Sistema automatizado altamente usable para punto de venta en mini mercados

Rubén Darío Andrade Manrique

Universidad Santo Tomás VUAD

Docente asesor

Pablo Emilio Cuenca Rivera

Trabajo presentado como requisito de grado para optar por el título de

Ingeniero Informático

Facultad de Ingeniería Informática virtual abierta y a distancia VUAD

Programa de Ingeniería Informática

Bogotá, 20 de mayo de 2017

A mis padres, hermanos, esposa e hija, que

han contribuido a mi realización personal

y profesional.

AGRADECIMIENTOS

Quiero expresar mi agradecimiento con los tutores Mario Dustano Contreras Castro y Pablo Emilio Cuenca Rivera, por el apoyo y lineamiento en este trabajo de grado, así como a cada uno de los profesores que fueron parte de mi formación profesional en la universidad Santo Tomás y universidad San Martin. Cada uno de ellos con sus consejos y catedra contribuyeron en parte para formar al profesional que soy hoy.

También quiero expresar mi agradecimiento a esa ayuda incondicional de mi esposa, padres y hermanos, los cuales me han apoyado en cada momento bueno y malo de mi carrera profesional, académica y personal.

Por último, agradecer al supermercado EL DORADO, por la total disposición, confiabilidad y paciencia durante todo el proceso de implementación de la solución informática a la que dio lugar el desarrollo de este trabajo de grado.

Nota de aceptación

_____

_____

_____

_____

_____

_____

_____

Firma del presidente del jurado

_____

Firma del jurado

_____

Firma del jurado

_____

Firma del jurado

Bogotá DC __/__/2017

Tabla de Contenido

INTRODUCCIÓN 8
Resumen 9
ABSTRACT 11
1 – Datos generales del proyecto 12
1.1 Título descriptivo del proyecto 13
1.2 Descripción del problema informático 13
1.2.1 Usabilidad 14
1.2.2 Conectividad 15
1.2.3 Automatización 15
1.3 Formulación del problema 20
1.4 Planteamiento del problema 20
1.5 Objetivos 21
1.5.1 Objetivo general 21
1.5.2 Objetivos específicos 21
1.6 Requisitos previos 22
1.7 Justificación 22
1.8 Alcances y delimitaciones 24
1.8.1 Alcances 24
1.8.2 Delimitaciones 24
1.9 Viabilidad técnica 24
1.10 Viabilidad económica 26
1.11 Viabilidad de gestión 27
1.12 Cronograma de actividades 29
1.12.1 Duración en horas 29
1.12.2 Duración en semanas 31
2- Marco de referencia 33
2.1 Antecedentes 34
2.2 Marco teórico 36
2.2.1 Usabilidad 36
2.2.2 Automatización 37

2.2.3 Conectividad 37
2.2.4 Arquitectura de software 38
2.2.5 Metodologías de desarrollo 44
2.3 Marco conceptual 51
2.3.1 Sistemas POS 51
2.3.2 Rapidez en la operación del sistema 51
2.3.3 Simplicidad 51
2.3.4 Algoritmos informáticos 52
2.3.5 Requerimientos funcionales y no funcionales 53
2.3.6 Independización de responsabilidades 54
2.3.7 Levantamiento de la información 54
2.3.8 Requisitos de software 55
2.3.9 Compatibilidad 55
2.4 Marco metodológico 56
3- Ingeniería del proyecto 59
3.1 Modelo del dominio 60
3.2 Modelo de casos de uso 61
3.2.1 Caso de uso: ADM – 001 64
3.2.2 Caso de uso: ADM – 002 67
3.2.3 Caso de uso: ADM – 003 71
3.2.4 Caso de uso: ADM – 004 75
3.2.5 Caso de uso: ADM – 005 78
3.2.6 Caso de uso: ADM – 006 80
3.2.7 Caso de uso: BUS- 001 82
3.2.8 Caso de uso: BUS- 002 85
3.2.9 Caso de uso: BUS- 003 88
3.2.10 Caso de uso: BUS- 004 92
3.2.11 Caso de uso: BUS- 005 96
3.2.12 Caso de uso: BUS- 006 100
3.2.13 Caso de uso: BUS- 007 108
3.2.14 Caso de uso: BUS- 008 113
3.3 Modelo entidad relación 120

