

Universidad Nacional del Nordeste



Facultad de Ciencias Exactas y Naturales y Agrimensura

Cátedra: Taller de Programación l Año: 2021

Trabajo Proyecto Integrador

Alumno: Nuñez Luis Alberto

L.U. N°: 49675

D.N.I. N°: 95290429

***1.* Introducción**

El objetivo de desarrollo de una página enfocada a las ventas de instrumentos musicales . La aplicación a desarrollar, HOUSE MUSIC ofrecerá la funcionalidades como las que disponen las páginas de ventas en la red, dispondrá de un catálogo de instrumentos donde se podrá visualizar sus características más importantes y por supuesto tendremos la opción de poder comprarlos.

Para poder facilitar la localización de nuestros instrumentos, se dispondrá de una pequeña opción de búsqueda por palabras claves.

Dependiendo del tipo de usuario se dispondrá de determinadas funcionalidades, tendremos tres tipos de usuarios.

**El usuario anónimo** será el que menos funcionalidades tenga, podrá navegar por nuestra web, solo podrá ver los instrumentos, pero no podrá añadirlos al carrito y efectuar la finalización de la compra hasta que no se registre o se identifique, una vez realizada cualquiera de estas operaciones se convertirá en **usuario registrado** el cual ya podrá elegir los productos y finalizar la compra. El usuario con más privilegios será el **usuario administrador**, que además de disponer de las funcionalidades anteriores podrá gestionar el catálogo de productos, administrar clientes y categorías. En resumen se trata de desarrollar una aplicación para facilitar la venta de instrumentos así como ayudar a su gestión, de una forma sencilla y clara para los usuarios y el administrador de la aplicación.

Los objetivos concretos consistirán en:

* Mostrar un catálogo de los instrumentos a nuestros posibles clientes.
* Permitir la compra de los instrumentos que aparezcan en nuestra página.
* Facilitar el mantenimiento de dicho catálogo.
* Los resultados y productos que se piensan obtener:
* Facilitar la gestión tanto del material como de los usuarios.

El método para guiar el desarrollo del proyecto se basará en las siguientes etapas:

* La especificación de requisitos basada en las necesidades del cliente.
* La definición de la arquitectura basada en tres capas.
* La codificación de un prototipo de la página.
* La realización de pruebas técnicas del sitio Web.

Los recursos disponibles consistirán en:

* Un entorno de diseño Web basado en herramientas convencionales. – (Por ejemplo, XAMPP) el cual incluye (PHP, PHP5, MySQL y Apache.)  Un servidor Web.
* Un plazo de dos meses.
* Un alumno con dedicación completa.

# 2.Especificación de requisitos

Nuestra memoria correspondiente al desarrollo de una página Wep de ventas de instrumentos musicales, se va a desarrollar en base al estándar de especificación de requisitos de software IEEE 830-98.

**2.1. Introducción**.

En este apartado vamos a pasar a describir, el propósito, su ámbito, definiciones, acrónimos y abreviaturas, las referencias de la especificación y la visión global de nuestro proyecto.

**2.1.1. Propósito**. La ERS (especificación de requisitos) nos servirá de base para desarrollar nuestra página web en forma de una tienda virtual de venta de instrumentos musicales, detallar los requisitos de diseño de interfaz, contenidos y funcionalidad de la misma. Así como la especificación de los objetivos que queremos conseguir.

**2.1.2. Ámbito.**

Nuestro proyecto consistirá en la creación de una tienda virtual enfocada a la venta de instrumentos musicales, la cual estará dirigida hacia ***tipo de usuarios***.

En nuestra web HOUSE MUSIC se podrá navegar dentro de un catálogo de productos, de forma libre sin necesidad de que el usuario se registre, estas consultas se harán a través del catálogo de producto o a través de una búsqueda.

Dentro del catálogo se encontrarán todos los instrumentos disponibles, será el administrador el responsable de realizar la gestión de este material así como su correcta organización.

Todos los productos constarán de una descripción, imagen, precio y se podrá añadir a nuestro carrito de la compra, para efectuar la compra de cualquier producto añadido a nuestro carrito, el usuario deberá darse de alta en nuestra base de datos o si ya lo está, identificarse.

La gestión de nuestra Web será realizada por un usuario especial, el administrador, en el cual recaerá la responsabilidad de realizar el mantenimiento del catálogo de productos, altas y bajas de material, así como de gestionar pedidos y usuarios.

**2.1.3. Definiciones, acrónimos y abreviaturas.**

**Tienda virtual**: Esté término hace referencia a la aplicación informática capaz de ofrecer acceso a productos donde estos se pueden comprar y pagar por mediación de cualquier medio electrónico a

través de la red. Donde se suelen dar servicios muy similares a los de una tienda Física.

***Login*:** Es el nombre con el que se identifica a un usuario, que con anterioridad ha realizado un proceso de registro, suele formarse con un código alfanumérico de varios dígitos. Dependiendo de la página se le pueden asignar distintos roles y accesos.

***Password*:** palabra de paso o clave de acceso, permite el acceso a una persona, se le pueden asignar permisos. Compuesta al igual que el *login* por un conjunto alfanumérico de varios dígitos.

**Usuario anónimo**: Usuario que visita el portal y del cual no se tiene información de registro. **Usuario registrado**: Usuario que ha realizado el proceso de registro y por lo tanto se dispone de información personal para identificarlo y personalizar su visita a la web.

**Servidor**: Máquina que se encarga de ejecutar la aplicación para que los usuarios, clientes, puedan acceder a recursos y páginas.

**Cliente:** Cualquier elemento de un sistema de información que requiere un servicio mediante el envío de solicitudes al servidor.

**Navegador**: Aplicación para visualizar documentos WWW y navegar por Internet. En su forma más básica son aplicaciones hipertexto que facilitan la navegación por los servidores de navegación de Internet.

**Protocolo:** Conjunto de reglas y/o procedimientos para la transmisión de datos que ha de ser observado por los dos extremos de un proceso comunicacional.

**Acrónimos.**

**HTML:** *HyperText Markup Language*. Lenguaje de marcado de hipertexto, es el lenguaje estándar para describir el contenido y apariencia de las páginas en la web.

**Web:** Por esté término se conoce a WWW (World *Wide Web*), le red de redes, es básicamente un medio de comunicación de texto, gráficos y otros objetos multimedia a través de internet. **HTTP:** *Hyper Text Transfer Protocol*, protocolo de transferencia de hipertexto, es el método más común de intercambio de información en la web.

**PHP:** Es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas Web dinámicas, usado en el lado del servidor.

**MySQL:** Es un sistema de gestión de base de datos relacional, multihilo y multiusuario.

**SQL:** Lenguaje de consulta estructurado, *structured query language*, es un lenguaje declarativo de acceso a base de datos relacionales que nos permite especificar diversos tipos de operaciones en estas.

**XAMPP:** Usado para describir un sistema de infraestructuras de internet que usa, *Windows* como sistema operativo, Apache como servidor Web, MySQL como gestor de base de datos y PHP como lenguaje de programación.

**CSS:** *Cascading Style Sheets*, hojas de estilo en cascada, es un lenguaje usado para definir la presentación de un documento escrito en HTML o XML.

