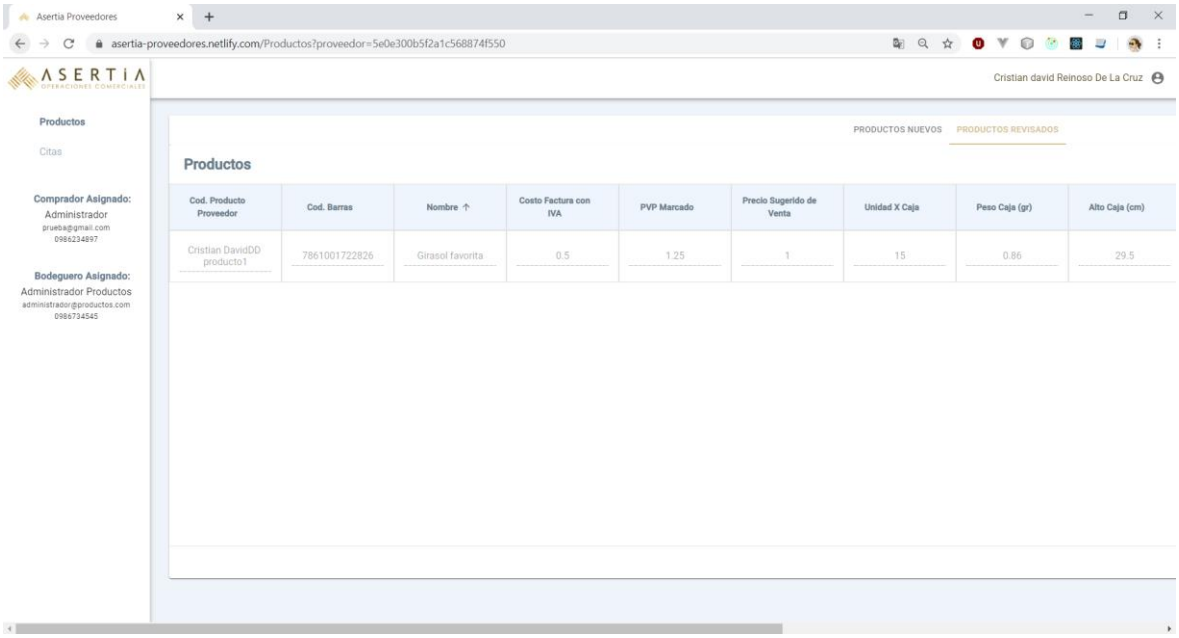



*Ilustración 48 Prueba del sistema. Requerimiento 6 del Sprint 2*

**Requerimiento 7:** Agregar un campo con nombre estado a cada producto revisado para saber si el producto fue aceptado o rechazado

**Criterio de aceptación:** El usuario con perfil de Proveedor podrá ver que productos fueron aceptados y que productos fueron rechazados por su comprador asignado.

**Descripción de prueba:** En la siguiente imagen se demuestra que al final de cada fila se demuestra el estado del producto, es decir, si fue aprobado o rechazado.

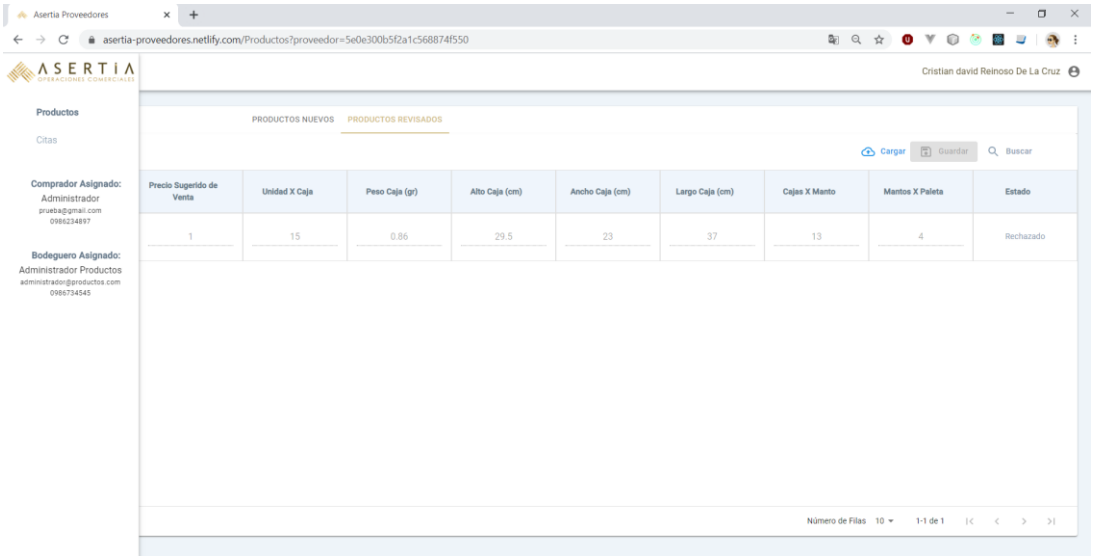



*Ilustración 49 Prueba del sistema. Requerimiento 7 del Sprint 2*

**Requerimiento 8:** Implementar un dialogo en el cual se pueda ver el comentario de un producto rechazado.

**Criterio de aceptación:** El usuario con perfil de Proveedor podrá ver la razón por la que un producto fue rechazado.

**Descripción de prueba:** En la siguiente imagen se demuestra que el usuario con el perfil de proveedor puede ver la razón por la cual un producto fue rechazado.

