# Requerimientos Funcionales

1. Cliente debe registrarse en el sistema.
2. Cliente debería poder ingresar a la aplicación con sus credenciales, si estas se olvidan, el sistema da la opción de recuperarlas mediante el correo.
3. Cliente puede ingresar al catálogo después de ingresar y poder ver productos disponibles.
4. Cliente puede seleccionar ese producto y su cantidad.
5. Cliente puede agregar más productos adicionales al producto ya inicialmente agregado.
6. Cliente puede remover el producto que desee de la lista de productos agregados.
7. Cliente debe confirmar selección antes de hacer el pago.
8. Cliente puede seleccionar su método de pago para completar el proceso de compra.
9. Cliente puede cancelar la compra en medio del proceso de pago.
10. Cliente debe ingresar información (nombre, apellido, dirección de domicilio, teléfono, email, etc ..)para proceder con el pago. (dirección y envio).
11. Usuario puede guardar su medio de pago favorito para futuras compras.
12. Al confirmar la orden, el sistema confirma la compra y dar la opción de retractarse.
13. El sistema debe confirmar la compra y el inicio de producción del producto.
14. El sistema debe informar cada estado de producción y envío (mediante correo) .
15. El sistema debe informar el inicio de envio ***del producto*** (Mediante correo).
16. El sistema debe confirmar la entrega de producto.
17. Usuario debe confirmar la entrega.
18. El sistema da la opción de compartir experiencia y compartirla al usuario.
19. Administrador puede agregar,editar o eliminar productos del catálogo.
20. Administrador debe agregar cantidad de producto neto (materia prima).
21. Administrador puede visualizar cualquier reporte de datos incluyendo (Cantidad de clientes alcanzados, reportes financieros, etc...)
22. Administrador puede crear, eliminar y modificar cuentas de operadores.
23. Operador puede cambiar, modificar o cancelar estado de producción.
24. Operador puede cambiar, modificar o cancelar estado de envío.

# Requerimientos No Funcionales

### **Eficiencia**

* Toda funcionalidad del sistema y transacción de negocio debe responder al usuario en menos de 5 segundos, desplegando la imágenes de catálogos y la opción de compra.
* El sistema debe ser capaz de operar adecuadamente con hasta 100 usuarios con sesiones concurrentes.
* Los datos modificados en la base de datos deben ser actualizados para todos los usuarios que acceden en menos de 2 segundos.

**Seguridad lógica y de datos**

* Los permisos de acceso al sistema podrán ser cambiados solamente por el administrador de acceso a datos.
* El nuevo sistema debe desarrollarse aplicando [**patrones y recomendaciones de programación que incrementen la seguridad de datos**](http://www.pmoinformatica.com/2014/01/consejos-seguridad-informatica.html).

**Usabilidad**

* El tiempo de aprendizaje del sistema por un usuario deberá ser menor a 5 minutos.
* La tasa de errores cometidos por el usuario deberá ser menor del 1% de las transacciones totales ejecutadas en el sistema.
* El sistema debe contar con manuales de usuario estructurados adecuadamente para el uso de cambios de estado por parte del operador.
* El sistema debe proporcionar mensajes de error que sean informativos y orientados a usuario final.
* El sistema debe contar con un módulo de soporte en línea.
* La aplicación web debe poseer un diseño“Responsive” a fin de garantizar la adecuada visualización en múltiples computadores personales, dispositivos tableta y teléfonos inteligentes.
* El sistema debe poseer interfaces gráficas bien desplegadas.
* El sistema será desarrollado como aplicativo WEB utilizando Play Java Framework para mayor escalabilidad.
* El procedimiento de desarrollo de software a usar debe estar definido explícitamente (en manuales de procedimientos) y debe cumplir con los estándares ISO 9000.
* La metodología de desarrollo de software será **Scrum**
* El sistema debe ser desarrollado utilizando las herramientas Play Framework, Foundation Framework y Mysql como base de datos.
* El proceso de desarrollo se gestionará por medio de Zoho.
* Cada dos semanas deberán producirse reportes gerenciales en los cuales se muestre el esfuerzo invertido en cada uno de los componentes del nuevo sistema.
* Las pruebas de software se gestionaran con Jmeter y se harán pruebas unitarias con variables constantes sobre las clases realizadas.
* La interfaz de usuario será implementada para navegadores web únicamente con HTML5 y JavaScript.
* El sistema no revelara a sus operadores otros datos personales de los clientes distintos a nombres y números de referencia.

**Dependabilidad**

* El sistema debe tener una disponibilidad del 99,99% de las veces en que un usuario intente acceder.
* La tasa de tiempos de falla del sistema no podrá ser mayor al 0,5% del tiempo de operación total.
* La probabilidad de falla del Sistema no podrá ser mayor a 0,05.

**Otros ejemplos de requerimientos de producto**

* La aplicación debe ser compatible con todas las versiones de Windows, desde Windows 95.
* La aplicación deberá consumir menos de 500 Mb de memoria RAM.
* La aplicación no podrá ocupar más de 2 GB de espacio en disco.
* La nueva aplicación debe tener los lenguajes Ingles y Español.

# Actores del Sistema

|  |  |
| --- | --- |
| Actores | Descripción |
| Cliente | Persona que a través del sistema realiza compras de producto, consulta catálogo y comparte experiencias. |
| Administración | Persona principal en el manejo de usuarios, ingreso de nuevos productos, acceso de reportes de datos de negocios(visitas, productos seleccionados, ventas de productos, compras realizada y demás) |
| Sistema | Acción que realiza el sistema para informar, procesar o modificar el proceso actual. |
| Operador | Persona la que realiza el proceso de producción y manufactura del producto. Cambiando los estados del producto llevado a producción. |

