

Sistema de Información para Negocio PegaMaz



Jorge Mario Guaqueta Restrepo

Universidad Distrital Francisco José de Caldas
Ingeniería de Software I
Facultad de Ingeniería
Bogotá D.C.
Abril 1 de 2017

Resumen

El proyecto consiste en un sistema de información en donde el cliente (**Pega Maz**) puede registrar sus ventas de manera manual, posteriormente al momento de realizar la contabilidad el cliente podrá ingresar a un modulo de visualización de ventas el cual le proporcionará un informe de ventas dentro de un rango de fechas que el usuario seleccione, por ultimo el cliente tendra acceso a la modificación de inventarios en donde podra agregar productos, eliminar productos y agregar cantidades a los productos.

Tabla de Contenido

Resumen	I
Introducción	1
1. Planeación	2
1.1. Planteamiento del Problema	2
1.2. Justificación	2
1.3. Objetivos	3
1.3.1. Objetivo General	3
1.3.2. Objetivos Específicos	3
1.4. Alcances	3
1.5. Limitaciones	3
1.5.1. Marco Metodológico	3
1.6. Factibilidad	6
1.6.1. Factibilidad Técnica	6
1.6.2. Factibilidad Operativa	7
1.6.3. Factibilidad Económica	7
1.6.4. Factibilidad Legal	8
1.7. Cronograma	8
2. Análisis	10
2.1. Análisis de Requerimientos	10
2.1.1. Requerimientos	11
2.2. Definición de Actores	11
3. Diseño	14
3.1. Casos de Uso	14
3.1.1. Diagramas de Casos de Uso	14
3.1.2. Documentacion de casos de uso	16
3.2. Diagrama de Clases	19
3.3. Diagramas de Secuencia	20
3.4. Diagramas de Componentes	20
3.5. Diagrama de Despliegue	20
3.6. Diagrama BPMN	21

Recomendaciones	27
Conclusiones	28
Bibliografia	29

Lista de Figuras

1.1. Ciclo de Scrum	6
1.2. Cronograma	9
2.1. Diagrama de contexto	11
3.1. Diagrama de Casos de Uso de Administrador	14
3.2. Diagrama de Casos de Uso de Ingreso	15
3.3. Diagrama de Casos de Uso de Inventario	15
3.4. Diagrama de Casos de Uso de Facturación	16
3.5. Diagrama de Casos de Uso de Contabilidad	16
3.6. Diagrama de Clases	20
3.7. Diagrama de secuencia del administrador	22
3.8. Diagrama de secuencia del usuario	23
3.9. Diagrama de secuencia del usuario de contabilidad	24
3.10. Diagrama de componentes del sistema de información.	25
3.11. Diagrama de despliegue del sistema de información.	25
3.12. Diagrama BPMN del proceso de facturación	25
3.13. Diagrama BPMN del proceso de manejo de inventarios.	26
3.14. Diagrama BPMN del proceso de edición de usuario.	26
3.15. Diagrama BPMN del proceso de generación del reporte de contabilidad.	26

Lista de Tablas

1.1.	Actividades durante un Sprint de SCRUM	4
1.2.	Costo de Recursos	8
1.3.	Costo de Integrantes del Proyecto	8
2.1.	Tabla de requerimientos funcionales	12
2.2.	Tabla de requerimientos no funcionales	13
2.3.	Tabla de definición de los actores	13
3.1.	Documentación de Caso de Uso	16

Introducción

Durante mucho tiempo las empresas o negocios han llevado su contabilidad de forma manual, así como también sus inventarios y su facturación por lo que se hace necesario implementar sistemas de información en los cuales se pueda realizar una cierta cantidad de actividades que permiten un mejor manejo de los costos, gastos y flujos que se presenten en el negocio.

En cuanto a la necesidad se plantea un sistema de información el cual permitirá facturar los productos que serán vendidos, además de permitir mediante el módulo de inventarios llevar un inventario de los productos que están siendo comercializados y manejar las cantidades de los productos que se encuentren dentro del inventario, y un módulo para obtener las ventas del día, de la semana o del mes, es decir, este módulo permitirá realizar filtros por fechas en donde el usuario ingresa unas fechas y si existen facturas dentro de esta fecha se puede generar el reporte.

Capítulo 1

Planeación

1.1. Planteamiento del Problema

El problema consiste en que al momento de realizar inventarios se hace necesario llevar a cabo un conteo de las unidades para anotarlas en un cuaderno o manual de inventarios, de igual manera restar las unidades vendidas dependiendo el producto y posteriormente este sera el valor numerico que se encuentra en stock.

Por otra parte se necesitaba hacer diario un registro de ventas y de productos, con la necesidad de que este no se podia acumular por que si se dejaba de hacer por unos dias se convertia en una lista de productos que debian salir del inventario causando errores que pueden llevar a afectar las cantidades en inventario con las unidades en stock.

1.2. Justificación

Es valido resolver este problema debido a que con esta solución de software se podra realizar inventario mensual, semanal y diario con una sola tirilla de productos y sus totales vendidos, de igual forma ya no sera necesario la facturación a mano disminuyendo el uso de papel y el porcentaje de error debido a trabajo humano.

Por otra parte se pueden observar con menor dedicación de tiempo parametros como ventas diarias, semanales, mensuales, las cuales se revisan para plantear metas y otros parametros como ventas por usuario las cuales pueden proporcionar informacion de rendimientos.

Por otra parte el modulo de contabilidad proporcionará un reporte conciso el cual sera usado por el usuario de contabilidad o el usuario administrador para revisar las ventas dentro de las fechas establecidas.

1.3. Objetivos

1.3.1. Objetivo General

Diseñar y desarrollar un prototipo de software basado en django web framework para llevar el manejo de facturación, contabilidad e inventario en un almacén de productos para venta por unidad o al por mayor.

1.3.2. Objetivos Específicos

- Desarrollar una sistema de información para negocio.
- Implementar el patron MVC.
- Desarrollar el sistema de informacion mediante web framework django.

1.4. Alcances

El proyecto podrá ofrecer un aplicativo editable para diferentes tipos de clientes, es decir, diferentes almacenes. Cada aplicativo tendra la posibilidad de llevar modulos segun el cliente los solicite y ser editado al estilo del almacén.

Asimismo se podran agregar modulos con facilidad para generar nuevas soluciones para requerimientos a petición del cliente.

1.5. Limitaciones

El proyecto prestara unicamente soluciones para empresas basadas en productos de venta libre, manejo de facturacion e inventario. Este no podra imprimir facturas, unicamente se realizara mediante correo electronico el envio de facturas.

1.5.1. Marco Metodológico

La metodología **SCRUM** es un proceso de desarrollo de software con metodologia agíl y flexible. Su principal objetivo es maximizar la inversión y disminuir el tiempo de desarrollo [1].

Este proceso permite entregar cada fin de la iteración un prototipo funcional y crear expectativas para cada iteración. Esta metodologia permite obtener los siguientes beneficios.