3.3.1 Modelo entidad relación administración del sistema 121
3.3.2 Modelo entidad relación negocio del sistema 123
3.4 Modelo de clases 127
3.4.1 ORM dominio seguridad 128
3.4.2 ORM dominio negocio 130
3.5 Modelo arquitectonico 131
3.5.1 Módulo de seguridad y autenticación 132
3.5.2 Módulo lógico de administración del sistema 132
3.5.3 Módulo lógico base POS 133
3.5.4 Módulo lógico ventas 133
3.5.5 Módulo lógico de inventario 134
3.6 Arquitectura de software 135
3.6.1 El lenguaje ubicuo 136
3.6.2 Capas arquitectónicas 137
3.6.3 Capa de presentación 138
3.6.4 Capa de negocio 139
3.6.5 Capa de dominio 140
3.6.6 Capas de infraestructura 141
3.7 Metodología de desarrollo 142
3.7.1 Creación proyecto scrum 142
3.7.2 Dashboard general 142
3.7.3 Features proyecto 143
3.7.4 PBI proyecto 143
3.7.5 Finalización proyecto 147
3.8 Implementación 148
4- Análisis de resultados 153
4.1 Conclusiones 154
4.1.1 Acerca del objetivo general 154
4.1.2 Acerca de los objetivos específicos 156
4.2 Recomendaciones 156
Bibliografía y referencias 158

INTRODUCCIÓN

En la actualidad los sistemas informáticos son la base central para el manejo de la información involucrada en procesos empresariales, institucionales y personales; agilizando, respaldando y procesando los diferentes flujos de datos convertidos en información que resulta indispensable y útil en una sociedad digitalizada.

Garantizar la usabilidad, agilidad de utilización, conectividad, confiabilidad, seguridad, funcionalidad y explotación de los datos son los rasgos de cara a un usuario final que un sistema informático debe proveer, así mismo como una arquitectura y diseño en sus cimientos de codificación para asegurar la mantenibilidad, escalabilidad y correcto funcionamiento del sistema; las cuales son el eje central y pilar principal a aplicar en este proyecto.

Teniendo en cuenta estas premisas mencionadas, en este trabajo de grado presentado para optar por el título de ingeniero informático de la universidad Santo Tomás, se realizará el análisis e ingeniería de requerimientos que darán paso a la realización del desarrollo de software basado en modelos arquitectónicos de software y metodologías de desarrollo de un sistema informático tipo POS (Point of Sale/ Punto de venta) para mini-mercados, atendiendo principalmente a las necesidades informáticas para el mini-mercado EL DORADO, haciendo énfasis en mejoras relacionadas con la usabilidad, automatización de tareas y conectividad; contemplando aun así funcionalidades básicas de este tipo de sistemas como lo son el procesamiento de ventas, créditos, reportes, proveedores, seguridad y funcionalidad.

La propiedad intelectual es de autoría del estudiante quien amparado por el articulo 9 y 72 de la ley 23 de 1982, autoriza el uso y reproducción de esta documentación para fines académicos.

Resumen

El mini-mercado EL DORADO, objeto de estudio para la realización e implementación de este proyecto de grado, ofrece a su clientela una amplia variedad de artículos de primera necesidad, caracterizándose por ofrecer precios competitivos y una atención rápida; esto en parte al uso de sistemas informáticos tipo POS utilizados desde hace cinco años. Desde entonces se han adquirido tres diferentes herramientas informáticas tipo POS, y de las cuales se han identificado problemáticas relacionadas con la usabilidad del sistema en ciertas pantallas críticas, falta de automatización de tareas repetitivas y poca o nula conectividad para la explotación, migración o análisis de los datos del sistema; por ende se ha decidido enfocar este trabajo de grado, que en conjunto con el análisis, ingeniería de requerimientos, arquitectura de software, metodología de desarrollo y por ultimo programación de software, produzcan una herramienta tipo POS con las mejoras expuestas que satisfagan la problemática anteriormente mencionada.