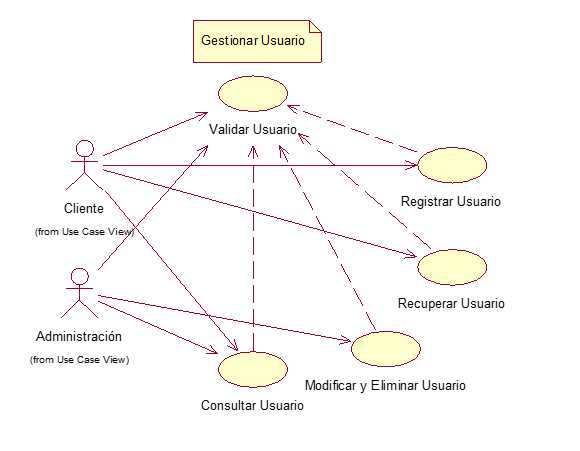
# Identificar Casos de uso del sistema

|  |  |  |
| --- | --- | --- |
| N° reque | Requerimiento | Nombre de Caso de usos |
| 1 | Cliente debe registrarse una cuenta en el sistema. | Gestionar usuarios. |
| 2 | Cliente debería poder ingresar a la aplicación con sus credenciales, si estas se olvidan, el sistema da la opción de recuperarlas mediante el correo. |
| 22 | Administrador puede crear, eliminar y modificar cuentas de operadores. |
| 19 | Administrador puede agregar,editar o eliminar productos del catálogo. | Gestionar productos. |
| 20 | Administrador debe agregar cantidad de producto neto (materia prima). |
| 3 | Cliente puede ingresar al catálogo después de ingresar y poder ver productos disponibles. | Gestionar pedido |
| 4 | Cliente puede seleccionar ese producto y su cantidad para crear un pedido. |
| 5 | Cliente puede agregar más productos adicionales al producto al pedido. |
| 6 | Cliente puede remover el producto que desee del pedido. |
| 7 | Cliente debe confirmar pedido antes de hacer el pago. |
| 8 | Cliente puede seleccionar su método de pago para completar el proceso de compra. | Gestionar pago. |
| 9 | Cliente puede cancelar pedido en medio del proceso de pago. |
| 11 | Cliente puede guardar su medio de pago favorito para futuras compras. |
| 12 | Al confirmar la orden el sistema confirma la compra y dar la opción de retractarse. |
| 13 | El sistema debe confirmar la compra y el inicio de producción del producto. |
| 10 | Cliente debe ingresar información (nombre, apellido, dirección de domicilio, teléfono, email, etc ..)para proceder con el pago. (dirección y envio). |
| 14 | El sistema debe informar cada estado de producción y envío (mediante correo) . | Gestionar producción y entrega. |
| 15 | El sistema debe informar el inicio de envio ***del producto*** (Mediante correo). |
| 16 | El sistema debe confirmar la entrega de producto. |
| 17 | Cliente debe confirmar la entrega. |
| 18 | El sistema le da la opción de compartir experiencia y compartirla al cliente . |
| 23 | Operador puede cambiar, modificar o cancelar estado de producción. |
| 24 | Operador puede cambiar, modificar o cancelar estado de envío. |
| 21 | Administrador puede visualizar cualquier reporte de datos incluyendo (Cantidad de usuarios alcanzados, reportes financieros, etc...) | Analizar datos. |

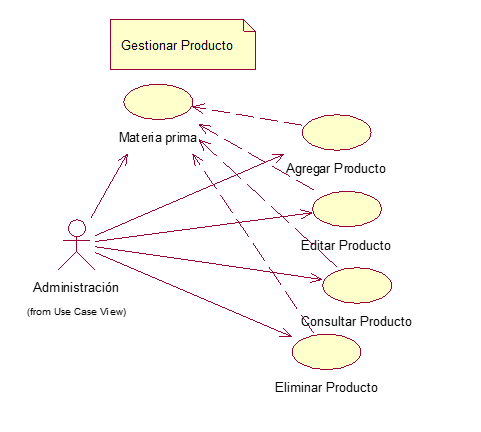
# 

# Diagrama de casos de Usos

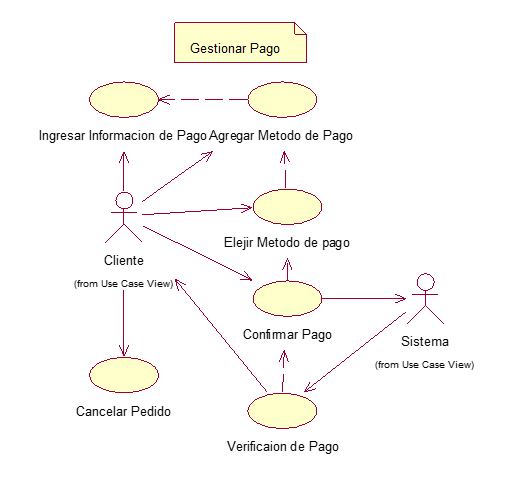
## 



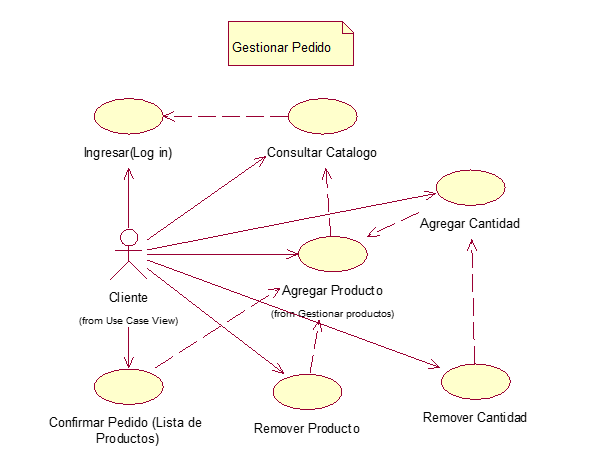
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CU 1 | CU Gestionar Usuarios | | | |
| Versión | 0.1 | | | |
| Dependencias | CU-01: Validar Usuario (Login) | | | |
|
|
| Precondición | El usuario debe ingresar a la aplicación mediante un explorador de escritorio o móvil desde su celular, tener la capacidad de ingresar los datos mediante teclado y mouse (teclado virtual y taps para dispositivos móviles) | | | |
|
| Descripción | El siguiente caso permite al usuario ingresar a la aplicación si por algún motivo no se le permite ingresar se le da la opción de registrarse para poder ver el catálogo completo de productos además de poder realizar un pedido y ver el estado de este. | | | |
| Secuencia Normal | Paso | Acción | | |
| 1 | Se solicita ingresar con su correo electrónico (email) y su contraseña. | | |
| 2 | Se valida el Ingreso a la aplicación. | | |
| 3 | Se genera la sesión de cliente | | |
| 3.1 | Si es la sección de administrador tiene la opción de Consultar, Modificar Y Eliminar Usuario | |
| Postcondicion | Al ingresar exitosamente se deben permitirle al cliente las opciones de ver catalogo, pedido y estado de su pedido\*. | | | |
|
| Excepciones | Paso | Acción | | |
| 1 | 1.1 | Si no se encuentra registrado se le da la opción de registrarse (Ingresando su nombre, email y contraseña). | |
|  | 2 | 2.1 | Si no se puede validar el usuario se le da la opción de recuperar su usuario. | |
|  | 2.2 | Usuario debe ingresar su email para la recuperación de usuario | |
|  | 2.2.1 | Si no se encuentra el email se la opción de registrarse (1.1) |
| Comentarios | \*Hay diferentes usuarios o secciones: Administrador, Operador y Cliente. Dependiendo de cuál es la desplegada después de este proceso mostrará diferentes opciones en el sistema. | | | |
|
|