**JavaScript:** es un lenguaje de *scripting* basado en objetos no tipeado y liviano, utilizado para acceder a objetos en aplicaciones, no requiere compilación.

**IEEE:** *Institute of Ekectrical and Electronics Engineers*, es una asociación técnico-profesional mundial dedicada a la estandarización.

**2.1.4. Referencias.**

WORLD WIDE WEB CONSORTIUM (W3C). Documentos, guías y estándares. [Consulta: fecha].

Disponible en: http://www.w3c.es

WIKIPEDIA LA ENCICLOPEDIA LIBRE. Definiciones. [Consulta: fecha].Disponible en:

http://es.wikipedia.org

REAL ACADEMIA ESPAÑOLA. Ortografía. [Consultas: fecha]. Disponible en: http://rae.es

**2.1.5. Visión global.**

Vamos a centrar la especificación de requisitos en la descripción de la aplicación, sus características, sus restricciones generales, sus funciones, sus supuestos y dependencias, que son las que nos aportarán una mayor información del proyecto a desarrollar.

Cuando terminemos de desarrollar la descripción general, pasaremos a la descripción de los requisitos específicos de nuestra aplicación.

**2.2. Descripción general.**

En la descripción general se aborda la perspectiva del producto, los distintos usuarios que utilizarán la aplicación, restricciones, supuestos y dependencias de nuestro proyecto.

**2.2.1. Perspectiva del producto.**

Nuestra aplicación, realizará las funcionalidades normales que implementa cualquier tienda virtual, venta y búsqueda de productos, en nuestro caso instrumentos musicales, la gestión y administración tanto de productos como de usuarios y pedidos.

**2.2.2. Funciones del producto**.

Clasificaremos las funciones de nuestra aplicación en bloques.

**Funciones de búsqueda**. Cualquier usuario que acceda a nuestro portal web, este registrado o no, podrá realizar búsquedas de material en el catálogo. Cuando el usuario encuentre el producto se mostrará la imagen del mismo, con su precio, opción para visualizar sus detalles y añadir a al carrito de la compra.

**Funciones de compra**. Para poder finalizar la compra de un producto el usuario deberá identificarse o en el caso que no esté registrado realizar la operación de registro.

**Funciones de gestión**. Las funciones de gestión solo podrán llevarse a cabo por el administrador, será el encargado de realizar altas y bajas de productos así como de la edición o rectificación de estos. También gestionará el catálogo de productos, creando o eliminando según crea conveniente, además de poder eliminar usuarios, consultar pedidos y modificar su estado, así como listar productos, usuarios, categorías y pedidos.

**2.2.3. Características del usuario**.

Nuestro sistema constará de tres tipos de usuarios que son los que podrán interactuar con nuestra web: Usuario anónimo, usuario registrado y usuario administrador.

**Usuario anónimo**: Podrá realizar consultas al catálogo de productos y navegar de forma libre por la zona pública de la web. Ver productos, descripciones de productos…

**Usuario registrado:** Es aquel usuario que habrá efectuado su alta en el sistema y que por lo tanto sus datos están en nuestra base de datos. Estos usuarios podrán realizar las funciones de usuario anónimo y podrán efectuar finalizar las compras añadidas al carrito de la compra.

**Usuario administrador:** Este usuario será el encargado de realizar todas las funciones de gestión del sistema, modificaciones, listados, mantenimiento de la base de datos y comprobación del buen funcionamiento de la web.

**2.2.4. Restricciones generales**.

Las consideraciones de seguridad que existirán en nuestra aplicación serán la restricción de control y acceso a la información, ya que dependiendo de nuestro *login* y *password* dispondremos de unos privilegios para poder realizar determinadas funciones.

Será muy importante cumplir con la ley de protección de datos para así mantener la información de nuestra base de datos protegida ante accesos no deseados Esto será requisito indispensable en nuestro servidor que será donde se almacene toda la información delicada.

En el parte del cliente cualquier equipo sobremesa o portátil, con una configuración media baja, que soporte un navegador web estándar serían suficientes para poder ser utilizados, ya que los requisitos necesarios para visualizar nuestra aplicación son escasos.

Por el lado del servidor dependeremos del número de clientes que estén conectados de forma simultánea, ya que esto haría que nuestro servidor funcionara de forma inadecuada. La solución sería aumentar la potencia del servidor para evitar la saturación debida a la ejecución de mucho código, muchos accesos a nuestra base de datos…

Otro posible problema del lado del servidor, de fácil solución actualmente, es el ancho de banda para servir las páginas, solución contratar mayor ancho de banda.

Algo muy importante son las políticas de seguridad en el servidor, así evitaremos posibles intrusiones en el sistema. Es fundamental que la configuración de nuestro servidor Apache este optimizada al máximo. No dejar la configuración que viene por defecto cuando instalamos Apache, cambios de contraseña y número de puerto de acceso son fundamentalmente acciones que debemos realizar para optimizar la seguridad.

Las copias de seguridad de la base de datos, páginas del servidor y configuraciones del servidor, se deben realizar de forma periódica.

**2.2.5. Supuestos y dependencias.**

Nuestra página Wep HOUSE MUSIC, se podrá ejecutar en cualquier sistema operativo actual, basado en tecnología *Windows o Linux*. Ya que vamos a utilizar para su desarrollo una aplicación del tipo XAMPP, Windows Apache MySQL PHP.

De esta manera con una única aplicación nos facilitará el trabajo y configuración de todos sus componentes, evitaremos las instalaciones múltiples que pueden resultar complicadas y a veces incompatibles entre ellas.

Para poder efectuar las pruebas es necesario instalar Apache en la máquina que funcionará como servidor. En nuestro caso el servidor tendrá instalado *Windows 8*. El desarrollo de la aplicación se realizará con HTML y PHP, para su edición se utilizará Sublime Text. Para la gestión de la base de datos se utilizará PhpMyAdmin.

**2.3. Requisitos específicos.**

En este apartado se presentan los requisitos funcionales que deberán ser satisfechos por el sistema.

**2.3.1. Requisitos de interfaces externas.**

**Interfaces de usuario.**

En la interfaz de usuario de nuestra aplicación inicialmente se accederá a una página principal donde, se dispondrá de un menú navegacional donde se les presenta información de la página, la opción de realizar consultas, información de los empleados como gerente y la opción de ver el catálogo de productos disponibles . El apartado de carrito nos permitirá visualizar los productos que hemos ido añadiendo, esta opción nos dejará eliminar artículos o cambiar las unidades del mismo.

Dispondremos de un apartado que nos permitirá identificarnos, y en el caso de que el usuario no sea socio, le permitirá realizar su respectivo registro enviando un formulario y procediendo a su alta. Según los permisos de cada tipo de usuario, se podrá acceder a las distintas opciones de la aplicación.

Se dispondrá de una aplicación que nos facilitará la búsqueda de productos, introduciendo una palabra buscará los artículos que la incluyan.

Por último dispondremos de un logout para poder desconectarnos de la sesión.

**Interfaces *hardware*.**

Los clientes necesitarán una máquina que disponga de acceso a internet, actualmente cualquier equipo que soporte un navegador web, con una tarjeta de red, sería lo recomendado, o bien un modem.