1. **Cumplimiento de expectativas:** Esta metodología permite que el cliente genere unas expectativas y el arquitecto de scrum planea cada iteración y genera prioridades.
2. **Flexibilidad a cambios:** Esta metodología presenta una alta capacidad al cambio especialmente en los requerimientos.
3. **Reducción del Time to Market:** Se le puede dar uso a los diferentes prototipos y se puede entregar un prototipo parcial para realizar su evaluación.

La Tabla 1.1 presenta las diferentes actividades realizadas con base en las fases de la metodología SCRUM[1].

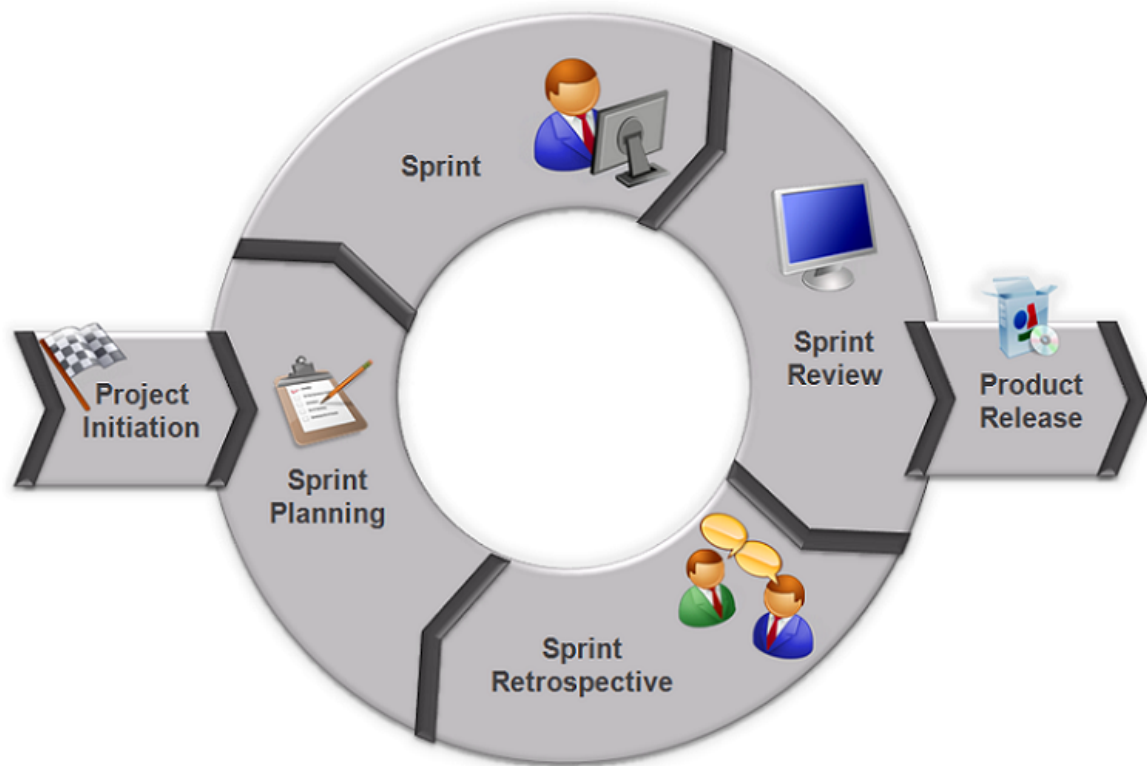
Tabla 1.1: Actividades durante un Sprint de SCRUM

Etapa	Actividad	Descripción
Planeamiento del sprint	Planeación del sprint	Durante la planeación del sprint se llaman a reunión todos los participantes del equipo, durante la consecución de este evento se decide que requerimientos o actividades van a ser asignados a los integrantes. Cada integrante deberá asignar un elemento de tiempo en el cual se va a realizar dicha actividad asignada. Esto permitirá obtener el tiempo en el cual se ejecutará el sprint.
Reunion de Equipo de Scrum	Reuniones diarias	<p>Cada día los elementos del equipo de scrum se reuniran con una duración no maxima a 15 minutos. Estas reuniones se llevaran a cabo en el mismo horario cada día, durante estas reuniones los miembros del equipo deberán responder las siguientes preguntas:</p> <ul style="list-style-type: none"> ▪ ¿Qué se hizo ayer? ▪ ¿Qué tiene planeado hacer hoy? ▪ ¿Qué obstáculos encontró en el camino? <p>Estas reuniones le permitiran a todos los miembros del equipo realizar una realimentación entre ellos como una manera de apoyo, en caso tal de que alguno tenga inconvenientes, se buscará la manera de resolver el obstáculo; de no ser encontrada una solución se deberá realizar una reunión para buscar solución a este obstáculo.</p>

Refinamiento del Backlog	Revisión del backlog	El director de scrum revisa cada elemento dentro del backlog, esto con el fin de esclarecer dudas que se presenten con respecto a las tareas que tiene el equipo de trabajo. Esto tambien permite recalcular el tiempo y esfuerzo que se va a dedicar para el sprint, y si es necesario ingresar nuevas actividades al backlog.
Revisión del Sprint	Revisión del sprint entre miembros	Los miembros del equipo de desarrollo y los clientes se reúnen para mostrar el prototipo que se obtuvo durante el sprint. Durante esta reunion se realiza una prueba de todos los desarrollos funcionales que se han finalizado. Esta presentacion estará a cargo del Scrum Master y el Product Owner.
Retrospectiva del Sprint	Realimentación del Sprint	<p>Durante esta actividad el Product Owner se reunirá con todo el equipo para dialogar sobre lo que se presente durante el sprint, los puntos principales para esta reunión serán:</p> <ul style="list-style-type: none"> ▪ Que cosas se hicieron mal durante el sprint, esto para poder mejorar el proximo ▪ Que cosas se hicieron bien para continuar realizando las actividades de esta manera. ▪ Que inconvenientes se encontraron y no permitieron avanzar como se planifico.

Ciclo del scrum, tomado de Manage your complex projects using Agile and Scrum - Scrum Master workshop[2].

Figura 1.1: Ciclo de Scrum



1.6. Factibilidad

1.6.1. Factibilidad Técnica

Para el desarrollo e implementación del proyecto es necesario contar con los siguientes dispositivos:

- Equipo de computo para desarrollo de aplicaciones.
- Equipo de computo para ejecutar el proyecto.

Así mismo, es necesario contar con los siguientes recursos de software:

- python +3
- MySQL
- Django
- Navegador web

1.6.2. Factibilidad Operativa

Este proyecto será elaborado por un estudiante de Ingeniería de Software I, el cual cuenta con el conocimientos previos en las herramientas python 3, django y MySQL.

1.6.3. Factibilidad Económica

El costo del proyecto se presenta con base en:

1. Costo de los equipos necesarios para el diseño y desarrollo de los prototipos. El costo se desglosa en la Tabla 1.2
2. Costo de mano de obra de los integrantes. El costo se desglosa en la Tabla 1.3.