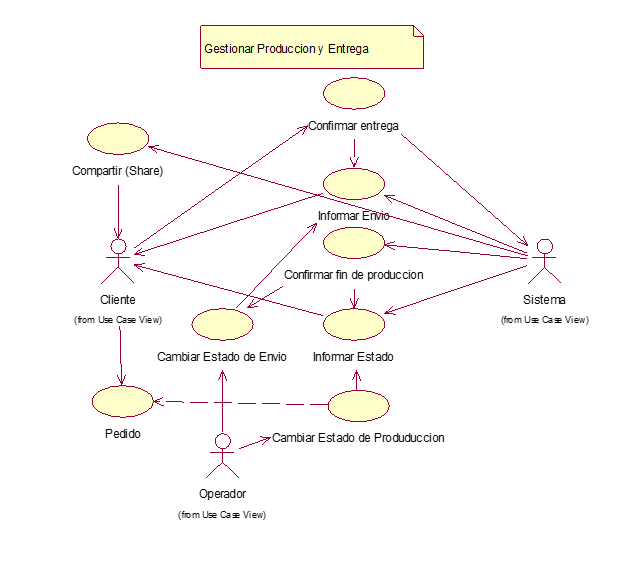
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CU 2 | CU Gestionar Producto | | | |
| Versión | 0.1 | | | |
| Dependencias | CU-01: Validar Usuario (Login) | | | |
|
|
| Precondición | El administrador debe validar su usuario e ingresar en la página de gestionar productos. | | | |
|
| Descripción | El siguiente caso de uso permite ingresar, consultar, editar y eliminar productos. | | | |
| Secuencia Normal | Paso | Acción | | |
| 1 | Administrador agrega el tipo de producto que está agregando y cuanta cantidad tenemos. En este caso lo llamamos (Materia Prima) | | |
| 2 | Después de tener la materia prima, se agrega el producto como tal (con su nombre, descripción (imágenes), precio y disponibilidad) | | |
| 2.1 | Si ya hay producto agregado se le da la opción de buscar el producto por su nombre o id. | |
| 2.1.1 | Si es encontrado el producto el administrador tiene la opción de editar o eliminar el producto |
| Postcondicion | Se muestra mensaje de confirmación cada vez que se hace algún proceso en el producto de igual manera se reflejan los cambios realizados en el catálogo. | | | |
|
| Excepciones | Paso | Acción | | |
| 2 | Si no hay materia prima para ese producto en específico se le pide al Administrador agregar más cantidad a la materia prima. | | |
| 2.1 | Si el producto no se encuentra en la consulta, se le da la opción al Administrador de agregar un nuevo producto. | | |
| Comentarios | En el catálogo se podrá consultar los productos mediante filtros. | | | |
|
|



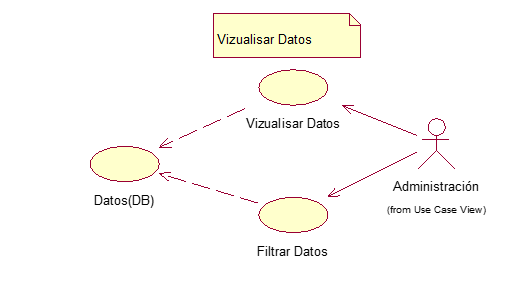
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| CU 4 | CU Gestionar Pago | | | | | |
| Versión | 0.1 | | | | | |
| Dependencias | CU 1 Gestionar Usuario , CU 3 Gestionar Pedido | | | | | |
|
|
| Precondición | El cliente debe tener un pedido ya confirmado para que el sistema le da la opción de pago. | | | | | |
|
| Descripción | En el siguiente caso de uso se da la opción de pago al cliente del pedido ya creado en el anterior caso de uso CU 3 | | | | | |
| Secuencia Normal | Paso | Acción | | | | |
| 1 | El sistema desplegará la página de pago de pedido. | | | | |
| 3 | El sistema desplegará diferentes métodos de pagos (efecty o tarjeta de crédito\*). | | | | |
| 4 | El cliente debe agregar un método de pago | | | | |
| 4.1 | El cliente puede agregar más de un medio de pago |  |  |  |
| 4.2 | Puede seleccionar de la listar de medios de pago, su preferido. | | | |
| 5 | Cliente debe ingresar información de pago (nombre, apellido y dirección ). | | | | |
| 6 | Sistema realiza el proceso de pago. | | | | |
| 7 | Después de realizar el proceso de pago le pide al cliente confirmar el pago. | | | | |
| 8 | El sistema envía una verificación de pago. | | | | |
| 9 | El sistema cambia el estado de pedido a producción. | | | | |
| Postcondicion | Al verificar el pago se debe enviar correo con factura, resumen de pedido y la estimación del tiempo de entrega. | | | | | |
|
| Excepciones | Paso | Acción | | | | |
| 4.2 | Cliente puede eliminar cualquier medio de pago. | | | | |
| 5 | La información que se pide depende del medio de pago elegido. | | | | |
| 5.1 | Se confirma la dirección de entrega del pedido. | | | |
| 7 | Se vuelve a verificar si la transacción se ha realizado con éxito antes de enviar la verificación. | | | | |
| Comentarios | \* En este momento se piensa solo realizarlo con Efecty. Además, en este proceso antes de confirmar el cliente puede cancelar el pedido y se realiza la devolución del dinero y se elimina el pedido. | | | | | |
|
|