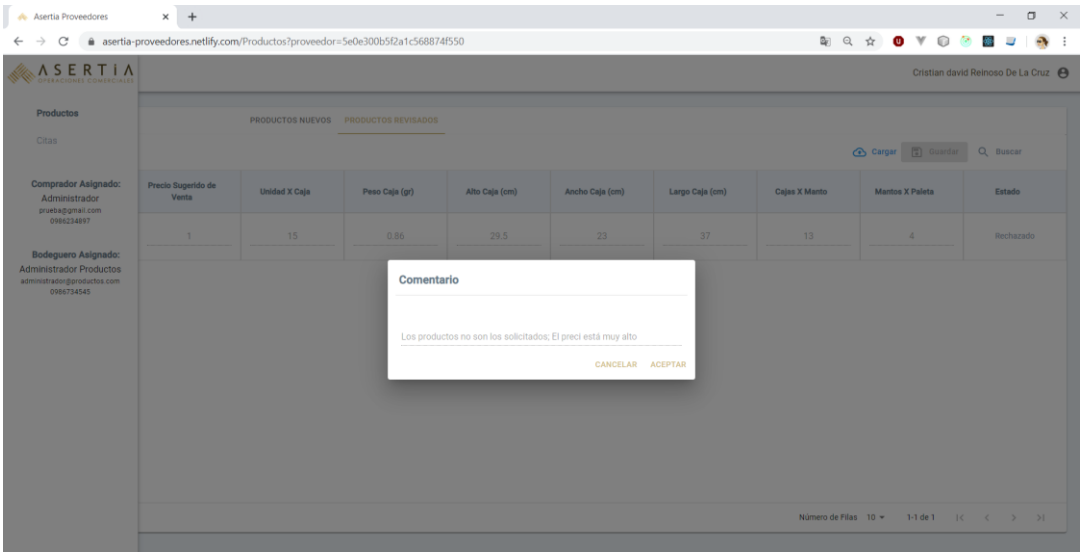



*Ilustración 50 Prueba del sistema. Requerimiento 8 del Sprint 2*

**Requerimiento 9:** Implementar un calendario en el cual se pueda ver las citas disponibles del comprador asignado a un proveedor.

**Criterio de aceptación:** El usuario con perfil de Proveedor podrá visualizar en un calendario las citas disponibles de su comprador asignado.

**Descripción de prueba:** En la siguiente imagen se demuestra que el usuario con perfil de proveedor puede visualizar un calendario con la información de las citas disponibles.

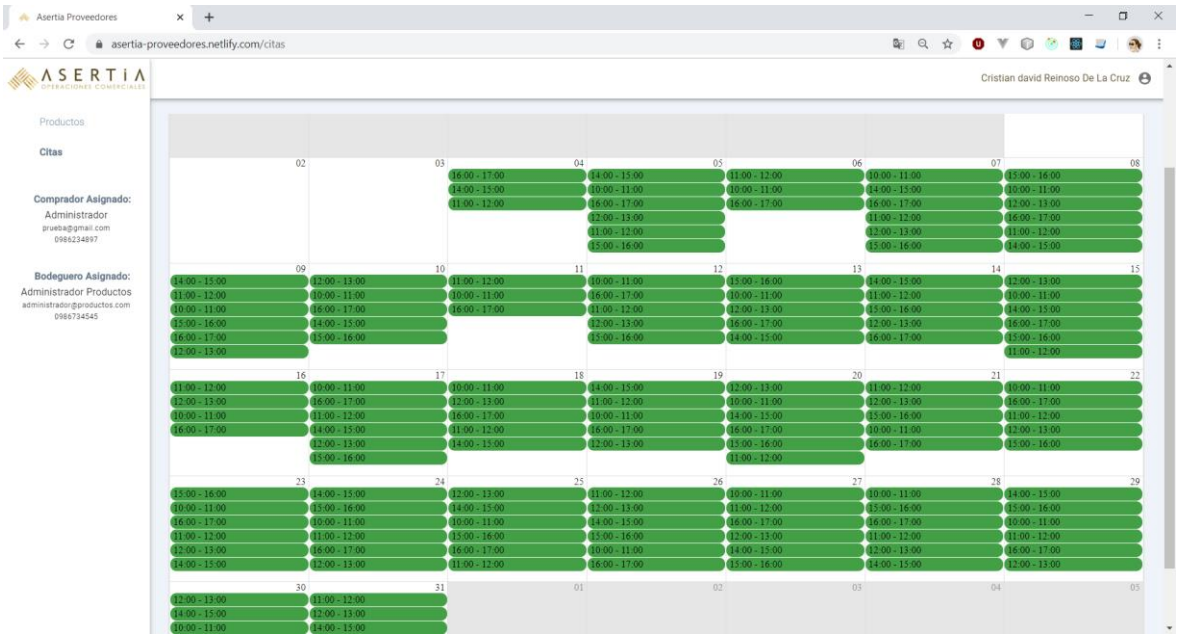



*Ilustración 51 Prueba del sistema. Requerimiento 9 del Sprint 2*

**Requerimiento 10:** Enviar un correo electrónico al proveedor cuando su comprador asignado cancele una cita ya agendada.

**Criterio de aceptación:** El usuario con perfil de Proveedor recibirá un correo electrónico cuando un comprador cancele una cita agendada.

**Descripción de prueba:** A continuación, se muestra un ejemplo del correo electrónico que se envía cuando un comprador cancela una cita.




*Ilustración 52 Prueba del sistema. Requerimiento 10 del Sprint 2*

### 3.6.3 Pruebas Sprint 3

**Requerimiento 1:** En la pantalla de citas de un comprador, implementar un dialogo que permita agregar una cita**.**

**Criterio de aceptación:** El usuario con perfil de Comprador podrá agregar una cita.

**Descripción de prueba:** En la siguiente imagen se demuestra que el usuario con perfil de comprador puede agregar una nueva cita.