Realizando este proyecto se adquirirán las competencias necesarias para la identificación de problemáticas en sistemas de información empresariales, demostrando capacidades de planteamiento y formulación de soluciones a este tipo de requerimientos, todo esto basado en un marco de trabajo sustentado. Con este análisis se procede a los siguientes interrogantes:

¿Se genera un valor agregado al diseñar, desarrollar e implementar un tipo de sistema POS con beneficios en su funcionalidad, abierto para la explotación de sus datos y conectividad, contemplando la utilización de algoritmos informáticos junto con pantallas altamente usables que proporcionen la automatización de tareas repetitivas para el tipo de negocios como los mini-mercados, tiendas al detal, pequeños distribuidores y negocios de atención y despacho de mercancías de flujo mínimo?

Palabras clave: POS, usabilidad, automatización, conectividad, detal (retail), Aplicaciones de una sola página SPA (single page application).

ABSTRACT

In order to realize this document's readers, the grocery store EL DORADO, target implementation store, for this bachelor degree aspiration work. EL DORADO is offering to its customers a wide kind of products ‘first need' mainly, featuring reasonable prices and a quick and agile attention, due mainly by using informatic POS systems, for five years ago. Since then EL DORADO has purchased two different POS of which have founded several issues related with the ease of use in such critical system forms, an automation left on repetitive task and a left or nonexistent to data system in order to exploit, migrate or leverage it. For that reason, has decide to focus this bachelor degree work, that in conjunction with an analysis, requirements engineering, software architecture, development methodology and lastly software development, to produce one POS system with the advantages that solve the above issue related

Developing this project, it will acquire the require skills to identify such issues at business process systems, showing abilities on planning and solving to these kind of issues, all of this supported by self for a framework.

As bottom line, we procced to ask ourselves: ¿we make the difference, when we design, developing and implementing a POS system with advantages on its functionality, open to data exploit and connectivity, using computer algorithm programs aside highly usable forms to make automation and ease use of repeated task for a wide kind of business like EL DORADO, grocery stores, small retailers and so on small business.

1 – Datos generales del proyecto

1.1 Título descriptivo del proyecto

Sistema automatizado altamente usable para punto de venta en mini mercados.

Contempla el análisis, ingeniería de requerimientos y programación de un sistema de información para el registro de ventas, inventarios, entradas y salidas de mercancías, ingreso de clientes, usuarios del sistema y reportes, teniendo en cuenta diseñar cada uno de los artefactos involucrados en pro de brindar una alta usabilidad, automatización en tareas repetitivas para el usuario y posibilidad de explotación de los datos del mini mercado EL DORADO,

1.2 Descripción del problema informático

Inicialmente cabe resaltar que el problema investigado, está respaldado en la formación recibida en el programa de ingeniería informática de la universidad Santo Tomás, y está relacionado con el núcleo problémico y los objetos de estudios definidos por el programa en aspectos como: capacidad de aplicar estrategias, técnicas de modelamiento, simulación de software, adaptación de software a la medida e ingeniería de software.

Actualmente la sistematización en mini mercados brinda a sus propietarios una serie de ventajas como son la rapidez en la consecución de información de precios, registros de los movimientos de ventas, inventarios actualizados, registro de ubicaciones, reportes que brindan información importante para la toma de decisiones entre otras.

El mini mercado EL DORADO ubicado en el noroccidente de la ciudad de Bogotá, objeto de estudio en el análisis de la información de este proyecto, así como destino de la implementación del producto final obtenido a través de la ingeniería de requerimientos, diseño y programación correspondiente; ha utilizado por más de 5 años diferentes soluciones de software que se ajustan a

este tipo de negocios y que son conocidos como software POS para suplir sus necesidades informáticas.