|  |  |  |  |
| --- | --- | --- | --- |
| CU 3 | CU Gestionar Pedido | | |
| Versión | 0.1 | | |
| Dependencias | CU 1 Gestionar Usuario, CU 2 Gestionar Producto | | |
|
|
| Precondición | Cliente debe validar su usuario e ingresar al catálogo de productos para realizar su pedido agregando productos. | | |
|
| Descripción | El siguiente caso de uso describe el proceso de creación de pedido de los productos ofrecidos en el catálogo. | | |
| Secuencia Normal | Paso | Acción | |
| 1 | Cliente debe consultar en el catálogo el producto que desea agregar al pedido. | |
| 2 | Después de seleccionar su producto del catálogo puede agregarlo al pedido cliqueando el botón agregar al pedido. | |
|  | 2.1 | Si ya hay un producto en el pedido, el nuevo producto seleccionado se agrega a este pedido. |
|  | 2.2 | El cliente puede eliminar independientemente los productos que desee. |
| 3 | Cuando el producto este agregado puede agregar o remover la cantidad que desee del mismo producto. | |
| 4 | Al finalizar su pedido el cliente debe confirmar el pedido. | |
| Postcondicion | Después de realizar el pedido, se procede a realizar el proceso de Pago. | | |
|
| Excepciones | Paso | Acción | |
| 1 | El cliente solo puede seleccionar productos que tengan el estado disponible en el catálogo. | |
| 2.2 | Si el cliente elimina el producto del pedido eliminará toda la cantidad que haya agregado de este. | |
| 3 | Se limitará la cantidad del producto dependiendo de la materia prima que cuente este producto. | |
| 4 | El cliente antes de confirmar el pedido, puede cancelar todo el pedido y volviendo a la página principal. | |
| Comentarios | En la ventana emergente del pedido se mostrará la lista de productos agregados, además de la opción de eliminar el producto o una cantidad específica. Además, con el resumen del costo del pedido, con el precio de cada uno de los productos y el total del pedido. Si hay algún descuento se mostrará de la misma manera. | | |
|
|



|  |  |  |  |
| --- | --- | --- | --- |
| CU 5 | CU Gestionar Producción y Entrega | | |
| Versión | 0.1 | | |
| Dependencias | CU 1 Gestionar usuario, CU 3 Gestionar Pedido, CU 4 Gestionar Pago | | |
|
|
| Precondición | Para administrar la producción y la entrega el cliente debe tener un pedido y ya debe estar pago. | | |
|
| Descripción | El siguiente caso de uso describe el proceso de cómo el operador cambia el estado de producción en que se encuentra el pedido; El proceso es de manufactura hay diferentes estados (Tallado, Pintado, Secado, Detallado, Bordado, Tejido). Después de finalizar la producción otro operador cambia el estado de entrega, en cada uno de estos el sistema informa al cliente desplegando el estado en la página de consulta del pedido. | | |
| Secuencia Normal | Paso | Acción | |
| 1 | Operador después de terminar proceso de manufactura cambia el estado de pedido mediante botones. | |
| 2 | Sistema actualiza estado del pedido en la página del pedido además de enviar correo al cliente. | |
| 3 | Operador debe confirmar el fin de la producción, | |
| 4 | Sistema debe informar al cliente mediante la aplicación y correo que se acabó el proceso de producción y el de entrega comenzó. | |
| 5 | Operador debe actualizar mediante la aplicación el estado de envió (Empacado, Listo para arrancar, En camino, Percance, Entregado) | |
| 6 | Al entregarse el pedido el cliente debe confirmar la entrega del pedido mediante la aplicación. | |
| 7 | Sistema confirma pedido y le ofrece compartir la experiencia a el cliente. | |
| Postcondicion | Poder compartir su experiencia y almacenar el pedido en la base de datos como exitoso. | | |
|
| Excepciones | Paso | Acción | |
| 1 | Si ocurre un percance no previsto se le dará una opción de un estado personalizado en el proceso de manufactura para informar al cliente y al administrador. | |
| 6 | Si el cliente no confirma el pedido se cambia de estado del pedido a Entregado no recibido y se le da un periodo de 30 días al cliente para confirmar el pedido. | |
| 7 | La opción de compartir es obligatoria, se le da varias opciones desde redes sociales a mensajes privados, si se evita el compartir cada vez que ingrese a la aplicación se negara el acceso hasta que no comparta. | |
|  |  | 7.1 | El sistema de compartir tiene un indicador de satisfacción con estrellas donde si este es menor a 3 no se hace público. |
| Comentarios | Este proceso da la idea de negocio general del proyecto, dar una opción y una solución a pequeño manufactureros, productores y vendedores una opción de comercio electrónico. | | |
|
|