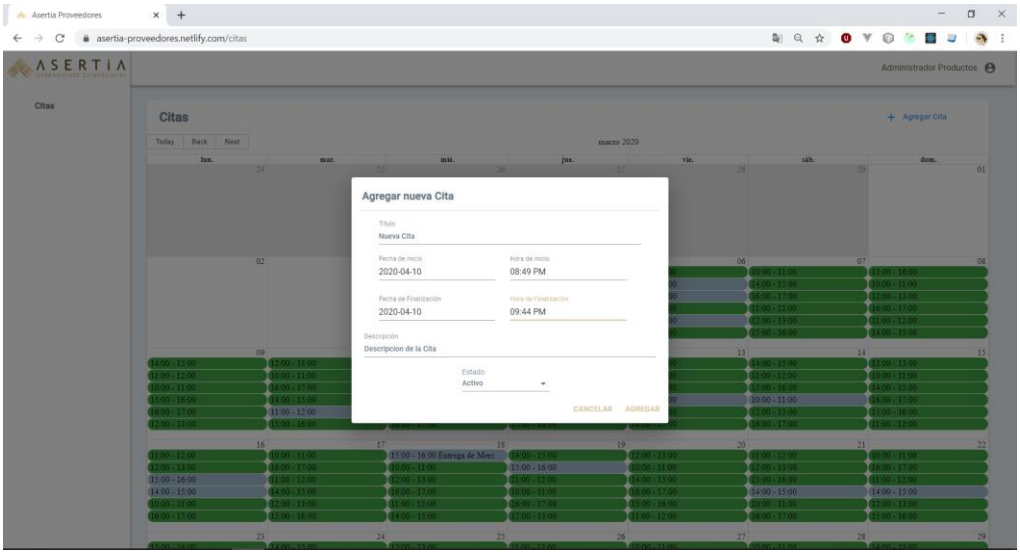



*Ilustración 53 Prueba del sistema. Requerimiento 1 del Sprint 3*

**Requerimiento 2:** Agregar citas desde las 10 de la mañana hasta las 5 de la tarda a todos los compradores con una duración de 1 hora por cada cita, exceptuando la hora de almuerzo de 1 de la tarde a 2 de la tarde.

**Criterio de aceptación:** Los compradores por defecto tendrán citas creadas de lunes a viernes desde las 10 am hasta las 5 pm con tiempo de duración de una hora.

**Descripción de prueba:** En la siguiente imagen se evidencia que un usuario con el perfil de comprador tiene 6 citas por día creadas por defecto.

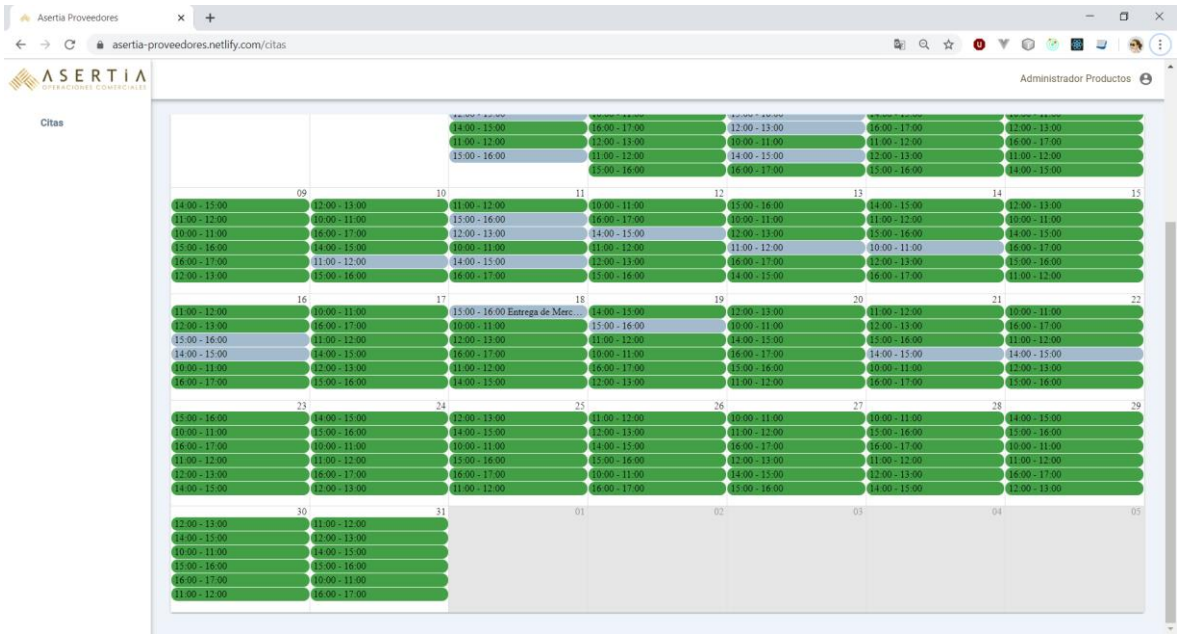



*Ilustración 54  Prueba del sistema. Requerimiento 2 del Sprint 3*

**Requerimiento 3:** Enviar un correo electrónico al comprador cuando un proveedor agende una cita.

**Criterio de aceptación:** El usuario con perfil de Comprador recibirá un correo electrónico cuando un proveedor agende una cita.

**Descripción de prueba:** A continuación, se muestra un ejemplo del correo electrónico que se envía cuando un proveedor agenda una cita.




*Ilustración 55  Prueba del sistema. Requerimiento 3 del Sprint 3*

**Requerimiento 4:** Enviar un correo electrónico al comprador cuando un proveedor cancele una cita.

**Criterio de aceptación:** El usuario con perfil de Comprador recibirá un correo electrónico cuando un proveedor cancele una cita.

**Descripción de prueba:** A continuación, se muestra un ejemplo del correo electrónico que se envía cuando un proveedor cancela una cita.




*Ilustración 56  Prueba del sistema. Requerimiento 4 del Sprint 3*

**Requerimiento 5:** Implementar una tabla en la cual se pueda ver los productos enviados por un proveedor con dos iconos al final de cada producto que permitan aprobar o rechazar el producto.

**Criterio de aceptación:** El usuario con perfil de Comprador podrá aprobar o rechazar productos.

**Descripción de prueba:** En la siguiente imagen se muestra que un comprador podrá aprobar o cancelar un producto mediante los iconos ubicados al final de cada fila.