La máquina que realizará la función de servidora dispondrá de un repositorio en cual almacenará toda la información del catálogo de productos además de la base de datos con la información de los usuarios registrados. Esta máquina necesitará una conexión a internet con una banda ancha mayor, para así poder asistir las peticiones de los clientes.

**Interfaces *software.***

Nuestra aplicación va a ser desarrollada bajo el sistema operativo *Windows 8*, nuestro *Windows* instalaremos nuestro sistema XAMPP, que incorporará Apache como servidor, la versión 5.6.40, MySQl como gestor de base de datos, el cual incluye una interfaz amigable llamada PHPMyAmin y como lenguaje de implementación PHP.

Para el desarrollo del interfaz web utilizaremos Sublime Text que nos ayudará con el HTML, las hojas de estilo CSS y con PHP.

Los clientes que accedan a nuestra web no será necesario que dispongan de un sistema operativo concreto, el navegador deberá estar actualizado a las últimas versiones para evitar visualizaciones erróneas o falta de *plug-ins*.

Nuestra aplicación va a ser desarrollada con una arquitectura de tres capas.

**Interfaces de comunicaciones.**

Los protocolos de comunicaciones van a ser TCP/IP entre los clientes y la web y la comunicación entre los navegadores y el servidor HTTP.

Los usuarios realizarán su comunicación a través de Internet y el administrador tendrá la posibilidad de trabajar vía Internet o en local.

**2.3.2. Requisitos funcionales**.

## Función de autentificación de usuario (funcionalidad del usuario y administrador)

**Introducción:** Mediante la introducción de usuario y contraseña, los usuarios registrados y el administrador accederán a la aplicación y a sus características asociadas.

**Entradas:** Tanto para el campo de usuario y contraseña se requerirán códigos alfanuméricos. **Proceso:** Se realizará una comprobación de la contraseña del usuario coincida dentro de nuestra base de datos, si su identificación es correcta se mostrará una pantalla de bienvenida.

En caso contrario aparecerá una pantalla con un aviso de acceso incorrecto y se dará la posibilidad de registro.

**Salida:** en el caso que la identificación sea correcta nos llevará a una pantalla con un mensaje de bienvenida pudiendo acceder a las opciones específicas del usuario correspondiente. En caso negativo nos aparecerá una pantalla de advertencia.

## Funciones de visualización de productos (funcionalidad de cualquier usuario)

**Introducción:** Una vez seleccionada una categoría nos aparecerá un listado de productos, el usuario selecciona uno de los productos de la lista para visualizarlos.

**Entradas:** Categoría y producto seleccionados.

**Proceso:** Nuestra aplicación abrirá el correspondiente vínculo que hace referencia a las características y al producto seleccionado dentro de la categoría, todas las características del producto serán recuperadas del correspondiente repositorio.

**Salida:** En la pantalla del cliente aparecerán todos los detalles del producto seleccionado.

## Funciones de registro de usuario (funcionalidad del usuario anónimo)

**Introducción:** El nuevo usuario tendrá acceso a un formulario donde deberá introducir sus datos para darse de alta en nuestra aplicación.

**Entradas:** Se introducirán en los campos del formulario tanto datos personales como nombre de usuario, contraseña y correo electrónico.

**Proceso:** La primera parte del registro realizará una comprobación de todos los campos del formulario a rellenar, que no sean espacios en blanco, que los campos no estén vacíos, comprobación del correo y clave, la segunda parte realizará la

Verificación del nombre de usuario, para evitar duplicidades, si existiera lanzaremos un mensaje de advertencia, para finalmente insertar una tupla, con estos datos en nuestra base de datos.

**Salida:** Un mensaje diciendo que la inserción se ha efectuado con éxito.

## Funciones de compra (funcionalidad del usuario registrado)

**Introducción:** Desde esta función se podrá realizar una compra de los productos seleccionados, siempre que el usuario esté registrado.

**Entradas:** El usuario selecciona una serie de productos del catálogo.

**Proceso:** Nuestro usuario podrá ir añadiendo productos a su carrito, será en el momento final de la compra cuando se verificará que el usuario este dado de alta, en caso que no sea así le enviará a una pantalla para que pueda hacerlo o bien le permita registrarse.

**Salida:** Aparecerá en la pantalla del cliente un mensaje de pedido realizado de forma correcta, asignando un número de pedido y factura a su compra por último terminará vaciando el carrito.

## Función de añadir nueva categoría (funcionalidad el administrador)

**Introducción:** Nuestro administrador a través de un formulario podrá introducir nuevas categorías dentro de nuestra aplicación.

**Entradas:** Una nueva línea con el nombre de la categoría.

**Proceso:** Una nueva entrada en la base de datos correspondiente con la nueva categoría.

**Salida:** Una nueva página con un mensaje de éxito.

## Función de borrar categoría (funcionalidad el administrador)

**Introducción:** Mediante un formulario nuestro administrador podrá eliminar categorías de nuestra aplicación.

**Entradas:** Debemos seleccionar la categoría a eliminar.

**Proceso:** Nuestro administrador deberá seleccionar una categoría a eliminar dentro del listado existente, para después confirmar pulsando el botón eliminar.

**Salida:** Mostraremos una página de éxito si la operación se ha podido llevar a cabo.

## Función listar categorías (funcionalidad el administrador)

**Introducción:** El administrador podrá efectuar una consulta de todas las categorías.

**Entradas:** Los valores para realizar la búsqueda.

**Proceso:** se efectuará una consulta en la tabla correspondiente a las categorías.

**Salida:** Se obtendrá en pantalla un listado de todas las categorías actuales.

## Función añadir nuevo producto (funcionalidad el administrador)

**Introducción:** Igual que en las anteriores opciones, el administrador a través de un formulario tendrá la opción de añadir un nuevo producto dentro de una categoría del catálogo existente.

**Entradas:** Todas las características del producto, nombre, marca, precio, descripción…

**Proceso:** Una vez introducidos todos los datos, realizaremos la inserción dentro de nuestra base de datos de productos.

**Salida:** Una página mostrando el éxito si se ha podido realizar la operación.

## Función borrar producto (funcionalidad el administrador)

**Introducción:** Nuestro administrador podrá efectuar el borrado de producto de nuestra aplicación.

**Entradas:** Deberemos seleccionar el producto a eliminar.

**Proceso:** Una vez seleccionada la opción de borrar producto, deberemos seleccionar el producto a eliminar del listado existente. Finalmente confirmaremos la operación con el botón de borrado. **Salida:** Nuevamente aparecerá una página con un mensaje de operación realizada de forma correcta si se ha podido llevar a cabo.

## Función listar productos (funcionalidad el administrador)

**Introducción:** El administrador podrá efectuar una consulta de todos los productos que estén dados de alta.

**Entradas:** Los valores para realizar la búsqueda.

**Proceso:** Se efectuará una consulta en la tabla de productos.

**Salida:** Se obtendrá en pantalla un listado de todos los productos que se dispongan en la tabla de productos.

## Función listar usuarios (funcionalidad el administrador)

**Introducción:** El administrador podrá efectuar una consulta de todos los usuarios que estén registrados.

**Entradas:** Los valores para realizar la búsqueda.

**Proceso:** Se efectuará una consulta en la tabla correspondiente a los usuarios registrados.

**Salida:** Se obtendrá en pantalla un listado de todos los usuarios registrados.

## Función eliminar usuario (funcionalidad el administrador)

**Introducción:** Desde esta opción podremos efectuar el borrado de un usuario registrado.

**Entradas:** Los datos correspondientes al usuario que queremos eliminar.

**Proceso:** Deberemos seleccionar del listado de socios el usuario que deseamos borrar, una vez seleccionado se eliminará la tupla de nuestra tabla correspondiente.