Tabla 1.2: Costo de Recursos

Recurso	Cantidad	Valor Unitario	Valor total
Equipos de Computo	2	\$1.000.000	\$2.000.000
Total			\$2.000.000

Tabla 1.3: Costo de Integrantes del Proyecto

Integrante	Valor hora	Horas	Valor total
Jorge Mario Guaqueta Restrepo	\$50.000	100	\$5.000.000
Total			\$5.000.000

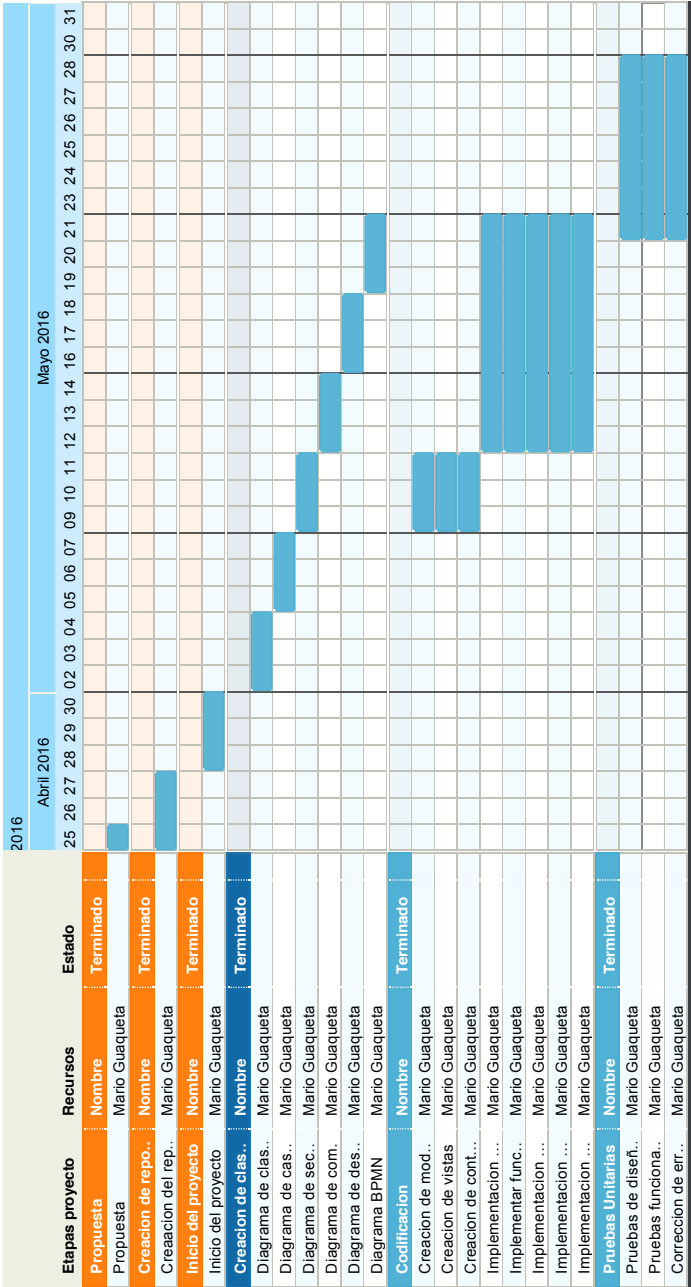
1.6.4. Factibilidad Legal

Se utiliza software libre para el desarrollo del sistema de información y para la base de datos, por lo cual se puede aplicar sin generar costos adicionales.

1.7. Cronograma

La Figura 1.2 presenta el cronograma para el desarrollo de la aplicación.

Figura 1.2: Cronograma



Capítulo 2

Análisis

2.1. Análisis de Requerimientos

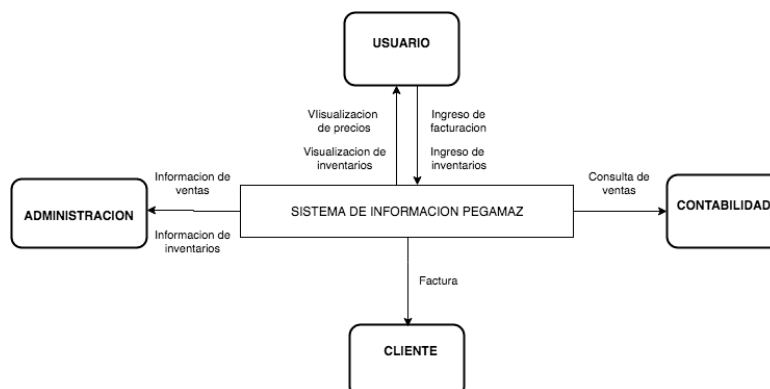
Este proyecto permite que la empresa PegaMaz lleve una facturación de forma digital, en donde los usuarios podran almacenar información del cliente, información de la compra y realizar una factura para las ventas.

Por otra parte se permitira que los usuarios realicen una busqueda de productos para facturarlos, un modulo de inventario en donde el usuario podra revisar la cantidad de unidades y por último un modulo de ventas en el cual se podra observar las ventas con un filtro de fecha.

La interacción de los usuarios con los modulos contruidos se presenta en la Figura 2.1, la cual describe los siguientes pasos.

- **Usuario:** El usuario es quien realiza los procesos de facturación, ingresa nuevos productos que vienen de los proveedores, al mismo tiempo que visualiza la facturación para informar al cliente y visualización de inventarios para conteos fisicos.
- **Contabilidad:** El usuario de contabilidad podrá visualizar todos lo elementos relacionados con la facturación y las ventas.
- **Cliente:** Recibirá la facturación mediante un correo electrónico.
- **Administrador:** Tendra la facultar de movilizarse por todos los modulos, al igual que podra ingresar productos a la lista de inventario y podra revisar contabilidad.

Figura 2.1: Diagrama de contexto



2.1.1. Requerimientos

Requerimientos Funcionales

Los requerimientos funcionales describen los diferentes módulos que el proyecto va a contener, La Tabla 2.1 presenta los requerimientos funcionales del sistema.

Requerimientos No Funcionales

Los requerimientos no funcionales describen diferentes características adicionales que el proyecto debe satisfacer para poder llevar a cabo los servicios que provee. La tabla 2.2 presenta los requerimientos no funcionales del sistema.

2.2. Definición de Actores

Con base en el análisis de requerimientos realizado, se establecen los actores que se presentan en la Tabla 2.3.