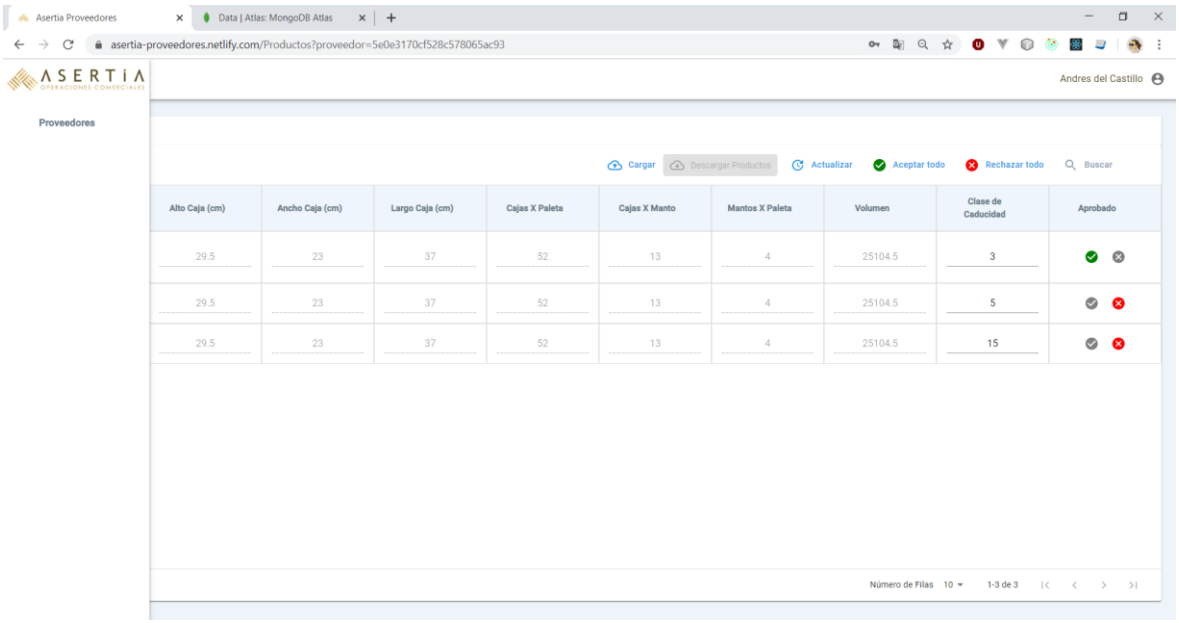



*Ilustración 57  Prueba del sistema. Requerimiento 5 del Sprint 3*

**Requerimiento 6:** Implementar, en la misma tabla de productos, entradas de texto donde su pueda editar de manera similar a una tabla de Excel, la información de los productos.

**Criterio de aceptación:** El usuario con perfil de Comprador podrá editar en la tabla la información de los productos enviados por el proveedor.

**Descripción de prueba:** En la siguiente imagen se demuestra que el usuario con perfil de comprador puede editar la información de los productos en la misma tabla.

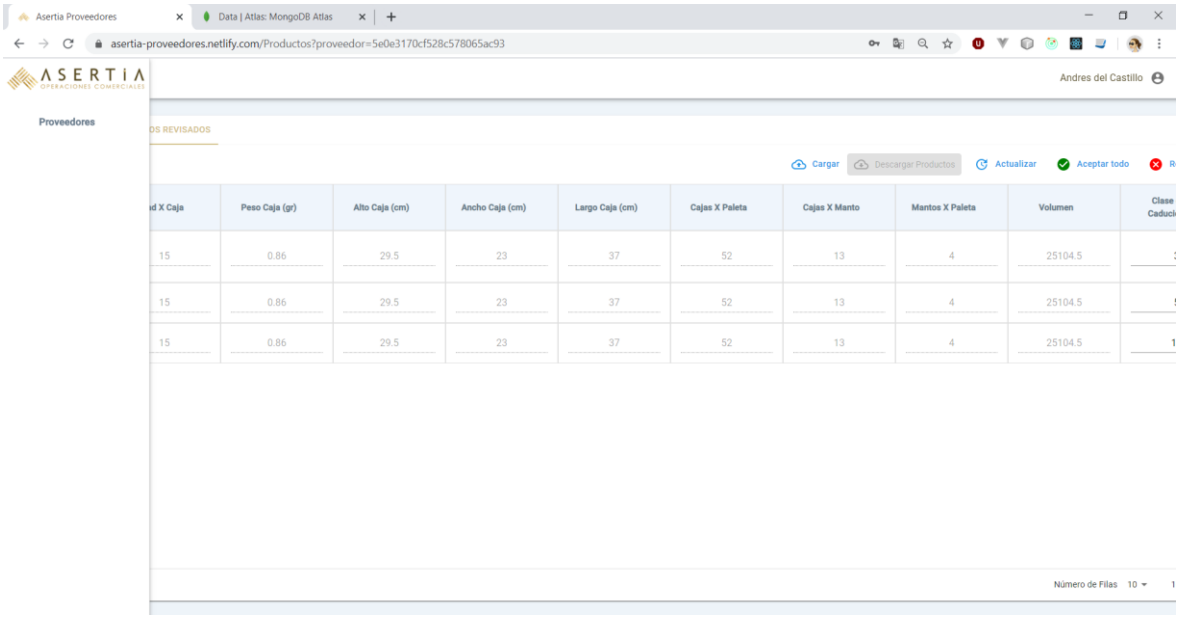



*Ilustración 58  Prueba del sistema. Requerimiento 6 del Sprint 3*

**Requerimiento 7:** Implementar una barra de búsqueda en la tabla de productos que permita buscar un producto por el código interno del producto o por el nombre del producto o por el código de producto proveedor o por la categoría del producto o por la subcategoría del producto o por la familia del producto.

**Criterio de aceptación:** El usuario con perfil de Comprador podrá buscar un producto por el código interno del producto o por el nombre del producto o por el código de producto proveedor o por la categoría del producto o por la subcategoría del producto o por la familia del producto.

**Descripción de prueba:** En la siguiente imagen se demuestra que el usuario con el perfil de comprador puede buscar un producto mediante una barra de búsqueda.