El mini mercado ha utilizado tres alternativas de software diferentes con el ánimo de encontrar una que se ajuste a sus necesidades. Estas soluciones son: Solvermedia ®, Software POS ® y tiendaTek ®. Las razones por las cuales no se ha establecido el uso de una única herramienta de software en particular, y que son el objetivo por conseguir con este proyecto de grado, son:

1.2.1 Usabilidad

La usabilidad es un conjunto de características que hacen el que el uso de una herramienta informática sea amigable, adaptable, intuitiva, permitiendo obtener resultados sin una intervención exhaustiva por parte de un usuario abriendo caminos al error humano o complicando el acceso/ingreso de la información de tal manera que un usuario final vea más como un obstáculo la solución de software en vez de una herramienta útil de trabajo. Especificando la mejora a desarrollar en este aspecto tomando como base las necesidades puntuales del mini mercado, así como de la experiencia de uso con las herramientas anteriormente mencionadas es facilitar el uso de la herramienta minimizando la navegación entre ventanas, facilitando el ingreso de ventas paralelas debido a la naturaleza del negocio y facilitar la búsqueda e ingreso de registros que la intervención del usuario necesite. Las soluciones anteriormente mencionadas carecen de la funcionalidad de ventas paralelas, registro de ítems por distintos factores de búsqueda como el texto explícito del producto y a su vez el código de barras de una manera usable, ingreso de créditos de una manera amigable y búsqueda de los ítems en el mini mercado de una manera ágil y rápida.

1.2.2 Conectividad

La mayoría de los sistemas que se han utilizado mantienen el almacenamiento de datos de forma cerrada, esto porque en su mayoría no dependen de modelos netamente relacionales sino de estructuras de datos alternas o por el estilo, en algunos casos basadas en archivos planos. El punto es que la conectividad y explotación del registro histórico y actual de los datos se condiciona en las opciones originalmente habilitadas por la herramienta de software sesgando o dificultando de manera abrupta el alcance que se le puede dar a este tipo de activo tan importante. Mencionando algunas de las restricciones más importantes se pueden destacar las migraciones a un sistema diferente, imposibilidad de reportes distintos a los expuestos, extensibilidad de la funcionalidad y exploración bruta de los datos.

1.2.3 Automatización

Un algoritmo es un conjunto finito de reglas bien definidas en su lógica de control, que permiten la solución de un problema en una cantidad finita de tiempo (Mancilla Herrera, Ebratt Gomez, & Capacho Portilla, 2015, pág. 2). La intervención humana en los sistemas informáticos es imprescindible ya que un algoritmo informático por más efectivo que sea nunca será tan capaz de tomar decisiones que contemplen demasiadas variables del entorno que en el caso particular, una persona fácilmente si podría. Sin embargo, no deja de ser inoficioso no poder automatizar tareas repetitivas o de un calibre de intervención humana bastante alto cuando se puede hacer uso de la informática para procesar cantidades importantes de registros o información apoyándose en algoritmos eficientes. Una de las ausencias en este tipo de sistemas es precisamente que han sido creados como una herramienta de procesamiento, facilitando el ingreso de datos en formularios o interfaces de usuario gráficas amigables, pero que caen siempre en el mismo ciclo de ingreso,

proceso, salida. Al no contar con una funcionalidad que analice este tipo de ciclo y poder sacar uso de algoritmos eficientes que analicen este tipo de comportamientos, se está desaprovechando inmensamente la capacidad de cómputo de una aplicación. Para el caso particular de la mejora propuesta en este proyecto, se pretende automatizar el pedido de productos, basado en el inventario y ventas históricas registradas para poder facilitar el ciclo de ingreso y faltantes, núcleo funcional de los sistemas tipo POS.