Tabla 2.1: Tabla de requerimientos funcionales

Requerimientos funcionales			
Identificador	Nombre	Descripción	Actores
Requerimiento 1	Ingreso	El sistema deberá permitir el ingreso de un usuario mediante su nombre de usuario y contraseña, realizando una validación	<ul style="list-style-type: none"> ■ Administrador ■ Contador ■ Usuario
Requerimiento 2	Registro	El sistema deberá permitir, mediante una contraseña de administrador, registrar un usuario el cual podrá tener un nombre de usuario, una contraseña, un email y un rol.	<ul style="list-style-type: none"> ■ Administrador
Requerimiento 3	Recuperación de contraseña	El usuario tendrá un email asignado para su recuperación de contraseña, la cual sera generada por el sistema.	<ul style="list-style-type: none"> ■ Administrador ■ Contador ■ Usuario
Requerimiento 4	Modulo de facturación	<p>Este modulo permitirá que el usuario realice una factura para el cliente con los siguientes aspectos:</p> <ul style="list-style-type: none"> ■ Subtotal ■ Iva ■ Valor final 	<ul style="list-style-type: none"> ■ Administrador ■ Usuario
Requerimiento 5	Modulo de inventarios	<p>Por otra parte, permitirá buscar productos en el inventario para ser agregados a la factura, al igual que ingresar un código para agregar el producto.</p> <p>Este modulo permitirá que el usuario realice una revisión de los inventarios con finalidad de solicitud de productos a los proveedores y conteo fisico.</p>	<ul style="list-style-type: none"> ■ Administrador ■ Usuario
Requerimiento 5	Modulo de contabilidad	Este modulo permitirá que el usuario realice una revisión de ventas por fechas.	<ul style="list-style-type: none"> ■ Contador

Tabla 2.2: Tabla de requerimientos no funcionales

Requerimientos no funcionales		
Identificador	Nombre	Descripción
Requerimiento no funcional 1	Base de datos MySQL	El sistema debera almacenar la información en una base de datos local.
Requerimiento no funcional 2	Navegador web	El equipo en el cual se ejecute el sistema de información debe contar con un navegador web para ejecutar el sistema de información.

Tabla 2.3: Tabla de definición de los actores

Actores		
Nombre	Descripción	Atributos
Administrador	Actor encargado de la administración del sistema, el cual tendra la posibilidad de modificar todos campos de los usuarios, gestionará usuarios, eliminacion de productos en inventarios, busqueda de ventas y eliminación de ventas.	<ul style="list-style-type: none"> ▪ Nombre ▪ Apellido ▪ email ▪ Telefono ▪ Rol
Usuario	Actor encargado de la facturación y la revisión de inventarios y actualización de inventarios.	<ul style="list-style-type: none"> ▪ Nombre ▪ Apellido ▪ email ▪ Telefono ▪ Rol
Contador	Actor encargado de la revisión de ventas para el manejo de la contabilidad.	<ul style="list-style-type: none"> ▪ Nombre ▪ Apellido ▪ email ▪ Telefono ▪ Rol

Capítulo 3

Diseño

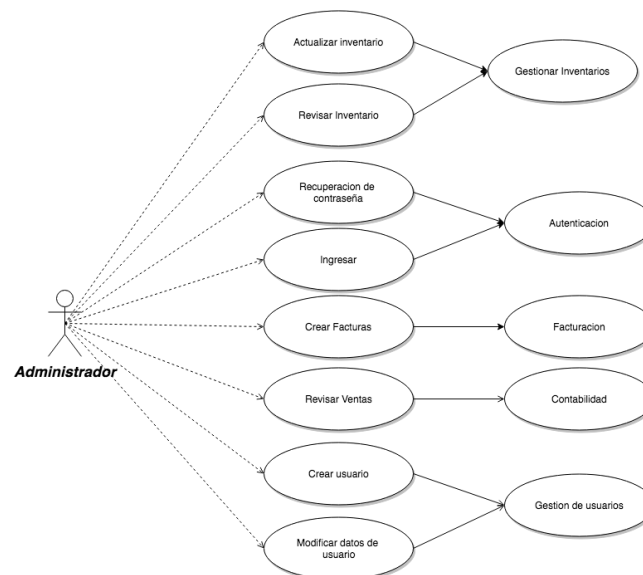
3.1. Casos de Uso

3.1.1. Diagramas de Casos de Uso

A través de los diagramas de uso se puede observar los servicios que ofrecerá el sistema.

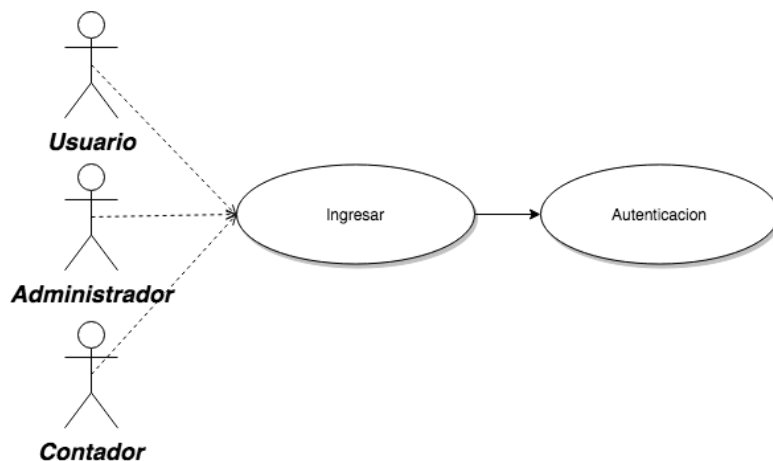
La Figura 3.1 presenta el diagrama de casos de uso para la gestión del administrador.

Figura 3.1: Diagrama de Casos de Uso de Administrador



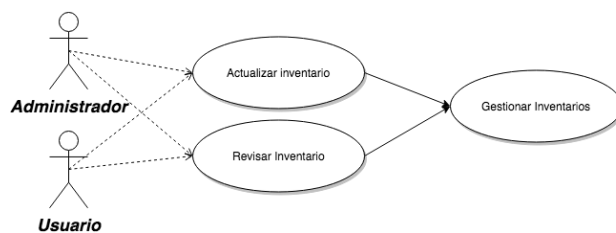
La Figura 3.2 presenta el diagrama de casos de uso para la gestión de las sesiones.

Figura 3.2: Diagrama de Casos de Uso de Ingreso



La Figura 3.3 presenta el diagrama de casos de uso para la gestión de inventarios.

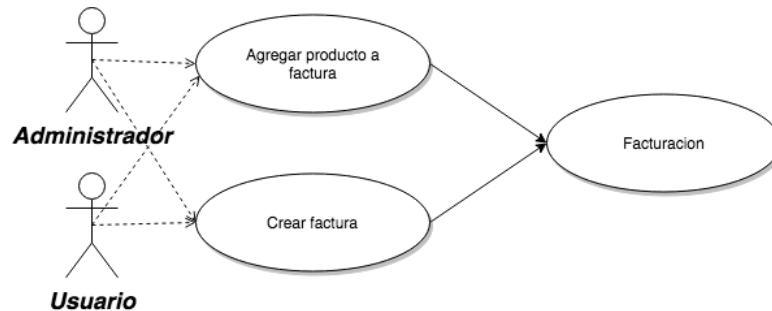
Figura 3.3: Diagrama de Casos de Uso de Inventario



Text

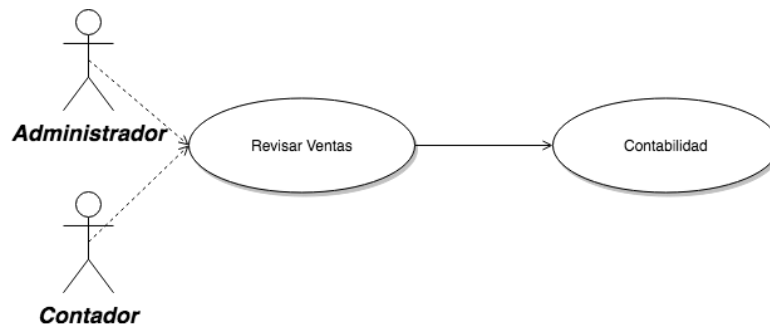
La Figura 3.4 presenta el diagrama de casos de uso para la gestión de facturación.