**Salida:** Como es habitual lanzaremos una página con un mensaje de éxito si se ha efectuado de forma correcta.

## Función listar pedidos (funcionalidad el administrador)

**Introducción:** Dentro de esta función se podrá obtener un listado de pedidos realizados.

**Entradas:** Los parámetros de búsqueda.

**Proceso:** Nuestra aplicación efectuará una consulta dentro de la tabla correspondiente. **Salida:** El listado con los pedidos realizados.

**2.3.3. Requisitos de rendimiento**.

Inicialmente no se han determinado requisitos específicos de eficiencia, aunque serían necesarios algunos requerimientos para que nuestra aplicación se convirtiera en un producto cuya finalidad fuera su uso a nivel profesional.

**2.3.4. Restricciones de diseño**.

Vamos a pasar a describir los estándares y las limitaciones de nuestro producto.

**Estándares cumplidos.**

Nuestro producto debe cumplir las especificaciones actuales de cualquier página web dinámica actual, tanto las de lenguaje HTML como las de los estándares seguidos por las hojas CSS. De esta forma tendremos la máxima compatibilidad con cualquiera de los navegadores modernos y los costos de mantenimiento de nuestra aplicación serán menores.

Deberemos cumplir con la actual ley de protección de datos, para asegurarnos de que la información que se dispone en nuestra base de datos no es filtrada para ningún uso ilícito y asegurar al usuario poder dar de baja la información de carácter personal.

**Limitaciones Hardware.**

Inicialmente las posibles limitaciones a nivel *hardware* residen en la conexión que se disponga en el lado del cliente, ya que actualmente es muy raro que algún equipo no cumpla las características mínimas para garantizar un funcionamiento correcto.

El sistema operativo utilizado tampoco debería ser ningún impedimento para la correcta visualización y funcionamiento de nuestra aplicación.

En cuanto al lado del servidor sus limitaciones vendrán dadas por la capacidad del mismo para satisfacer un número de accesos determinados de forma simultánea.

**2.3.5. Atributos.**

**Mantenimiento.**

El mantenimiento de nuestro sistema se realizará de una forma continuada por el administrador enviando sus peticiones al equipo de desarrolladores de la web así como los posibles problemas que puedan ir surgiendo. **Seguridad.**

Deberemos organizar un buen sistema de protección de información ya que en nuestra base de datos tendremos los datos personales de los usuarios que han efectuado su registro.

Teniendo en cuenta que nuestro sistema funcionará en la red, donde es más viable un posible problema de seguridad.

# Análisis

**3.1. Introducción.**

En este apartado vamos a abordar la fase de análisis, con ayuda de la metodología UML se desarrollará un modelo de la aplicación donde diferenciaremos a los actores que interactúan con los objetos del sistema mediante distintas relaciones.

Análisis por definición es el proceso de construcción de un modelo, o especificación detallada del problema del mundo real al que nos enfrentamos. Está desprovisto de consideraciones de diseño e implementación. Todo esto nos ayuda a crear un sistema robusto y mantenible.

La notación que se va a utilizar es la proporcionada por el estándar UML. En nuestro caso usaremos los diagramas de clases, casos de uso y diagramas de secuencia.

**3.2. Casos de uso.**

Un caso de uso es una técnica para la captura de requisitos potenciales de un nuevo sistema o una actualización de software.

Cada caso de uso proporciona uno o más escenarios que indican cómo debería interactuar el sistema con el usuario o con otro sistema para conseguir un objetivo específico. Normalmente, en los casos de usos se evita el empleo de jergas técnicas, prefiriendo en su lugar un lenguaje más cercano al usuario final

En otras palabras, un caso de uso es una secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal sobre el propio sistema. Los diagramas de casos de uso sirven para especificar la comunicación y el comportamiento de un sistema mediante su interacción con los usuarios y/u otros sistemas. O lo que es igual, un diagrama que muestra la relación entre los actores y los casos de uso en un sistema.

Una relación es una conexión entre los elementos del modelo, por ejemplo la especialización y la generalización son relaciones.

Los diagramas de casos de uso se utilizan para ilustrar los requerimientos del sistema al mostrar cómo reacciona a eventos que se producen en su ámbito o en él mismo.

Después de esta breve introducción de los casos de uso pasamos a mostrar cada uno de ellos. Comenzamos por mostrar los actores que intervienen en nuestro sistema, que como ya sabemos son usuario, usuario registrado y administrador. Estos dos últimos especializan del usuario.

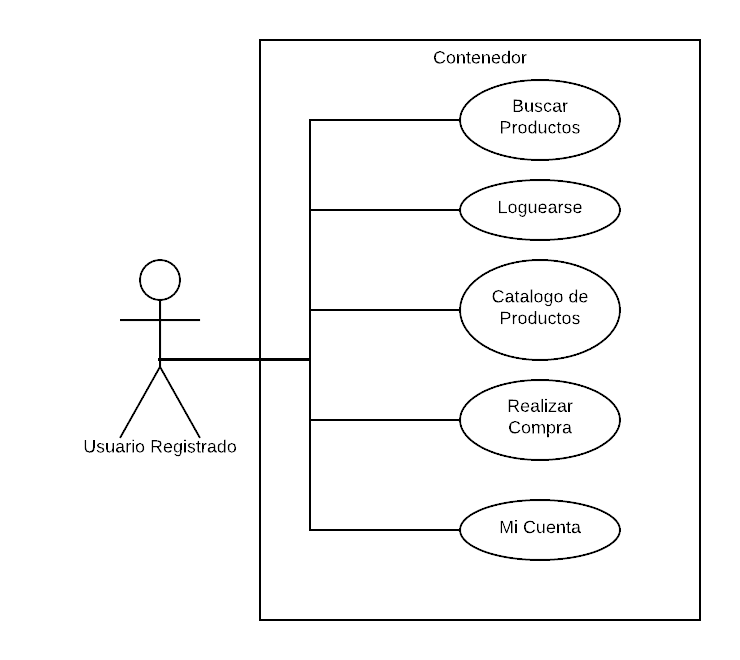
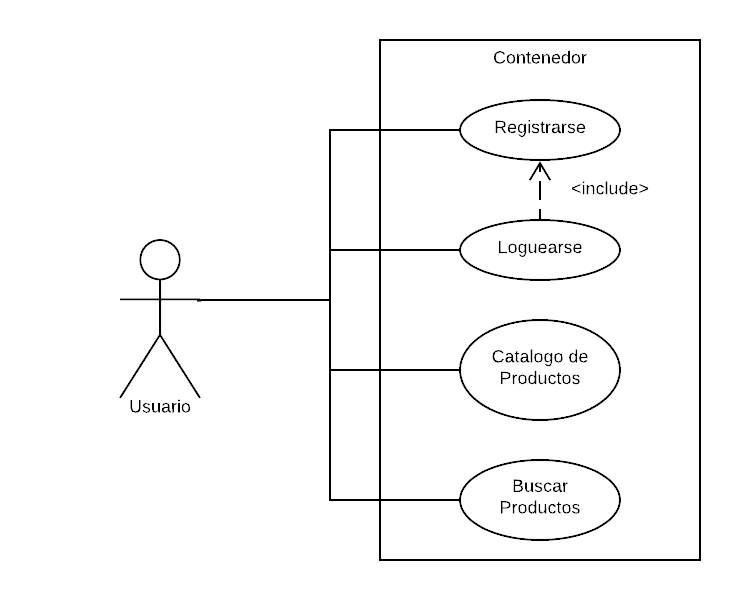
Ejemplo:

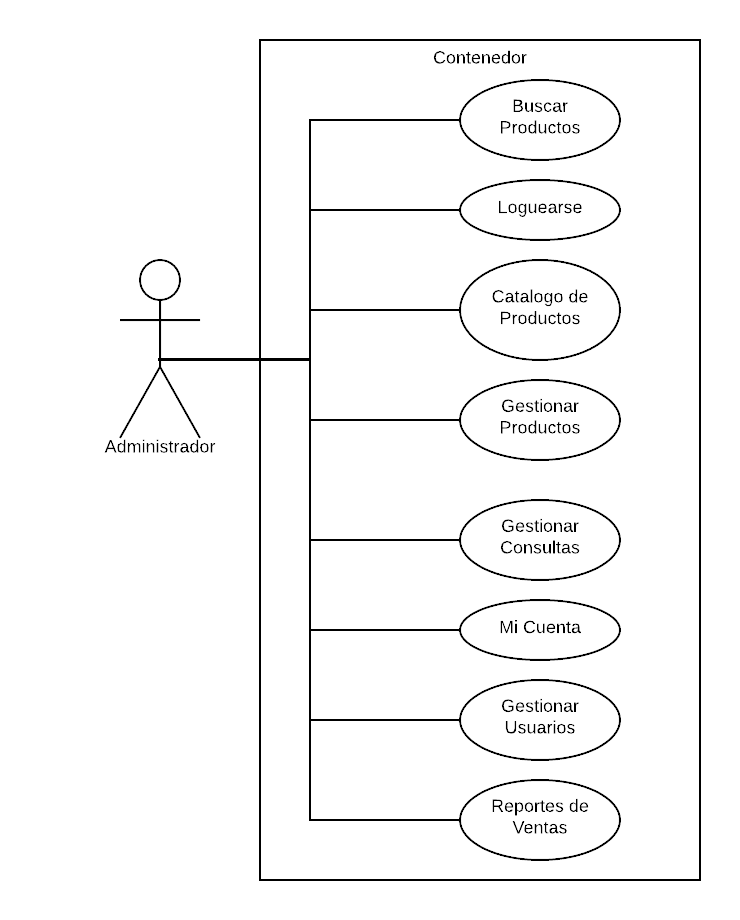
En los siguiente diagrama tenemos las funcionalidades de todos los actores, empezamos por el usuario, las funciones que podrá realizar serán, identificarse, registrarse, realizar búsqueda de productos y ver la información general de los productos del catálogo, es decir navegar a nivel general por la web.

Además el usuario puede proceder al registro, convirtiéndose en usuario registrado.

El usuario registrado además de poder realizar todo lo comentado anteriormente se le añadirá la funcionalidad de realizar las compras de los productos que crea oportuno.

Por último el administrador dispone de las funcionalidades de gestión de productos, , gestión de usuarios, gestión de las consultas, y por ultimo podrá ver los reportes de ventas.





**3.3 Diagramas de secuencia.**

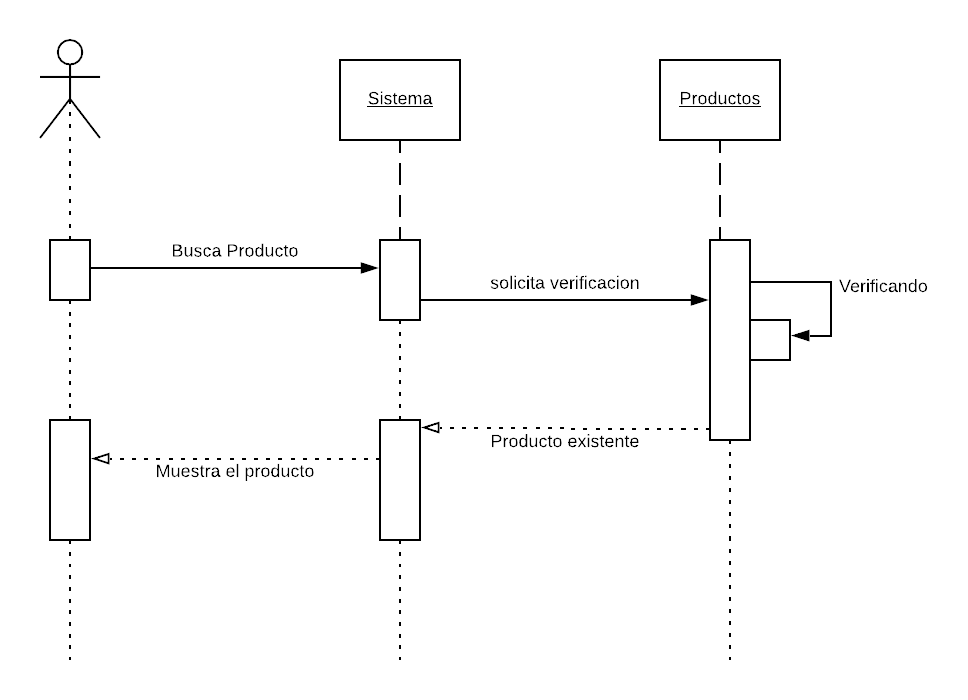
Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo. El diagrama de secuencia contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario, y mensajes intercambiados entre los objetos.

Típicamente se examina la descripción de un caso de uso para determinar qué objetos son necesarios para la implementación del escenario. Si se dispone de la descripción de cada caso de uso como una secuencia de varios pasos, entonces se puede "caminar sobre" esos pasos para descubrir qué objetos son necesarios para que se puedan seguir los pasos.

Un diagrama de secuencia muestra los objetos que intervienen en el escenario con líneas discontinuas verticales, y los mensajes pasados entre los objetos como flechas horizontales.