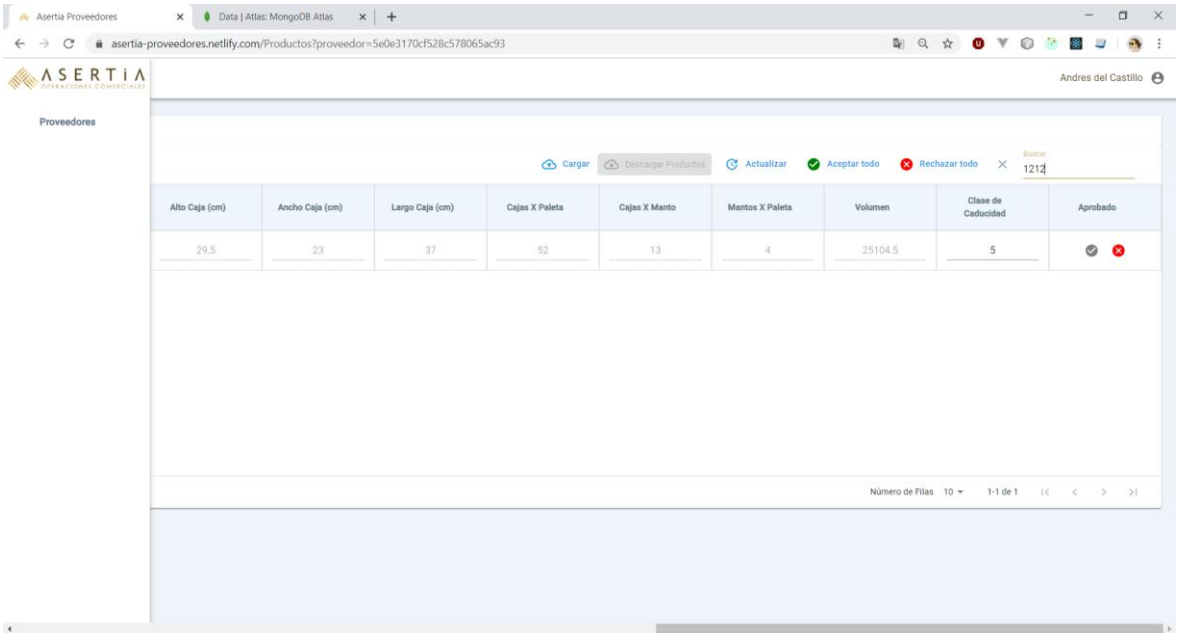



*Ilustración 59  Prueba del sistema. Requerimiento 7 del Sprint 3*

**Requerimiento 8:** Implementar un ordenamiento en la tabla de productos por: código interno, código producto del proveedor, código de barras, nombre, categoría subcategoría, familia, rentabilidad, costo con IVA, precio marcado, precio sugerido, unidad por caja, peso por caja, alto caja, ancho caja, largo caja, cajas por paleta, cajas por manto, manto por paleta, volumen, clase de caducidad y estado.

**Criterio de aceptación:** El usuario con el perfil de Compradores podrá ordenar productos por su código interno, código producto del proveedor, código de barras, nombre, categoría subcategoría, familia, rentabilidad, costo con IVA, precio marcado, precio sugerido, unidad por caja, peso por caja, alto caja, ancho caja, largo caja, cajas por paleta, cajas por manto, manto por paleta, volumen, clase de caducidad y estado.

**Descripción de prueba:** En la siguiente imagen se demuestra que el usuario con el perfil de comprador puede ordenar los productos presionando en el nombre de la columna.

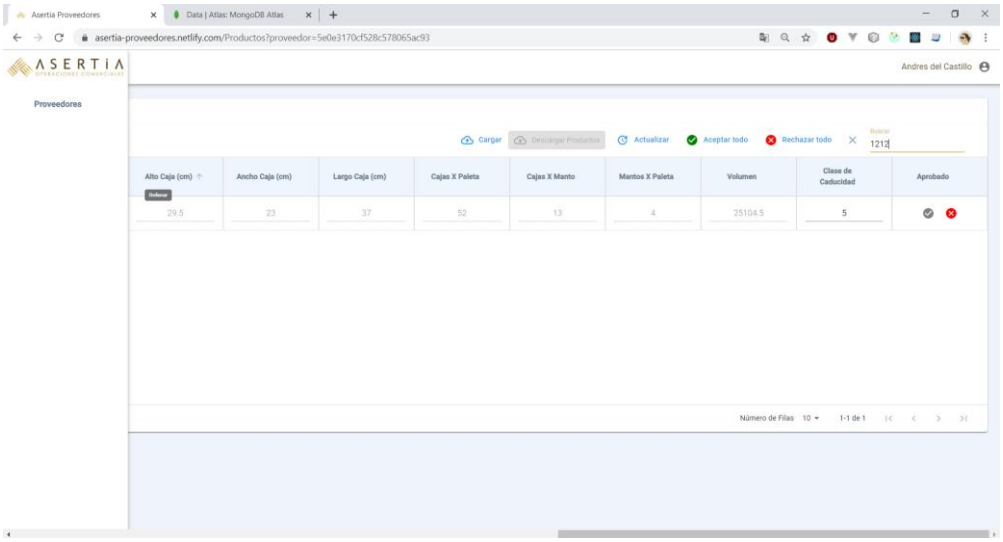



*Ilustración 60  Prueba del sistema. Requerimiento 8 del Sprint 3*

**Requerimiento 9:** Implementar dos pestañas en la pantalla de productos en los cuales se pueda ver una tabla con los productos revisados y productos no revisados

**Criterio de aceptación:** El usuario con perfil comprador podrá ver los productos revisados y productos no revisados.

**Descripción de prueba:** En la siguiente imagen se demuestra que el usuario con perfil de comprador puede visualizar los productos revisados y no revisados mediante dos pestañas ubicadas en la parte superior de la tabla