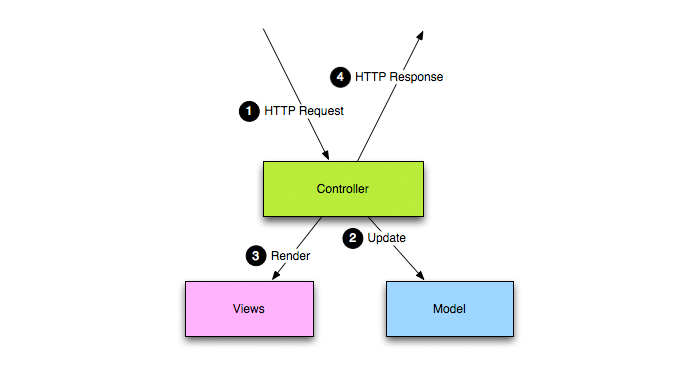
|  |  |  |  |
| --- | --- | --- | --- |
| CU 6 | CU Visualizar Datos | | |
| Versión | 0.1 | | |
| Dependencias | CU 1 Gestionar Usuarios, CU 2 Gestionar Productos. CU 3 Gestionar Pedidos. CU 4 Gestionar Pagos , CU 5 Gestionar Producción y Entrega | | |
|
|
| Precondición | Disponibilidad de datos en el sistema (Base de datos) | | |
|
| Descripción | Dar resultados de los procesos realizados por la aplicación. Poder analizar, predecir y reportes cronológicos de procesos. | | |
| Secuencia Normal | Paso | Acción | |
| 1 | Después de validar su usuario el Administrador puede ver la página de reportes y análisis de datos. | |
| 2 | Administrador puede filtrar el reporte además de otra opciones. | |
| 3 | El sistema despliega un reporte detallado de la consulta hecha. | |
| Postcondicion | Muestra de gráficos, tablas y reportes solicitados por el administrador. | | |
|
| Excepciones | Paso | Acción | |
| 2 | Los filtros se activan dependiendo de selección de otros filtros. | |
|  | 2.1 | Además de los filtros el administrador tiene la opción de seleccionar consultas especiales como predicciones. |
| Comentarios | Como valor adicional se pueden personalizar estéticamente los reportes y gráficos que genere la página. | | |
|
|

# Arquitectura

## Modelo-Vista -Controlador

Para asegurar la calidad de la aplicación elegimos MVC (modelo- vista - controlador) en este caso utilizamos PLAY framework que integra todo los beneficios de utilizar esta arquitectura.

## Diagrama de Arquitectura



### Modelo

Es la capa que utiliza los datos y los trabaja. Estos datos están almacenado en este caso en la base de datos Primate (MYSQL 5.7.16). Play Framework utiliza una **ORM**[[1]](#footnote-0) llamada JAVA Ebean que nos permite trabajar con abstracción de la base de datos y persistencia de objetos, para así evitar las sentencias SQL directas. La capa de modelo definida en la carpeta **app** es una compilación de clases de Java utilizando todas las herramientas de la programación orientada a objetos que ofrece Java. Contiene estructuras de datos y operaciones que la aplicación utiliza. Cuando el modelo necesita guardar en almacenamiento persistente, puede contener artefactos de conexión como JPA.

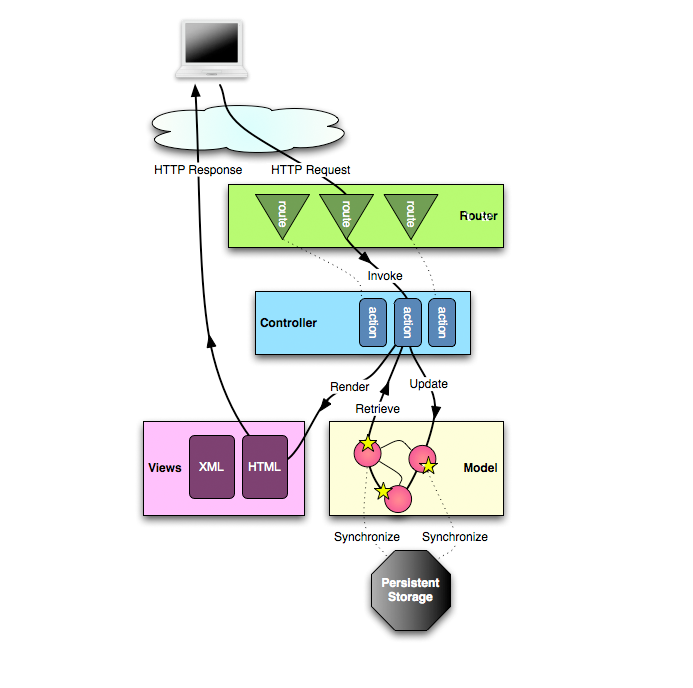
### Vista

Como se da entender por su nombre esta capa produce la visualización de la interfaz de usuario. Play framework utiliza HTML5 y CSS3 para su facil manejo utilizamos el framework Boostrap 3 que maneja efectivamente estas dos tecnologías. La capa de vista definida en la carpeta app contiene los archivos del template HTML, JSON que transforman la información obtenida del modelo por el controlador y después aplica el template para decorar estos objetos.

### Controlador

Esta capa contiene las “Acciones” código necesario para responder a las solicitudes de la aplicación, como visualizar Usuario, realizar pago de pedido. etc. Esta capa ayuda como enlace entre las vistas y los modelos para poder implementar la diversas necesidades del desarrollo. En una WEB application eventos son por lo general solicitudes HTML: un controlador escucha esta solicitud, obtiene información significativa del evento y aplica los objetos del modelo subyacentes. La capa de controlador est definida en el carpeta **app.**

### Ciclo de vida de la Solicitud



* Una solicitud es recibida por el framework
* El enrutador intenta encontrar la ruta más específica que pueda aceptar la solicitud.
  + La “acción” correspondiente es invocada.
* El código de la aplicación es ejecutado.
* Si una vista compleja necesita ser generada, un template es rendered.
* El resultado del método “acción” es entonces escrito como HTTP response.

# Requerimientos Técnicos

Ahora que la arquitectura ha sido explicada, se hace una pregunta que es PLAY Framework, que es Boostrap, Java Ebean, HTML5, CSS etc.. Que necesito para poder desarrollar en todas estas tecnologías.

## Tecnologías

Primero debemos saber todas las tecnologías utilizadas en el proyecto.

### PLAY framework

Para los que no saben que es Play es un framework de desarrollo web para Java y Scala. En 2007 fue un proyecto interno de Guillaume Bort para la compañía Zenexity (ahora Zengurality ) fue fuertemente influenciado de cómo ellos hacían proyectos web utilizando las mejores prácticas de JEE.

En 2009 decidieron compartir esta idea con la comunidad como un proyecto de código abierto (open source) gracias a esto actualmente tiene una comunidad activa de más de 10.000 personas, con un número creciente de aplicaciones corriendo en producción alrededor del mundo como  linkedin como una de la más famosas.

Gracias a ser un proyecto abierto para la comunidad ha dado mucha retroalimentación, así pudiendo solucionar errores y bugs, también incluyendo nuevas herramientas que no se consideraban en diseño original. Así como el proyecto ha crecido, han aprendido de la comunidad y de su propia experiencia - utilizando Play de una manera más compleja en variados proyectos.