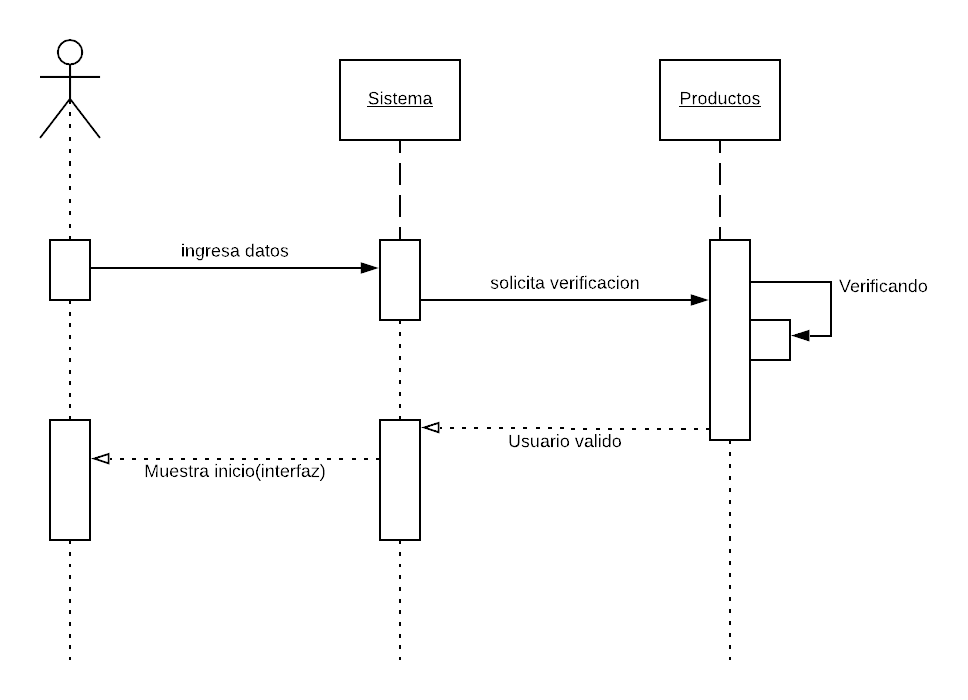
**Escenario: Buscar producto.**

Desde esta opción cualquier usuario puede efectuar la búsqueda de un producto. Se buscarán las coincidencias dentro de la categoría seleccionada y se mostrarán los resultados en un listado.



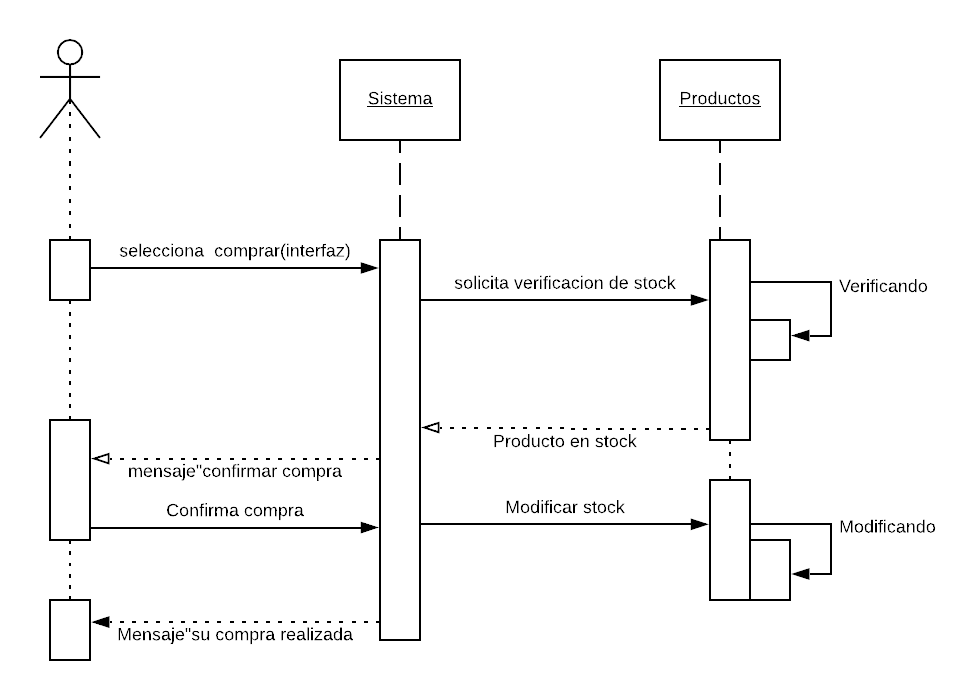
**Escenario: loguearse.**

En la pantalla principal nos aparecerá la opción de identificación, para que los usuarios registrados o los usuarios administradores puedan introducir su identificador y correspondiente contraseña, para poder tener la funcionalidad correspondiente a su rol.



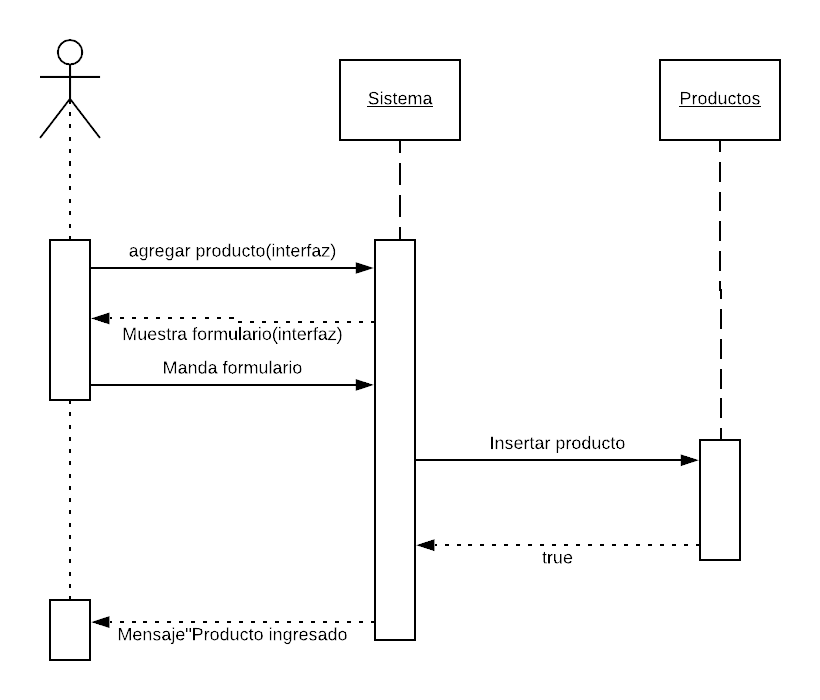
**Escenario: Comprar producto.**

Una vez seleccionados los productos a comprar, se confirma que se quiere realizar la compra, se guardan los datos del pedido en nuestra base de datos se envía un mensaje de confirmación o en caso contrario un mensaje de advertencia.