Con estas premisas se plantea la necesidad de desarrollar un sistema de software POS que cumpla con las funciones básicas ofrecidas actualmente por estos sistemas como son la administración de usuarios, permisos, navegabilidad entre otros y adicionalmente que cubra las necesidades puntuales anteriormente expuestas y que son el motivo del proyecto informático como tal.

Para obtener una mayor especificidad en la problemática recientemente expuesta se hace uso de la siguiente tabla de diagnóstico de la problemática y su ámbito en la necesidad puntual anteriormente expuesta.

Tabla 1

Tabla de diagnóstico de la problemática

Cuadro de diagnóstico				
Ámbito	Síntoma	Causa	Consecuencia	Constrol
Usabilidad	Se puede evidenciar la dificultad que tiene el empleado que atiende a los clientes cuando debe registrar ventas de forma paralela ya que al parecer la	Las funcionalidades expuestas para este tipo de manejo por los sistemas actuales obligan a realizar una serie de pasos para	Al tener que manipular la herramienta para tener acceso a esta funcionalidad tan importante en el	Es necesario que la funcionalidad de ventas paralelas tenga una usabilidad óptima basada en metodologías como

	ventana destinada para tal fin ha sido creada para ofrecer la funcionalidad para una fila o cola y atender las personas individualmente y una después de la otra. Aunque es un patrón general del proceso de venta, es necesario algunas veces despachar a un cliente distinto mientras el cliente inicial decide que desea comprar o cambiar algún producto.	poder levantar una ventana de venta nueva sin perder la actual y en algunos casos toca perder el registro actual para poder procesar uno nuevo en particular	momento que se desee hace que se caigan en errores de digitación además de demoras y pérdida de información volátil que hace que se tenga que caer en reprocesos repercutiendo en la atención al cliente.	SPA (Single page application, aplicaciones de una sola página) para que se pueda contar con una funcionalidad amplia sin necesidad de recaer en reprocesos innecesarios que a la larga se convierten en un servicio de mala calidad.
Usabilidad	Al registrar un ítem dentro del sistema bien sea para su ingreso en una venta, inventario o un proceso interno del establecimiento la mayoría de sistemas soportan la identificación del ítem basado en su código de barras y para tipos de productos que no manejan el código, el sistema permite codificarlos de forma personalizada. Sin embargo, no se cuenta con una opción explicita de	Los sistemas hasta ahora utilizados se basan en la utilización de código de barras y para cuando es inexistente estos hacen uso de una codificación similar manual lo cual satisface la parametrización e identificación del producto, pero recae en un reproceso porque los usuarios del sistema tienen que aprender una	Demoras en la consecución, registro y procesamiento de productos basados en su codificación personalizada. Reprocesos entre los diferentes momentos del ciclo de ingreso, análisis y respuesta de información del sistema. Forzar al operario del sistema a aprender un amplio	Es necesario contar con una entrada de datos homogénea y dinámica en toda la aplicación, para que el sistema esté en capacidad de recibir diferentes tipos de entrada, bien sea código de barras, codificación personalizada y entradas de texto relacionadas con el

	búsqueda por el nombre del producto lo cual hace que se pierda ese ámbito de búsqueda tan importante y de dominio transversal.	infinidad de códigos que no son estándares.	número de códigos que son de utilidad explicita del mini mercado.	producto
Usabilidad	Al ser solicitado un producto verbalmente no se cuenta con una funcionalidad que permita saber dónde se encuentra ese producto en el mini mercado de manera ágil ya que, aunque la mayoría de sistemas permiten vincular un producto con una ubicación específica, es preciso moverse a otra ventana u otra funcionalidad que indique el vínculo inicialmente mencionado ya que no se puede ubicar un producto por búsqueda de texto en las pantallas principales de venta sino únicamente por su código de barras.	No se ha pensado en un sistema íntegro que contempla varios escenarios como este donde un cliente busca una mercancía específica verbalmente y en donde el usuario del sistema desconoce su ubicación bien sea por inexperiencia o porque se trata de un producto nuevo del cual no tiene conocimiento.	Reprocesos a la hora de especificar o averiguar una ubicación donde un producto se pudiese encontrar y que no es de conocimiento tanto del personal ni del cliente que lo solicita, lo que se traduce en una mala atención al cliente y demoras en el servicio.	Realizar una funcionalidad que de manera ágil proporcione la funcionalidad específica del caso, mediante el uso de una parametrización programable y una visualización de las ubicaciones de manera ágil y que proporcionen una alta usabilidad del sistema.