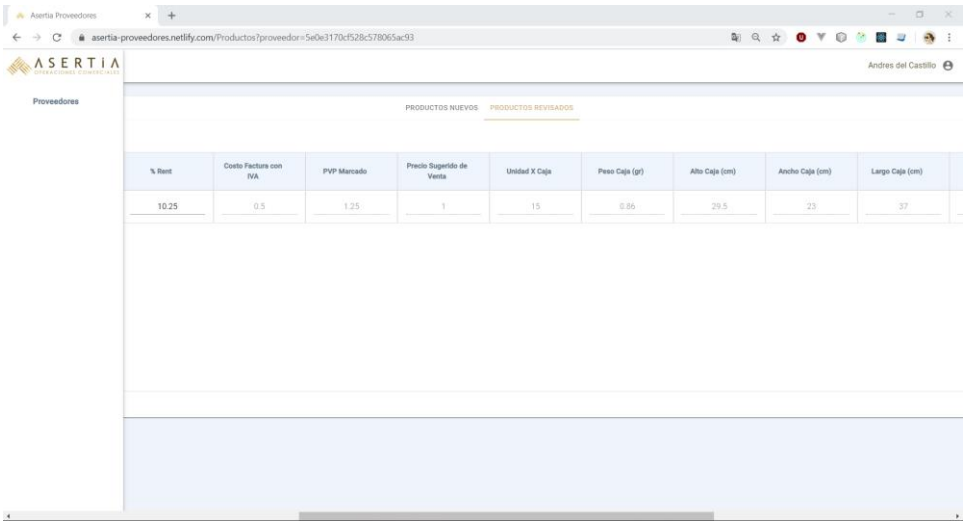



*Ilustración 61  Prueba del sistema. Requerimiento 9 del Sprint 3*

**Requerimiento 10:** Implementar un botón que permita aprobar todos los productos y otro botón que permita rechazar todos los productos.

**Criterio de aceptación:** El usuario con perfil de Comprador puede aprobar todos los productos o rechazar todos los productos.

**Descripción de prueba:** En la siguiente imagen se demuestra que el usuario con perfil de comprador puede aprobar todos los productos o rechazar todos los productos mediante los botones ubicados en la parte superior derecha de la tabla

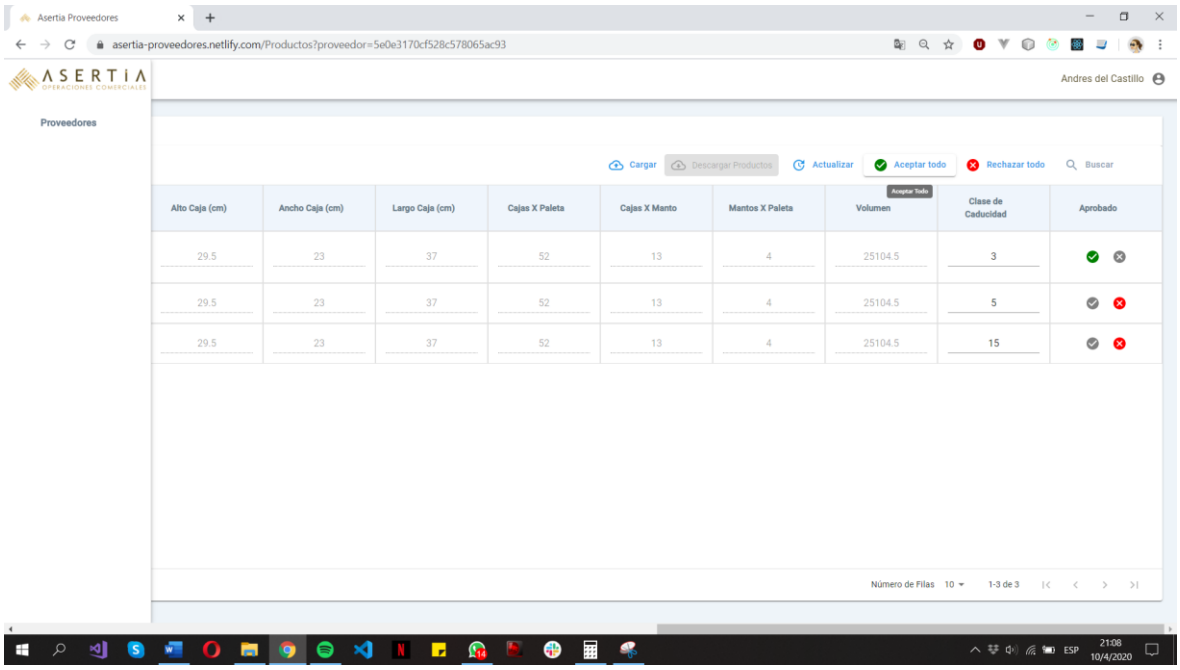



*Ilustración 62 Prueba del sistema. Requerimiento 10 del Sprint 3*

**Requerimiento 11:** Implementar un botón que permita descargarse los productos aprobados en un archivo en formato CSV con la información de los productos enviada por los proveedores más la información interna de la empresa.

**Criterio de aceptación:** El usuario con perfil Comprador podrá descargarse un archivo en formato CSV con la información de los productos aprobados.

**Descripción de prueba:** En la siguiente imagen se demuestra que el usuario con perfil de comprador puede descargaren formato CSV la información de los productos aprobados.