Mientras la tecnología y el internet continúan evolucionando. La Web ha sido el punto central de todas las aplicaciones. HTML, CSS y Javascript han evolucionado de una manera acelerada haciendo difícil para un framework de cliente mantenerse a este ritmo. La arquitectura de la web se está moviendo a un procesamiento en tiempo real.

Para esto se creó Play un nuevo framework para una nueva era.

#### SBT

Que es SBT? Scala Build Tool. Es un herramienta que provee la facilidad de compilar, correr, testear, y empaquetar proyectos. Play framework utiliza Activator para utilizar sbt con unos comandos simples como activator new, run, start.

#### JAVA

Java como el lenguaje numero uno de programación en 2014, donde se estima que hay mas 8 billones de dispositivos habilitados con esta tecnología. Siendo fácil aplicar la metodología de la programación orientada a objetos, patrones, además de sus extensas librerías. Hacen que java sea el lenguaje escogido para manejar los requerimientos del proyecto.

#### MYSQL

El sistema de gestión de bases de datos relacionales más popular del mundo. Utilizado en aplicaciones WEB como Joomla, Drupal o phpBB, su rápida lectura hacen de esta base datos la ideal para el proyecto.

#### Bootstrap

Framework desarrollada por Mark Otto y Jacbod Thornton de twitter es una de las herramientas más famosas para acelerar el desarrollo de interfaces de usuario utilizando HTML5 y CSS3. Gracias a su estructura modular y sus hojas de estilos CSS que implementan la variedad de componentes de la herramienta. Hacen más agradable a la vista de usuario la interfaz y mucho más fácil la creación de estas vistas.

##### HTML5

HyperText Markup Language es el lenguaje de marcado para la elaboración de páginas web. Es un estándar que define una estructura básica y un código para la definición de contenido web, como texto, imágenes, videos, etc… Con su más reciente versión 5 trae novedades y herramientas que hacen su fácil codificación y visualización.

##### CSS3

Cascading Stylesheets es un lenguaje para definir y presentar un documento HTML. Con este lenguaje se establece el diseño visual de la interfaces de usuarios, la separación del contenido del documento y la forma de presentación de este utilizando capas, colores, layouts y fuentes.

#### JAVASCRIPT

JS es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.[[2]](#footnote-1) Se utiliza para interactuar con una página web.

##### jquery

jQuery es una [biblioteca](https://es.wikipedia.org/wiki/Biblioteca_(inform%C3%A1tica)) de [JavaScript](https://es.wikipedia.org/wiki/JavaScript), consiste en un único fichero que contiene las funcionalidades comunes de DOM, eventos, efectos y AJAX.

La característica principal de la biblioteca es que permite cambiar el contenido de una página web sin necesidad de recargar, mediante la manipulación del árbol DOM y peticiones AJAX.

##### coffeeScript

CoffeeScript es un lenguaje de programación que se compila a [JavaScript](https://es.wikipedia.org/wiki/JavaScript). El lenguaje añade [azúcar sintáctico](https://es.wikipedia.org/wiki/Az%C3%BAcar_sint%C3%A1ctico) inspirado en [Ruby](https://es.wikipedia.org/wiki/Ruby), [Python](https://es.wikipedia.org/wiki/Python) y [Haskell](https://es.wikipedia.org/wiki/Haskell)2 para mejorar la brevedad y la legibilidad de JavaScript, y añade características más sofisticadas, como la comprensión de listas y la coincidencia de patrones

#### Ebean

Es un producto de mapeo de objeto-relacional ORM escrito en JAVA que incorpora características de ‘SQL/Relational’, para así poder controlar fácilmente las consultas SQL, llamando proceso almacenados (stored procedures). El objetivo de Ebean es proveer las mejores herramientas ORM de JPA a una sola Framework de persistencia.

## Infraestructura



### Cliente

#### Hardware

* Sistema operativo: Windows, Linux o Mac (Ultimas versiones y variaciones)
* Navegador: Google Chrome, Mozilla firefox, Safari, Edge o Operan (Últimas versiones)

### Servidor

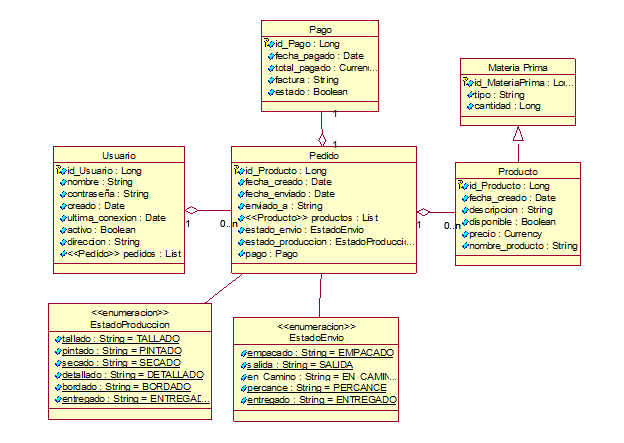
#### Hardware

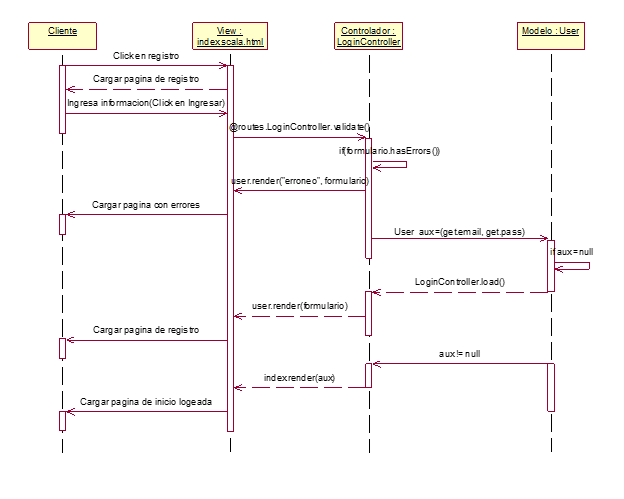
* AMD FX 8320 a 3.5GHz
* 8192 MB de memoria RAM.
* 2438000 MB de espacio libre en el disco duro.
* Conexión de internet de 10MB/s download y 2 MB/s upload.