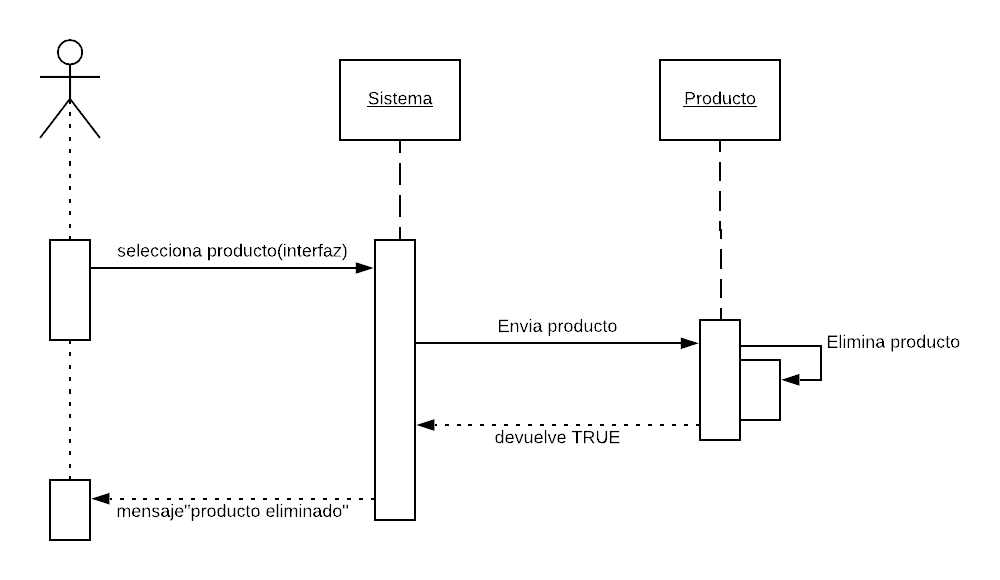
## Escenario: agregar Producto

El administrador una vez identificado, tiene la opción de introducir nuevos productos en nuestro catálogo de productos. Para ello introducirá los detalles del producto y se realizará el alta en nuestra base de datos.



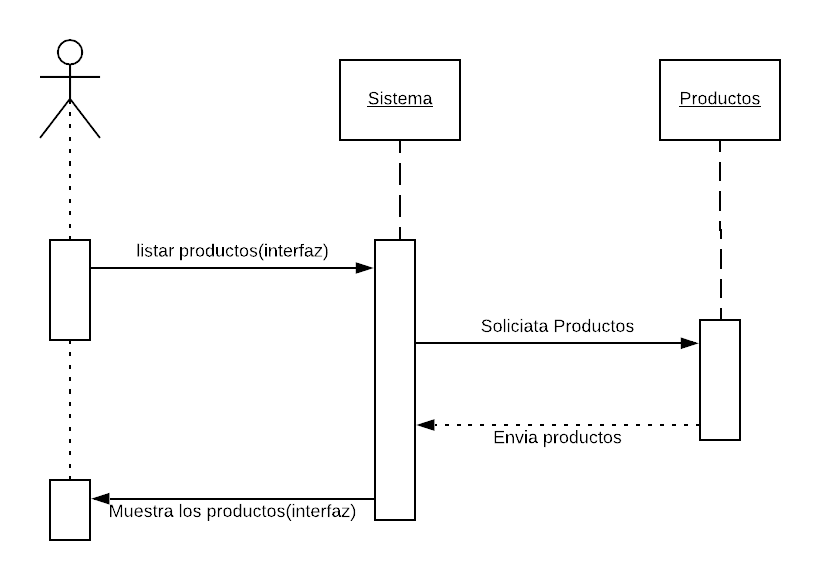
**Escenario:eliminar Producto.**

Al igual que la operación de agregar será el administrador el encargado de efectuar las bajas de productos. Se seleccionará el producto que queremos borrar de nuestra base de datos y pulsando el botón de eliminar efectuaremos la operación indicada.



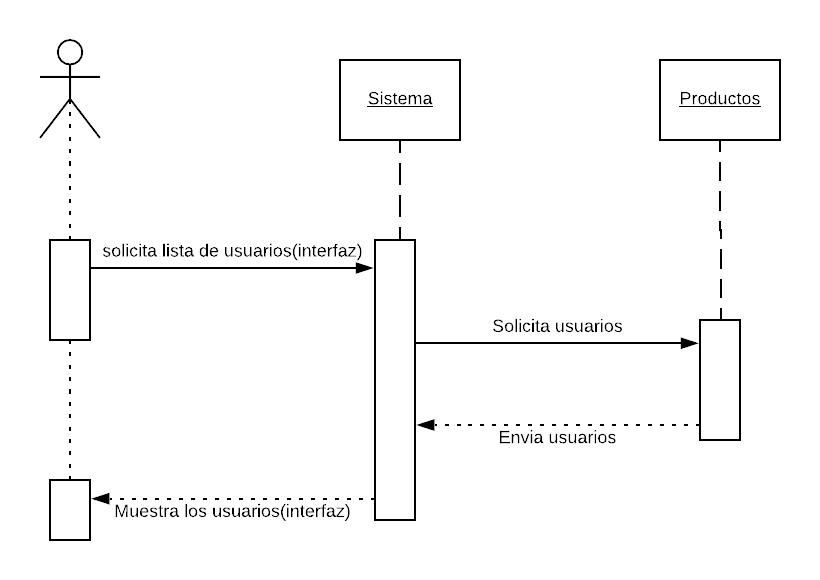
**Escenario: Listar Productos.**

Dentro de esta opción el usuario realiza una petición de todos los productos que están registrados dentro de nuestra base de datos.



**Escenario: Listar usuarios registrados.**

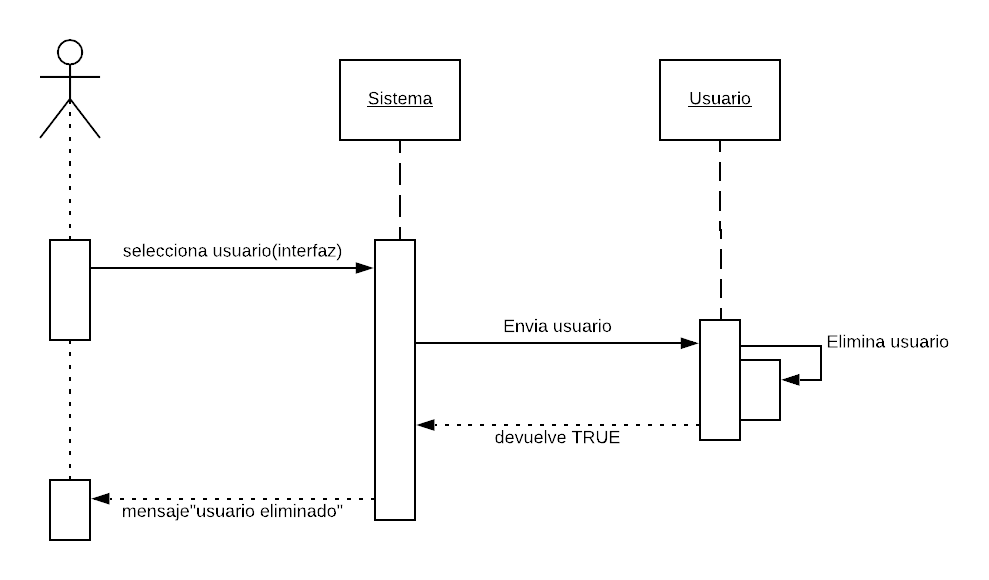
Dentro de esta opción el usuario realiza una petición de todos los usuarios que están registrados dentro de nuestra base de datos.



**Escenario:Eliminar usuario registrado.**

Una vez listado los usuarios seleccionaremos el usuario que se desee eliminar de la base de datos.

.



**3.4 Diagrama de clases.**

El primer paso va a ser la realización del diagrama de clases, es el diagrama principal para el *modelado orientado a objetos (esto es un ejemplo)).* Representa las clases del sistema junto con sus relaciones estructurales agregación y asociación. También se definirán los atributos y operaciones. Con el diagrama de clases entenderemos qué funciones pueden ser llevadas a cabo y quién las puede realizar. Cómo se pueden desarrollar y qué tipo de objetos interactúan entre sí.