Figura 3.4: Diagrama de Casos de Uso de Facturación



La Figura 3.5 presenta el diagrama de casos de uso para la gestión de la contabilidad.

Figura 3.5: Diagrama de Casos de Uso de Contabilidad



3.1.2. Documentación de casos de uso

En la siguiente descripción se encontrarán las características propias de cada caso, al igual que el flujo de interacción de cada uno de ellos, de igual forma el flujo de eventos entre el actor y el sistema. La tabla 3.1 describe los casos de uso del sistema.

Tabla 3.1: Documentación de Caso de Uso

ID	Caso de uso 1
Nombre	Administrador
Descripción	El usuario de administración podrá interactuar con todos los módulos del sistema
Actor	Administrador
Precondición	El rol del usuario debe ser administrador
Poscondición	El usuario puede interactuar con todos los módulos del sistema

Eventos	Actor	Sistema
	1. Ingresa los datos de usuario	
		2. Valida la información
		3. Ingresa la información
		4. Muestra todos los modulos del sistema
Excepción	Ninguno	
Autor	Mario Guaqueta	
Fecha	Abril de 2017	
ID	Caso de uso 2	
Nombre	Ingreso	
Descripción	El usuario ingresa los datos para iniciar la sesión en el sistema.	
Actor	Administrador, Usuario, Contabilidad	
Precondicion	El usuario ingresará los datos para inciar la sesión.	
Poscondicion	El usuario podra ver los modulos dependiendo el rol.	
Eventos	Actor	Sistema
	1. El usuario ingresa los datos	
		2. Valida información del usuario
		3. Inicia sesión en caso exitoso
		4. Muestra los modulos dependiendo el rol
Excepción	Si los datos no son correctos no inicia sesión	
Autor	Mario Guaqueta	
Fecha	Abril de 2017	
ID	Caso de uso 3	
Nombre	Manejo de inventarios	
Descripción	El Administrador agrega o elimina productos al ivnentario, el usuario agregar productos a el inventario.	
Actor	Administrador, Usuario	
Precondicion	El administrador y el usuario deben estar autenticados.	
Poscondicion	El inventario es modificado	
Eventos	Actor	Sistema
	1. Agrega o elimina producto al inventario	
		2. Verifica que no exista o que exista dependiendo el caso
	3. Acepta agregar el producto e ingresa las cantidades	
		4. Almacena el producto en el inventario

Excepción	Si el producto no existe no puede ser eliminado, si el producto ya existe se suman las cantidades a las existentes.	
Autor	Mario Guaqueta	
Fecha	Abril de 2017	
ID	Caso de uso 4	
Nombre	Facturación	
Descripción	El Administrador o el usuario agrega productos a la factura y procede a generar factura.	
Actor	Administrador, Usuario	
Precondicion	El administrador y el usuario deben estar autenticados, los productos deben tener existencias en el inventario.	
Poscondicion	Se agregan productos a la factura	
Eventos	Actor	Sistema
	1. Agrega o elimina producto a la factura	
		2. Verifica que si exista el producto, verifica que las cantidades sean mayores a las solicitadas.
		3. Agrega las cantidades a la factura
	4. Debe generar factura	
		5. Debe eliminar las cantidades de la base de datos
Excepción	Si no hay cantidades del producto este no sera agregado a la factura, o si las cantidades solicitadas son mayores tampoco sera agregado al sistema.	
Autor	Mario Guaqueta	
Fecha	Abril de 2017	
ID	Caso de uso 5	
Nombre	Contabilidad	
Descripción	El Administrador o el usuario de contabilidad podran generar un reporte de ventas para su posterior revisión.	
Actor	Administrador, Usuario de contabilidad	
Precondicion	El administrador y el usuario deben estar autenticados.	
Poscondicion	El usuario debe agregar las fechas en las cuales desea ver el reporte.	

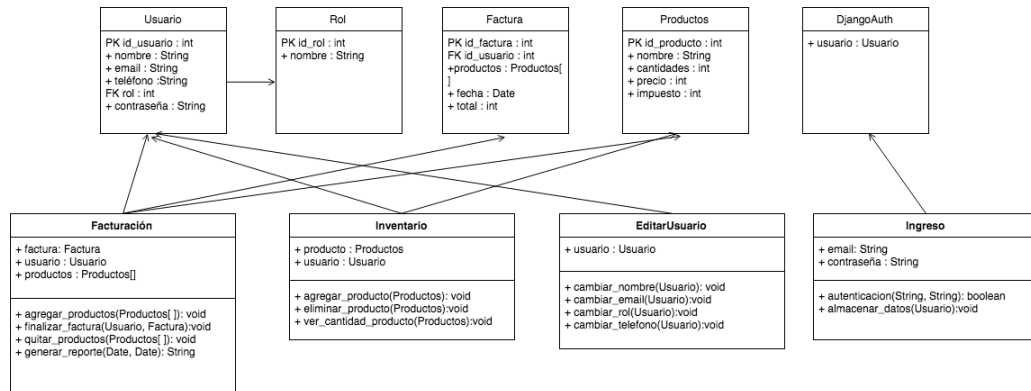
Eventos	Actor	Sistema
	1. Agrega las fechas de inicio y fin	
		2. Busca las ventas entre los rangos.
		3. Genera el reporte
	4. Decide que hacer con el reporte	
Excepción	Si no hay ventas en las fechas de filtro no generará nada.	
Autor	Mario Guaqueta	
Fecha	Abril de 2017	

3.2. Diagrama de Clases

Con base en los casos de uso y su documentación se crea el diagrama de clases que se presenta en la Figura 3.6. El diagrama de clases cuenta con las siguientes clases:

- **Usuario.** Esta clase representa los atributos del usuario.
- **Rol.** Esta clase representa los atributos del rol de cada usuario.
- **Facturas.** Esta clase representa los atributos de las facturas.
- **Productos.** Esta clase representa los atributos de los productos.
- **Djangoauth.** Esta clase representa la clase authentication de Django, la cual es una contribucion con el paquete.
- **Facturación.** Esta clase contiene los metodos y atributos que son utilizados al momento de realizar una factura.
- **Inventario.** Esta clase representa los atributos que son necesarios al momento de verificar los inventarios.
- **Editar usuario.** Esta clase representa los atributos necesarios y los servicios al momento de que un administrador va a editar un usuario.
- **Ingreso.** Esta clase representa los atributos necesarios para que un usuario pueda acceder a utilizar el sistema de información.

Figura 3.6: Diagrama de Clases



3.3. Diagramas de Secuencia

En las figuras 3.7, 3.8, 3.9 se observa la secuencia que sigue el proceso al momento de que el usuario, dependiendo de su rol, realiza una acción relacionada a su rol. Por otra parte se observan las secuencias que seguidamente el sistema al momento de realizar cada acción.