Automatización Conectividad	Se evidencia un trabajo repetitivo a la hora de gestionar la programación de los pedidos dentro del sistema ya que es preciso realizar un análisis de las ventas que generalmente se encuentra en pantallas diferentes a la de las ventas lo que se hace que se tenga que ocupar el sistema por parte de quien realiza pedidos generalmente el administrador del mini mercado.	Los sistemas POS utilizados en el mini mercado EL DORADO son basados principalmente en aplicaciones de escritorio lo que conlleva a que no existe fácilmente la posibilidad de conectar un cliente y realizar dos tareas simultáneamente. A esto se le suma un factor importante y es la falencia de no tener toda la información necesaria en un solo lugar para la toma de decisiones por parte del usuario final.	Errores en la digitación de la información, faltantes en la información por diligenciar, faltantes en el inventario, excesos en el inventario, tareas repetitivas y que deben ser realizadas en horas en las que el sistema no se encuentra en uso de registro de ventas	Realizar una funcionalidad que realice un análisis del registro histórico de ventas y automatice tareas repetitivas como la programación de productos faltantes en él inventario brindando la información de forma completa para una intervención mínima del usuario. Además de facilitar el acceso a la funcionalidad desde un terminal diferente al del registro de ventas contemplando incluso accesos desde una red remota
Conectividad	No se cuenta con un repositorio de datos explorable o accesible fácilmente de cara al usuario final, administrador del mini mercado e incluso al usuario técnico para una eventual	La mayoría de estos sistemas persisten sus datos de manera implícita y no es un bien al que el usuario final pueda acceder fácilmente salvo en las	Imposibilidad de explotar los datos a otro nivel o incluso conectar el sistema a un diferente programa para poder hacer uso de funcionalidades	Realizar la persistencia de los datos de la aplicación en un motor de base de datos relacional que facilite tanto su explotación de la

	migración o análisis de los datos.	pantallas que el sistema diseño para tal utilidad. En muchos casos la persistencia de estos se realiza de manera privada, es decir, no es abierta ni estándar.	adicionales. Al tener este modelo de persistencia de los datos es una tarea tediosa incluso migrar a una herramienta diferente porque no se cuenta con una estandarización en este sentido.	información, así como su extracción y futura manipulación, obviamente contemplando la seguridad necesaria para garantizar el correcto funcionamiento del sistema.

1.3 Formulación del problema

El proceso recolección de la información necesaria para poder hacer el análisis, el diseño, y la ingeniería de requerimientos para el desarrollo e implementación del software tipo POS que satisfaga las necesidades puntuales expuestas en la descripción del problema, nos conduce a cuestionarnos objetivamente:

¿Cómo realizar el desarrollo a la medida de un sistema tipo POS, que proporcione características óptimas en cuanto a la usabilidad, conectividad, automatización y explotación de sus datos, para el mini mercado EL DORADO?

1.4 Planteamiento del problema

Con base en lo anterior se pretende en este proyecto realizar el análisis y levantamiento de la información que permita realizar una ingeniería de requerimientos y posteriormente el diseño, desarrollo e implementación de la solución de software que cumpla con las necesidades de usabilidad, automatización y conectividad expuestas.

Es preciso formular los siguientes interrogantes para precisar la formulación del problema:

¿Cómo garantizar que la interfaz de operación cumpla con la usabilidad pretendida y al mismo tiempo cumplir con la funcionalidad que se busca?