Pasamos a describir nuestro diagrama:

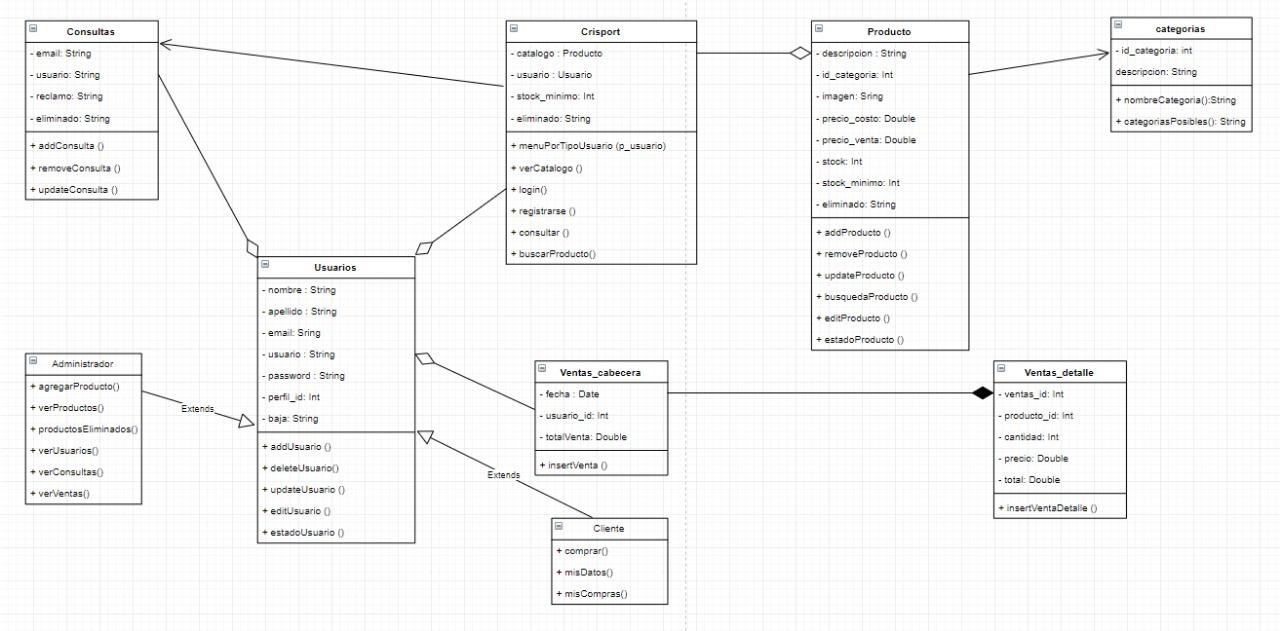
Tenemos tres tipos de usuarios, el usuario, que representará al usuario anónimo, es el más básico de ellos, solo podrá realizar consultas de productos y ver sus características asociadas.

El usuario registrado será una especialización del usuario corriente el cual tendrá la opción añadida de poder efectuar las compras de los productos.

Por otro lado tendremos el administrador que dispondrá de la posibilidad de gestionar el catálogo de productos, usuarios y listados de pedidos.

La clase principal de nuestra aplicación es Tienda, que se compone de distintas categorías, las cuáles a su vez están formadas por productos.

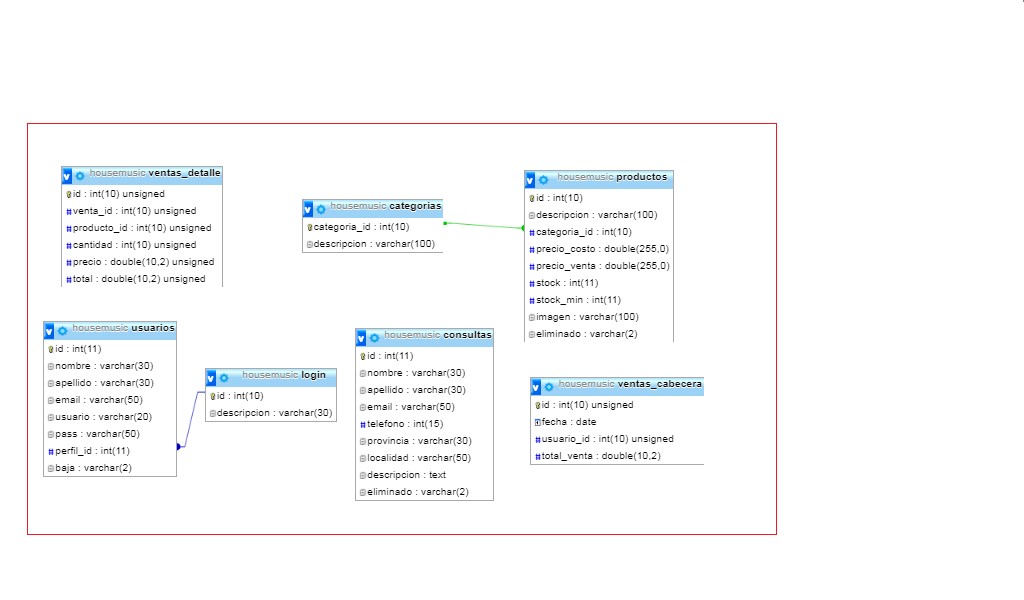
A continuación presentamos nuestro diagrama de clases, con las características descritas anteriormente: Ejemplo:



**3.5 Modelo Entidad Relacion.**

El diagrama Entidad Relación es un modelo de red que describe con un alto nivel de abstracción la distribución de datos almacenados en un sistema.

Una de las razones para realizar este modelo, es porque las estructuras de datos y las relaciones pueden ser tan complejas que sea necesario examinarlas independientemente del proceso. A continuación presentamos nuestro una captura de la relación de las entidades en la base de datos, esto hace referencia al modelo entidad relación.



**En la siguiente tabla se detallan los requerimientos funcionales**

|  |  |
| --- | --- |
| **RF** | **DESCRIPCION** |
| RF#1 | El sistema debe permitir registrar a un nuevo usuario ingresando los datos correspondiente. |
| RF#2 | El sistema debe permitir el logueo ingresando(nombre usuario y contraseña). |
| RF#3 | El sistema debe permitir realizar una compra. |
| RF#4 | El sistema debe permitir realizar una consulta. |
| RF#5 | El sistema debe permitir insertar un nuevo producto al catálogo. |
| RF#6 | El sistema debe permitir eliminar un producto del catálogo. |
| RF#7 | El sistema debe permitir poder ver una lista de los productos. |
| RF#8 | El sistema debe permitir eliminar un usuario |
| RF#9 | El sistema debe permitir la visualización de los usuarios. |
| RF#10 | El sistema debe permitir la visualización de los reportes de ventas. |
| RF#11 | El sistema debe permitir la visualización de las consultas. |
| RF#12 | El sistema debe permitir eliminar las consultas. |
| RF#13 | El sistema debe permitir borrar productos del carrito. |
| RF#14 | El sistema debe permitir modificar datos de usuarios. |
| RF#15 | El sistema debe permitir modificar productos. |