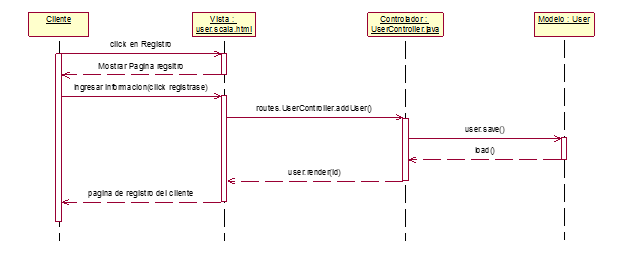
#### Software

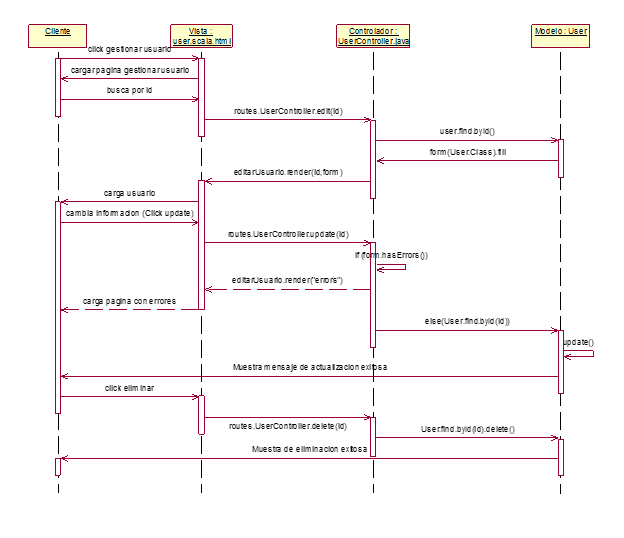
* MySql Database 5.7.16
* Play framework 2.5.x
* JAVA version 1.5.1.101
* SBT 0.13
* Jquery 3.1.1
* Bootstrap 3.3.7
* Ebean 3.0.2
* coffeescript 1.0.0
* Sistema operativo: windows 10 profesional (Anniversary Update)