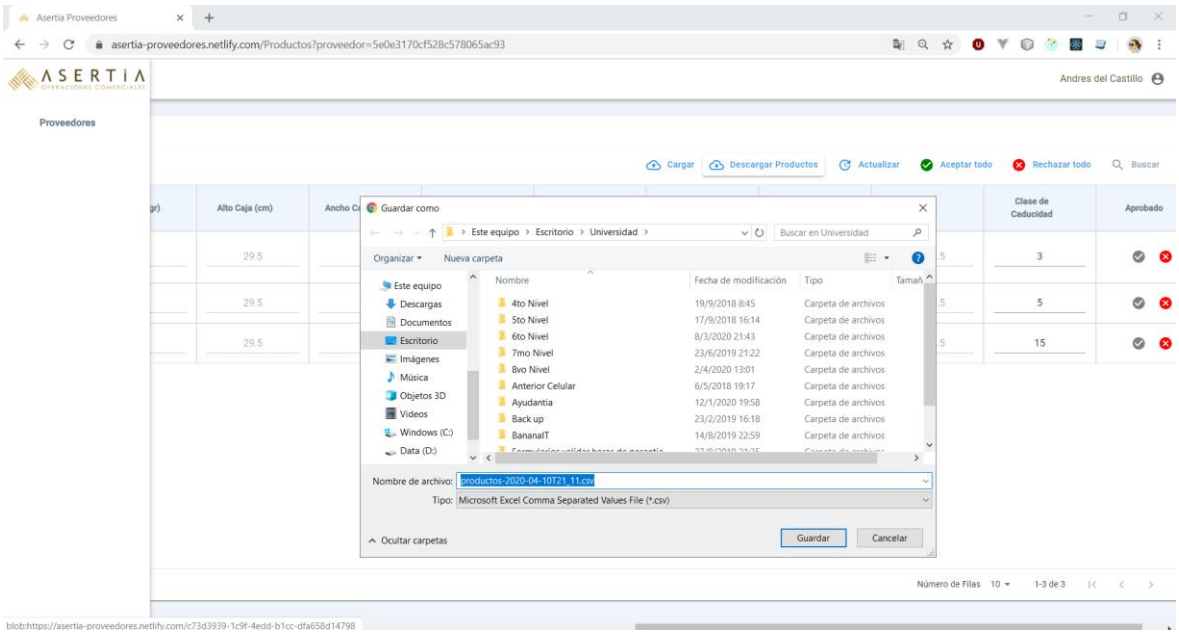



*Ilustración 63 Prueba del sistema. Requerimiento 11 del Sprint 3*

**Requerimiento 12:** Implementar un icono para editar en la pantalla de proveedores que despliegue un dialogo que permita editar la información de un proveedor

**Criterio de aceptación:** El usuario con perfil de Comprador podrá editar la información de los proveedores que tiene asignado.

**Descripción de prueba:** En la siguiente pantalla se demuestra que el usuario con perfil de comprador puede editar la información de un proveedor mediante un dialogo.

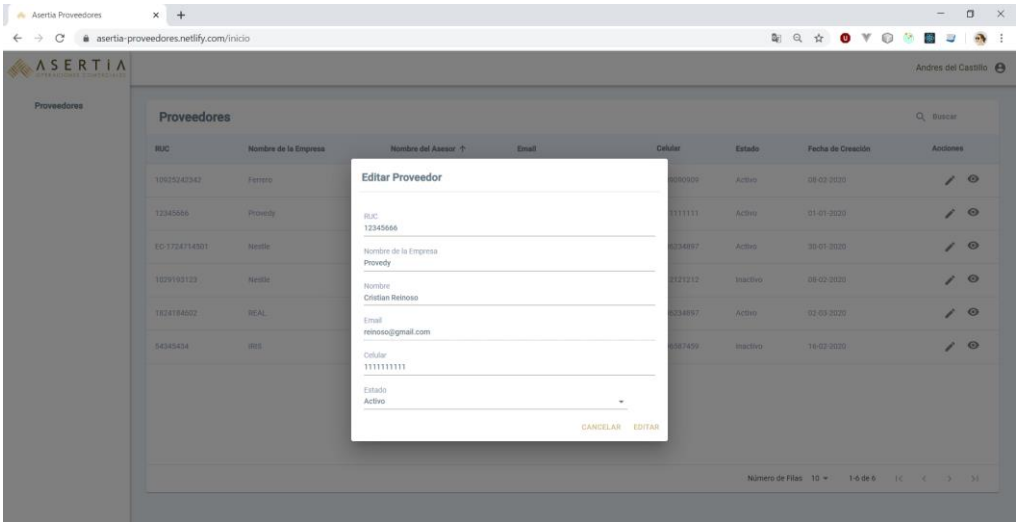



*Ilustración 64  Prueba del sistema. Requerimiento 12 del Sprint 3*

**Requerimiento 13:** Implementare un componente de selección donde el comprador pueda activar o desactivar a un proveedor.

**Criterio de aceptación:** El usuario con perfil de Comprador podrá activar o desactivar a un proveedor.

**Descripción de prueba:** En la siguiente imagen se demuestra que el usuario con perfil de comprador puede activar o desactivar a un proveedor.

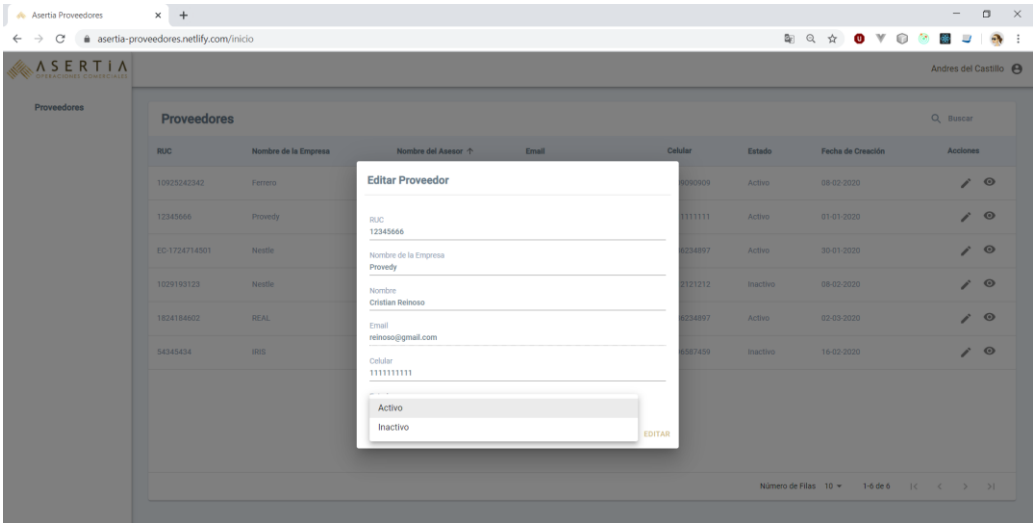



*Ilustración 65  Prueba del sistema. Requerimiento 13 del Sprint 3*

### 3.6.4 Integración del Sistema

Después de que el sistema web aprobó todas las pruebas exitosamente, se realizó la entrega formal del producto a Juan Francisco De La Cruz, representante de Asertia, quien es el encargado de socializar el nuevo sistema web con el departamento de compras y proveedores de la empresa.