En el sistema se encuentran tres tipos de secuencia para cada tipo de usuario dependiendo su rol.

3.4. Diagramas de Componentes

Para el diagrama de componentes se visualizan los componentes que se encuentran en el sistema, los cuales son modificados por cada usuario y lo que puede realizar el usuario en el componente. De igual forma se realiza una relación de como funciona el servicio.

En la figura 3.10 se aprecia el diagrama de componentes del cual está hecho el sistema de información, incluyendo su capa de persistencia.

3.5. Diagrama de Despliegue

Para el diagrama de despliegue, que se observa en la figura 3.11, se encuentran los componentes del sistema, como el sistema va a ser ejecutado desde el ordenador y no hay necesidad de implementar servicios únicamente usar los servicios proporcionados por las contribuciones de django.

3.6. Diagrama BPMN

Para los diagramas de BPMN se observan cuatro diagramas, los cuales muestran el flujo de los procesos de negocio, para cada actividad realizada y el usuario encargado.

En la figura 3.12 se observa el diagrama BPMN para la actividad de facturación en la cual el usuario puede ser el usuario normal o el administrador, involucrando otro actor denominado cliente, el cual da inicio a la actividad. Al momento de agregar el producto a la factura se valida si hay existencias de este producto y las existencias que se tengan sean mayores o iguales a las necesarias por el cliente.

En la figura 3.13 se observa el diagrama BPMN para la actividad de manejo de inventarios, en donde el usuario puede ser el usuario normal o el administrador, el cual puede realizar modificaciones en los productos del inventario o agregar productos nuevos que lleguen. Las modificaciones a los productos deben hacerse siempre y cuando los productos existan dentro del sistema, si este no existe, no podrá ser modificado.

En la figura 3.14 se observa el diagrama BPMN para la actividad de edición de datos de usuarios, en donde el administrador es el encargado de realizar estas modificaciones, siempre y cuando el usuario al cual se va a modificar exista.

En la figura 3.15 se observa el diagrama BPMN para la actividad de generación del reporte de contabilidad en donde el usuario que realiza la actividad es el usuario de contabilidad o el administrador, y puede generar este reporte dentro de un rango de fechas siempre y cuando existan facturas en estas facturas.