¿Cómo asegurar que existirá una conectividad abierta, que sea segura, y estable para que sistemas terceros consuman nuestros recursos de información de manera fiable y óptima?

¿Cómo garantizar una automatización a través de algoritmos eficientes asegurando una intervención mínima o nula por parte del usuario del sistema y que además cumpla con las necesidades explícitas en cuanto a diligenciamiento o procesamiento de información en un sistema POS?

1.5 Objetivos

1.5.1 Objetivo general

Desarrollar un sistema de información tipo POS que proporcione alta usabilidad, conectividad a sistemas terceros, automatización en tareas repetitivas de intervención de usuario, sea abierto a la explotación de sus datos, y que además cuente con funcionalidades básicas de seguridad y registro de ventas generando valor y mejora de utilización en el mini mercado EL DORADO.

1.5.2 Objetivos específicos

- Identificar las necesidades puntuales que el mini mercado El DORADO desea satisfacer con la realización de un sistema de información tipo POS a la medida.
- Analizar el estado en que se encuentra la herramienta POS del mini mercado EL DORADO junto con el entorno del negocio con el fin de ajustar las bases de información documental

sobre las cuales desarrollar el diseño de la aplicación sistematizada.

- Determinar toda la ingeniería de requerimientos necesaria que forme las bases del diseño, modelamiento, programación e implementación de la solución de software.
- Programar el aplicativo tipo POS a la medida de la necesidad del mini mercado EL DORADO, utilizando una tecnología adecuada, soportada en la ingeniería de requerimientos previamente desarrollada.

1.6 Requisitos previos

Para poder llevar a cabo la solución a los problemas mencionados en el punto anterior se necesita desarrollar primeramente y de manera organizada el análisis y diseño de los siguientes aspectos del sistema:

- Levantamiento de información de los procesos involucrados dentro del mini mercado EL DORADO.
- Determinar el alcance que tendrá la solución, cumpliendo con cada uno de los objetivos específicos y objetivo principal.
- Realizar un análisis previo que defina el alcance de la ingeniería de requerimientos para poder especificar los módulos y distintas funcionalidades del sistema.
- Definición de las tecnologías a utilizar, metodologías de desarrollo, gestores de bases de datos, entre otros elementos, basados en la experiencia y contexto del proceso.

1.7 Justificación

Desde la elección de mi carrera profesional en las tecnologías de la información ha sido de mi preferencia enfocarme en el área del desarrollo de software en la cual he tenido la gran

oportunidad de tener una experiencia de más de 6 años programando en diferentes lenguajes y tecnologías aplicadas en variedad de proyectos empresariales lo cual me ha dado una experiencia valiosa para aplicar mis conocimientos en diferentes problemáticas que involucren la sistematización es por ello que existe un interés tanto profesional en la realización de este proyecto, como académico para dar cumplimiento de sustentación como trabajo de grado para optar por el título de ingeniero informático.

La principal motivación y justificación es dar solución a las problemáticas halladas en los diferentes sistemas que ha utilizado el mini mercado EL DORADO, generando un valor agregado en el proceso sistemático de esta herramienta POS en el establecimiento.

Además, es evidente que los establecimientos de comercio al detal del tamaño o volumen de trabajo como el del mini mercado EL DORADO, tienen ciertas particularidades tales como la atención de clientes de manera no estructurada, poca motivación a la explotación de los datos registrados y demás focos relacionados con las herramientas POS que implementan o no implementan; pero no dejan de ser esquemas de negocios similares donde una herramienta informática adecuada puede marcar la diferencia en competitividad y agilidad en la toma de decisiones para el buen desempeño del negocio. Es por ello por lo que el desarrollo de este proyecto puede generar un alto impacto en negocios de este tipo que decidan implementar una solución con los beneficios que este proyecto ofrece y esto aplicaría para establecimientos comerciales de la ciudad e incluso a nivel nacional.