El sistema web desarrollado en el presente plan de disertación fue llamado “Asertia Proveedores”, el cual está alojado en un servidor en la nueve y se puede acceder mediante cualquier dispositivo móvil que cuente con internet y un navegador web. El dominio por el cual se puede acceder al sistema web es: https://asertia-proveedores.netlify.com/

# 4 CONCLUSIONES Y RECOMENDACIONES

## 4.1 CONCLUSIONES

- Se desarrollo el sistema web para el plan de disertación “Análisis, diseño e implementación de una aplicación web para la gestión de productos y citas de proveedores en una empresa de distribución de productos masivos” para la empresa Asertia con el fin de automatizar procesos y brindar una herramienta que permita reducir el tiempo de revisión de productos y mejore la gestión de citas entre proveedores y compradores.
- Para desarrollar un sistema es fundamental comprender las necesidades de los usuarios, así como los requerimientos del sistema ya que estos son el pilar del sistema. En este trabajo de disertación los requerimientos del sistema fueron generados a partir de las necesidades de los próximos usuarios del sistema, es decir, de las historias de usuario.
- El prototipo del sistema web fue fundamental para la construcción del sistema web ya que le dio una idea al cliente de cómo será la apariencia de la aplicación junto a sus funcionalidades. Además, fue utilizado como molde al momento de desarrollar el sistema web.
- Las herramientas con las cuales se desarrollaron el presente sistema web fueron seleccionadas tomando en cuenta la funcionalidad que tendrá el mismo. Buscando siempre las herramientas que faciliten y optimicen el desarrollo del sistema web.
- Tomando en cuenta que la base de datos del presente sistema es no relacional, el modelo de base de datos fue construido basado en el contexto del negocio y los requerimientos del sistema.

- Gracias al diseño del sistema web existe la posibilidad de crear varias aplicaciones enfocadas para otros dispositivos, como por ejemplo aplicaciones móviles, debido a que la transferencia de datos entre el cliente y el servidor se realiza mediante API REST. Siendo la escalabilidad una ventaja del presente sistema web.
- Gracias a los estándares de desarrollo que se utilizaron para construir el presente sistema web, podemos concluir que cualquier otro desarrollador web, con conocimiento de las herramientas utilizadas, podrá entender y manipular fácilmente el código de este sistema web.
- Debido a que las pruebas del sistema se realizaron con datos reales históricos los cuales fueron proporcionados por los gestores principales de los procesos correspondientes y tomando en cuenta el diseño del sistema y los estándares de desarrollo del sistema podemos concluir que el sistema tendrá un comportamiento exitoso en un ambiente de producción.
- Mediante las pruebas realizadas, podemos concluir que el sistema web cumple con todos los criterios de aceptación establecidos en los requerimientos del sistema.

## 4.2 RECOMENDACIONES

- En caso de realizar un nuevo incremento o funcionalidad en el sistema web, se recomienda mantener los mismos estándares de desarrollo.
- Se recomienda tomar en cuenta que el espacio de almacenamiento de datos gratuito es de 512 MB, por lo que en caso de que se exceda este espacio se deberá pagar para ampliar el espacio de almacenamiento en la nube.

- En caso de realizar algún cambio en el sistema actual, se recomienda realizarlo en un ambiente de desarrollo, en el cual deberá ser probado los nuevos cambios para posteriormente ser desplegado en producción.
- Para hacer uso de los servicios web que utiliza este sistema web, se recomienda tomar en cuenta los aspectos técnicos que necesita cada servicio web para trabajar de forma exitosa.

# 5 Bibliografía


Apser. (14 de 07 de 2015). *Apser*. (Apser.es) Obtenido de Apser: http://www.apser.es/blog/2015/07/14/que-son-las-bases-de-datos-definicion-y-tipos/

Avison, D. E., & Fiztgerald, G. (1998). *Information Systems Development: Methodologies, Techniques, and Tools.* New York City: McGraw-Hill Higher Education.

Castejón, J. S. (2004). Arquitectura y diseño de sistemas web modernos. *Revista de Ingeniería Informática del CIIRM*, 6.

Edward, H., Teresa, G., Peñaherrera, A. O., & Quiña, G. N. (2019). Desarrollo backend para aplicaciones web, Servicios Web Restful: Node.js vs Spring Boot. *Revista Ibérica de Sistemas e Tecnologias de Informação*, 309-321.

ESLint. (2019). *ESLint.org*. Obtenido de https://eslint.org/

Express.js. (2019). *Express.js*. Obtenido de https://expressjs.com/es/

Gracia del Busto, H. y. (2012). Bases de datos NoSQL. *Telematica*, 21-33.

Hamad, H., Saad, M., & Abed, R. (2010). Performance Evaluation of RESTful Web. *Int. Arab J. e-Technol*, 1(3), 72–78.

Herron, D. (2016). *Node.js Web Development.* Birmingham: Packt Publishing.

Node.js, F. (26 de 8 de 2018). *About Node.js®.* Obtenido de https://nodejs.org/es/about/:

Pai, M, V., Asnika, S., & R, S. (2004). RESTful Web Services. International Journal of Advanced Information Science and Technology. *IJAIST*, 24(24), 46–50.

Pressman, R. (2010). *Ingeniería del software: un enfoque práctico.* McGraw-Hill Interamericana.

React.Js. (14 de Octubre de 2018). *React.Js*. Obtenido de https://reactjs.org/tutorial/tutorial.html#what-is-react

Redux. (s.f.). *Redux*. Obtenido de https://redux.js.org/

Sutherland, K. S. (2016). *La Guía Definitiva de Scrum.*

w3schools.com. (26 de 28 de 2018). *w3schools.com*. Obtenido de Node.js Tutorial: https://www.w3schools.com/nodejs/