Figura 3.7: Diagrama de secuencia del administrador

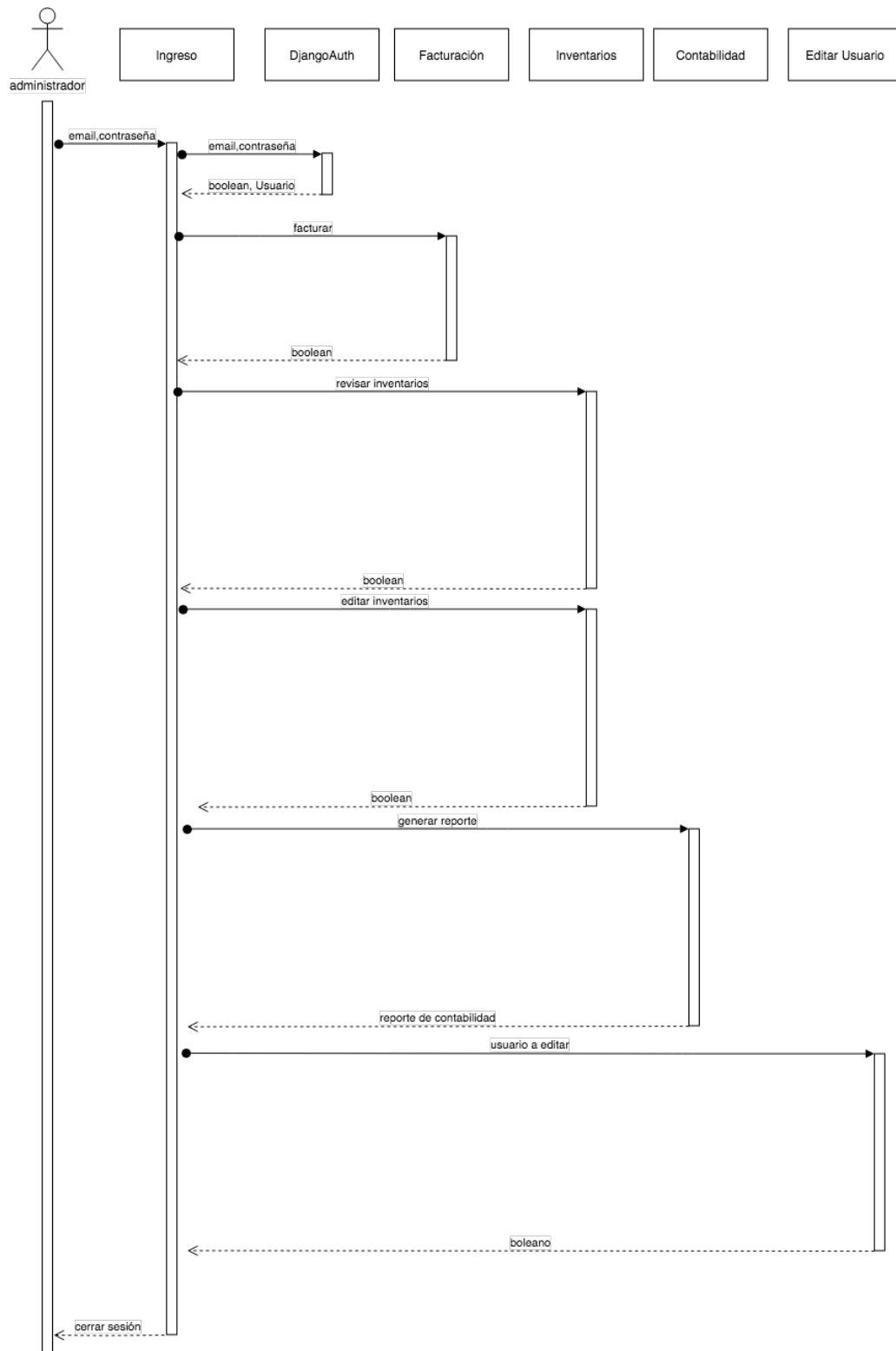


Figura 3.8: Diagrama de secuencia del usuario

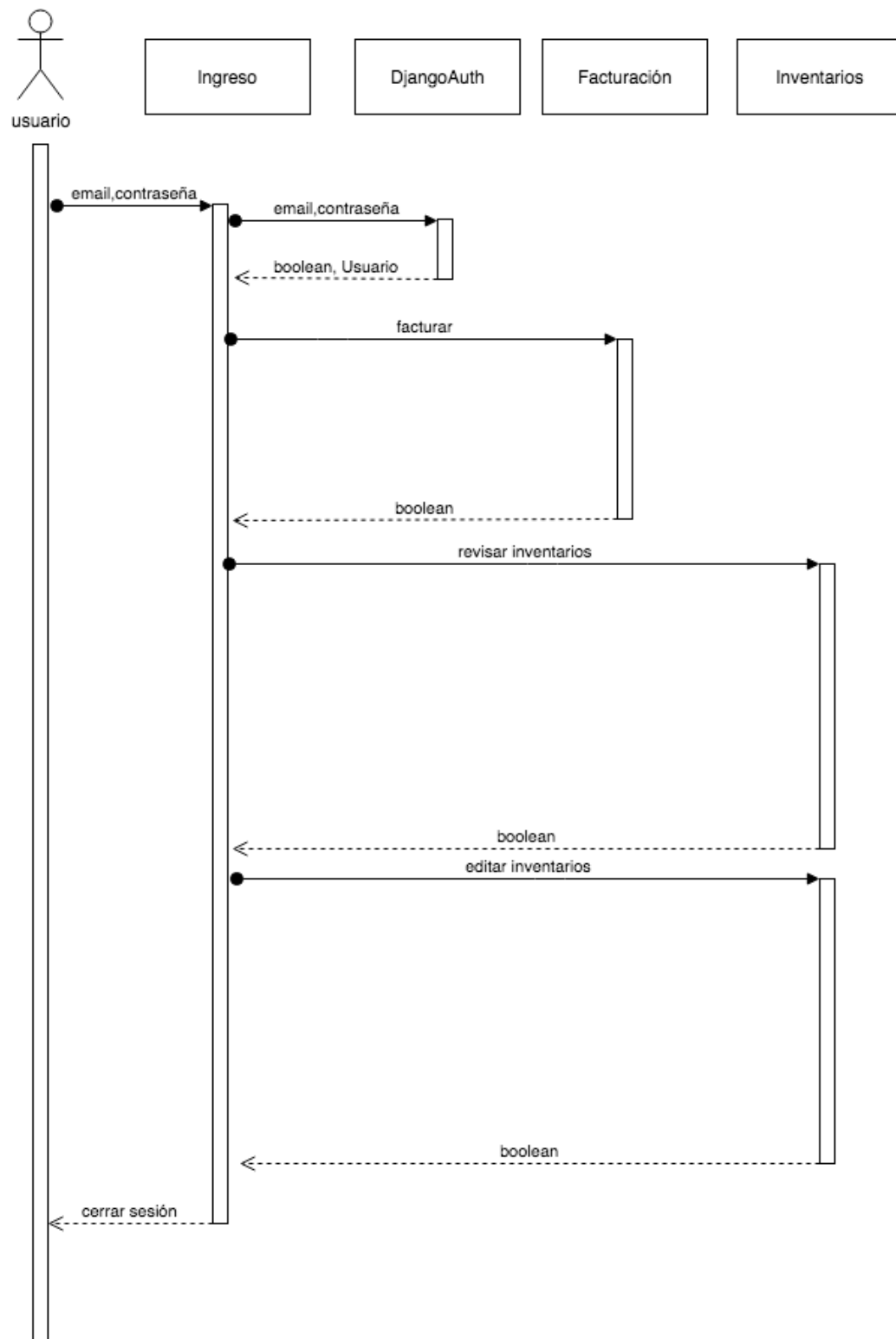


Figura 3.9: Diagrama de secuencia del usuario de contabilidad

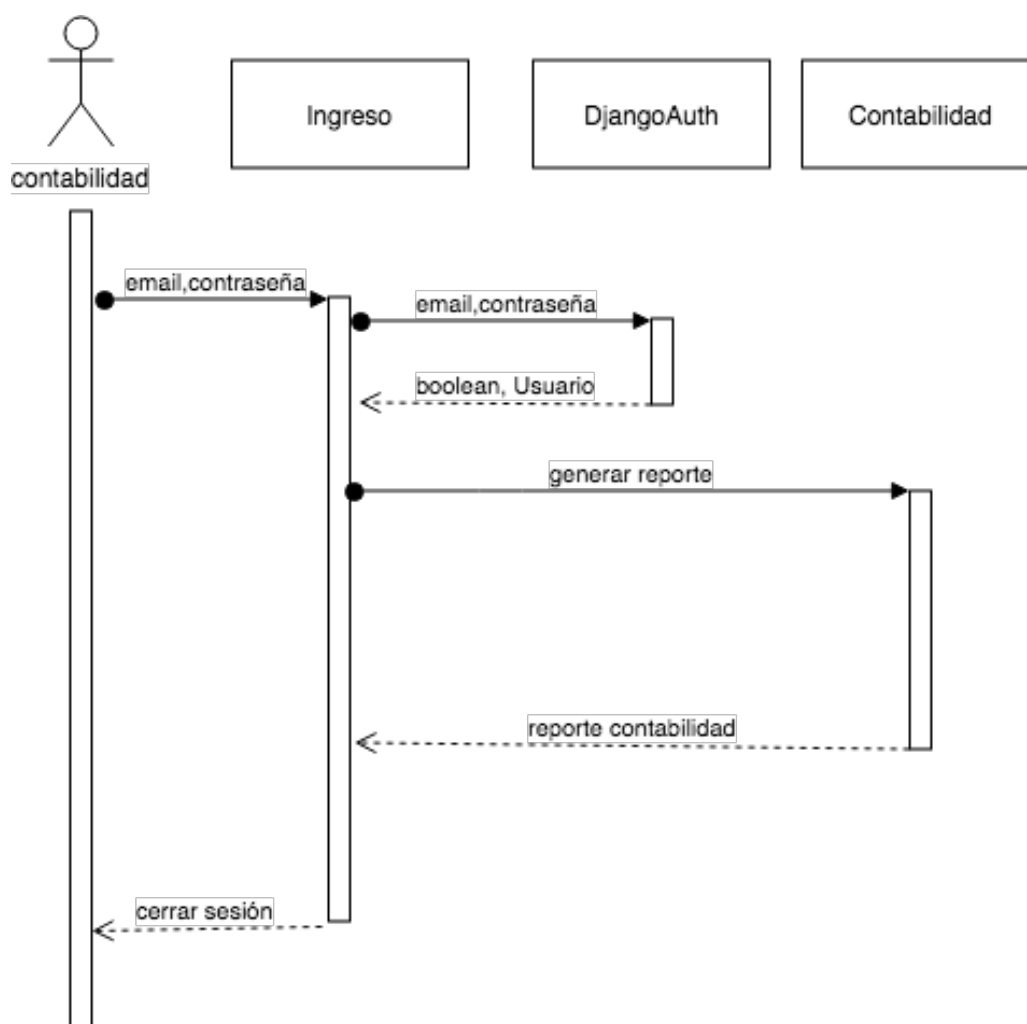


Figura 3.10: Diagrama de componentes del sistema de información.