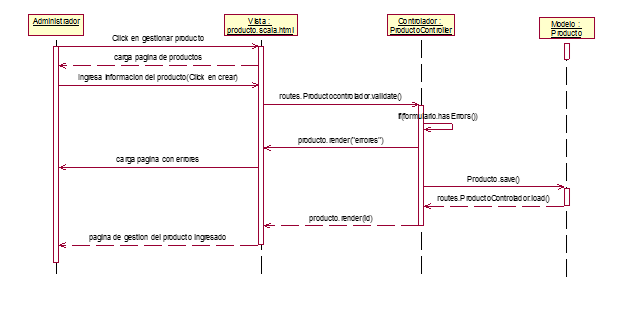
## Diagrama de Clases

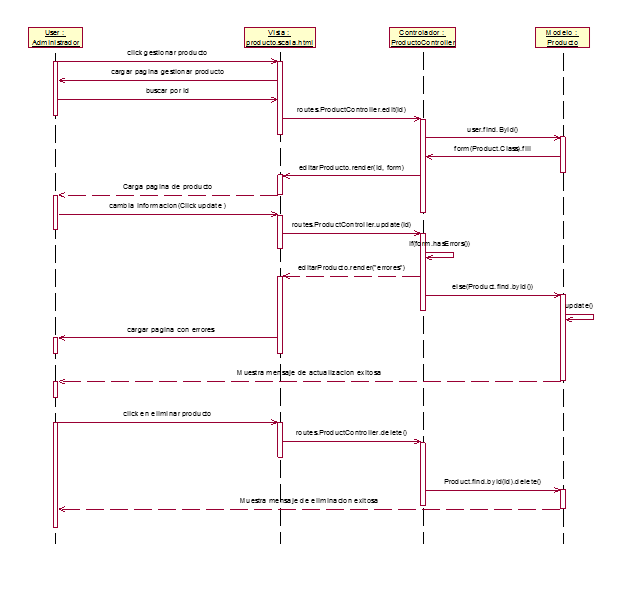


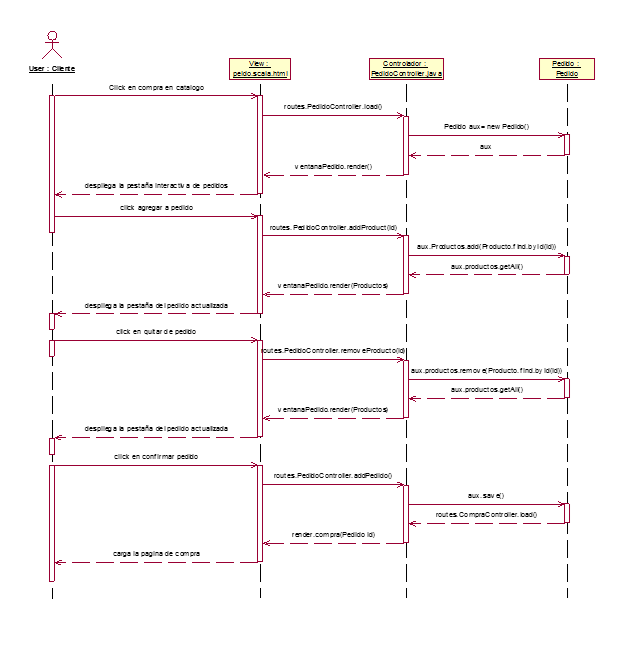


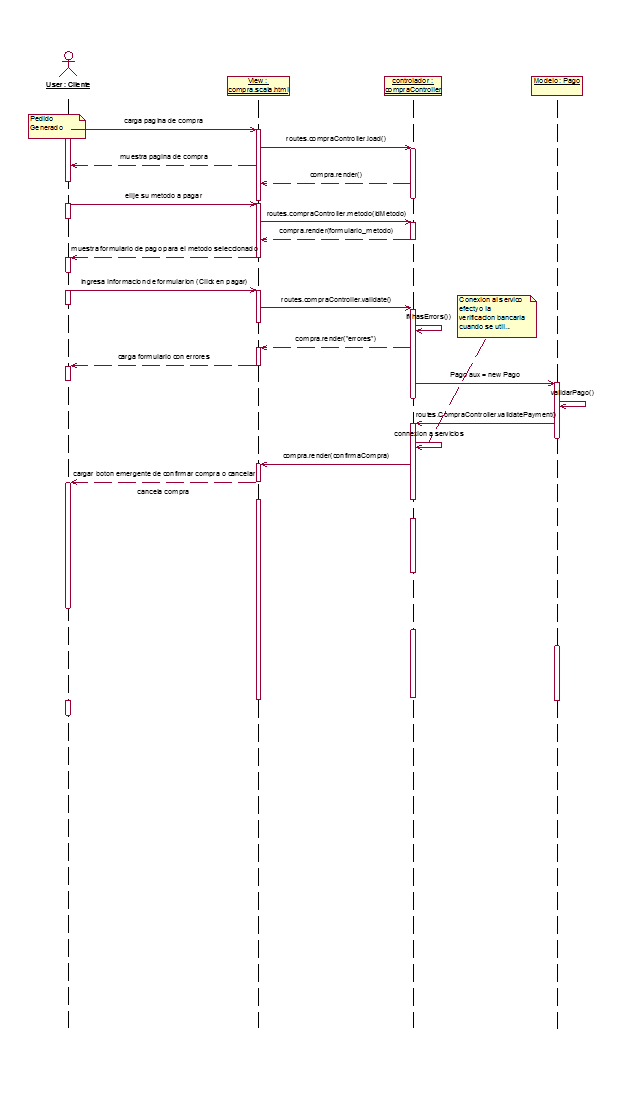


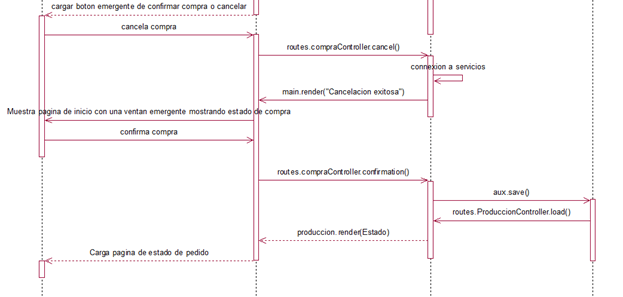


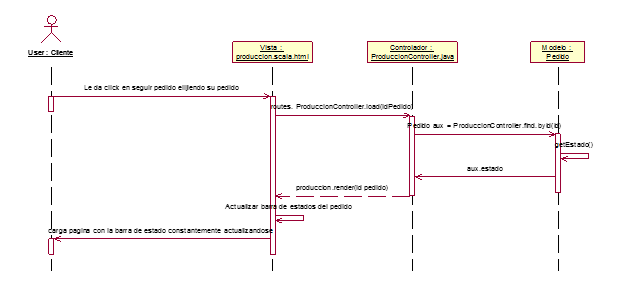












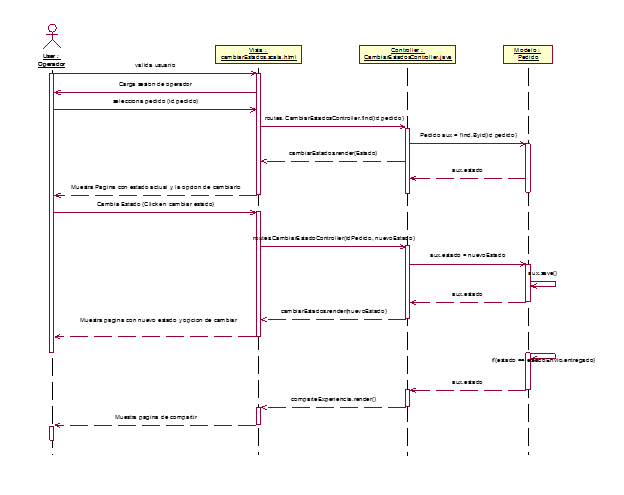
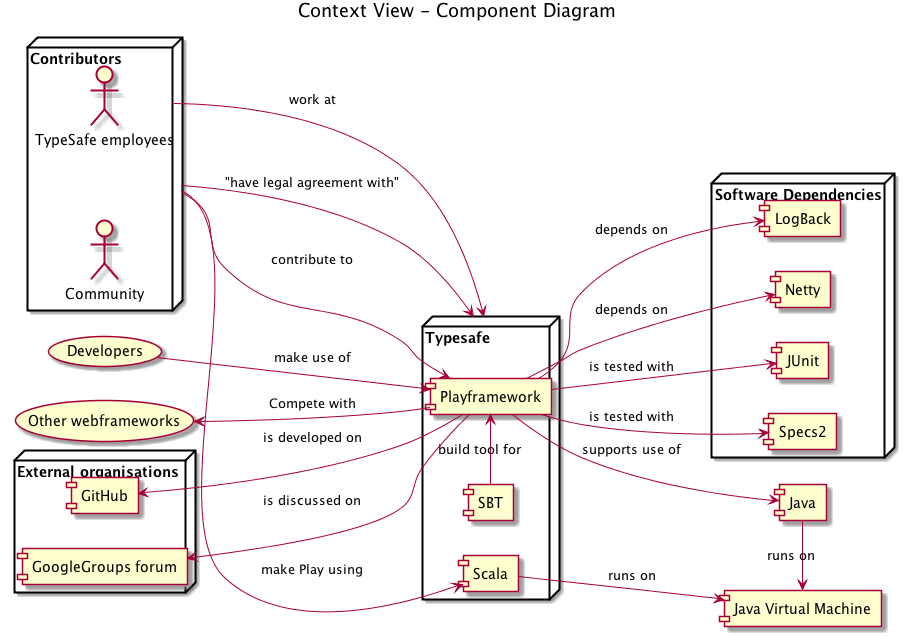


Diagrama de PlayFramework



1. [↑](#footnote-ref-0)
2. https://es.wikipedia.org/wiki/JavaScript [↑](#footnote-ref-1)