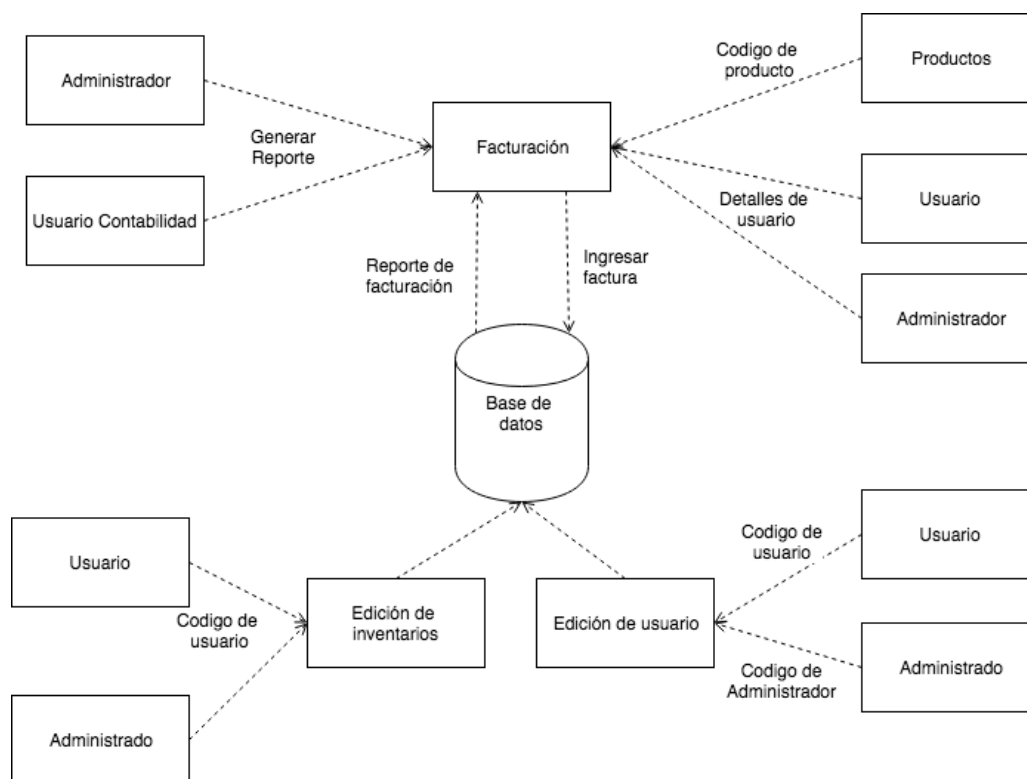


Figura 3.11: Diagrama de despliegue del sistema de información.

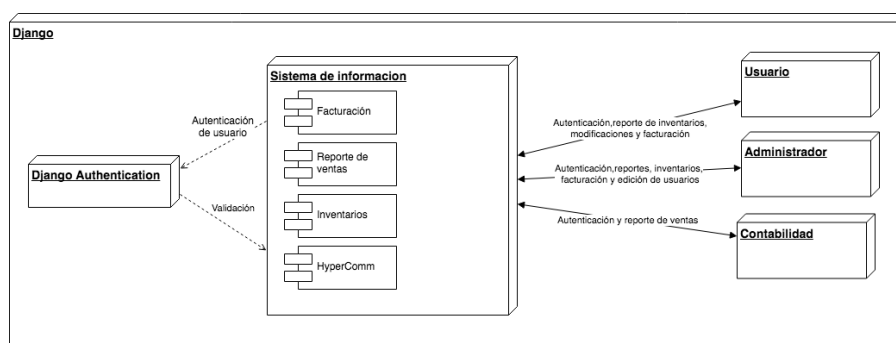


Figura 3.12: Diagrama BPMN del proceso de facturación

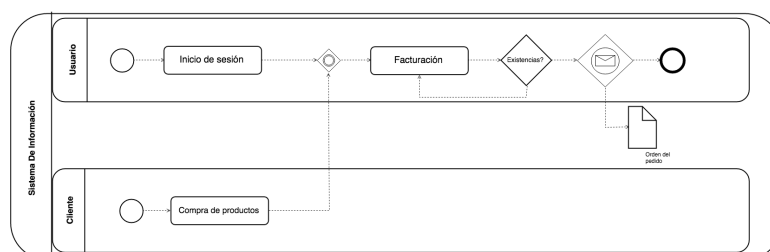


Figura 3.13: Diagrama BPMN del proceso de manejo de inventarios.

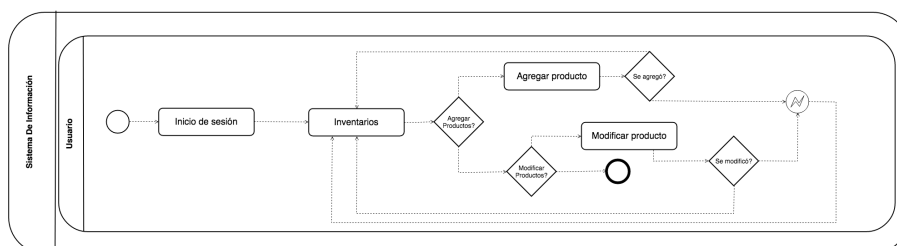


Figura 3.14: Diagrama BPMN del proceso de edición de usuario.

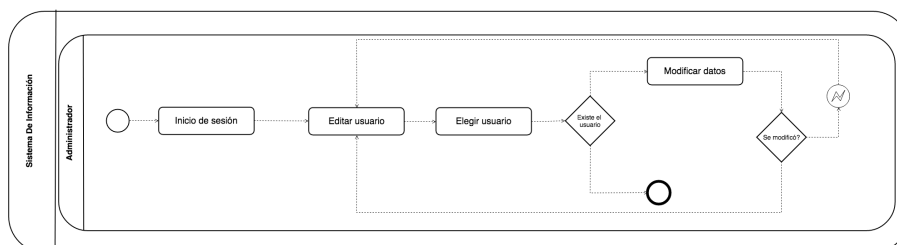
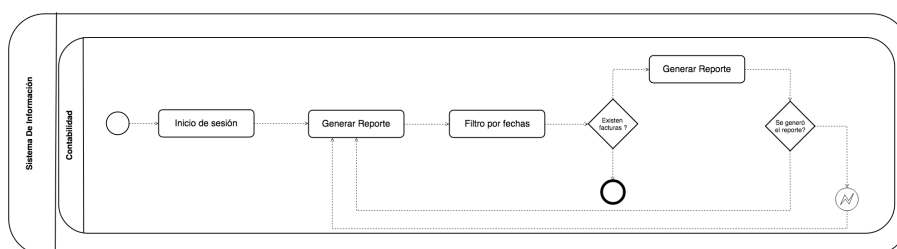


Figura 3.15: Diagrama BPMN del proceso de generacion del reporte de contabilidad.



Recomendaciones

Conclusiones

- Utilizar Django Web Framework como herramienta para el desarrollo del sistema de información, ya que esta herramienta permite realizar modificaciones a los modelos de manera rápida, permite utilizar plantillas para ser adaptadas a los proyectos y a su necesidad.
- Utilizar MySQL como administrador de bases de datos, ya que es un servicio implementable en Django y fácilmente utilizable, con bajos costos de utilización.

Bibliografía

- [1] W. Lara, “Scrum: Cómo funciona la metodología de trabajo scrum.”
- [2] SOFTTENG, “Metodología scrum para desarrollo de software aplicaciones complejas.”