1.8 Alcances y delimitaciones

1.8.1 Alcances

Cumplir con el objetivo principal, así como sus objetivos específicos que de manera conjunta solucionarían las necesidades expuestas, es el ámbito principal de trabajo de este proyecto.

Los sistemas tipos POS del mercado tienen un amplio rango de servicios que complementan la solución final, tales como: pasarelas de pago, centralización de datos en la nube, acceso a funcionalidades desde dispositivos móviles entre otros, que no son objeto de este proyecto.

1.8.2 Delimitaciones

Analizando el alcance, que en gran parte es dedicado a satisfacer la problemática relacionada con la funcionalidad, usabilidad, conectividad y automatización de tareas repetitivas, y teniendo en cuenta lo mencionado, se proveerá un sistema POS delimitado a proporcionar los servicios de: gestión de usuarios, gestión de clientes, gestión de ventas, gestión de inventario, gestión de créditos, reportes específicos y seguridad del sistema. Todo esto en un entorno diseñado y analizado para trabajar on premises (instalación de programas como gestor de base de datos y servidor web) en la máquina a utilizar sin hacer uso explícito de acceso a internet para su funcionamiento.

1.9 Viabilidad técnica

Para materializar el proyecto “sistema automatizado altamente usable para mini mercados”, es preciso tener un aval relacionado con aspecto técnica del proyecto el cual precisa contar con un equipo o individuo que tenga una experiencia amplia en desarrollo de software web, sistematización de procesos empresariales, modelamiento de datos, metodologías de desarrollo,

análisis e ingeniería de requerimientos, así como programación de servidor y maquetación de cliente específicamente HTML y CSS. Para este primer punto, personalmente poseo toda la experiencia necesaria para llevar a buen término la realización de este aspecto técnico del proyecto.

De igual manera, el mini mercado EL DORADO deberá proveer los recursos de infraestructura como lo son un computador de escritorio o portátil que tenga una capacidad de procesamiento, espacio en disco y memoria aceptable para la instalación de los programas necesarios para la ejecución del aplicativo POS, así como contar con un lector de código de barras para realizar el registro de los productos en el sistema. Con respecto a este segundo aspecto, el mini mercado EL DORADO cuenta con varios equipos portátiles modernos, con capacidad de almacenamiento de estado sólido, así como varios lectores de código de barras para la implementación de este proyecto.

Es preciso anotar, que como cada tecnología tiene su curva de aprendizaje y para la elaboración de un proyecto de tal magnitud es preciso contar con varias competencias necesarias operando de manera coordinada, y para este caso en particular donde la totalidad de la elaboración del cronograma es realizada por el estudiante aspirante al título de ingeniero informático, se ha decidido realizar el desarrollo de este proyecto bajo las siguientes tecnologías:

Tabla 2

Tecnologías a usar cliente/servidor

Concepto	**Tecnología**	**Versión**	**Fabricante**
Framework código servidor	ASP.NET MVC	5	Microsoft
Lenguaje programación código servidor	C#	5	Microsoft

Framework código cliente	jQuery	3.x	Jquery.org
Sistema SGBD (sistema gestor de base de datos)	SQL Server Express	2016	Microsoft
Lenguaje programación código cliente	JavaScript	5	EcmaInc.
Lenguaje marcado cliente estilos	CSS	3	W3C
Lenguaje marcado cliente contenido	HTML	5	W3C

Al utilizar y conocer de antemano estas tecnologías acá mencionadas para la realización del proyecto garantizan la realización desde su parte técnica.

En cuanto a la reutilización de código preexistente es un aval, en cuanto a las tecnologías web, ya que se cuenta con un amplio margen de documentación, codificación e incluso frameworks de trabajo disponibles para lograr los objetivos, esto basado en las recomendaciones de viabilidad técnica, según la administración de proyectos de software. (Kenneth E & Julie E., 2011, pág. 63)

1.10 Viabilidad económica

La viabilidad económica es la segunda parte de la determinación de los recursos (Kenneth E & Julie E., 2011, pág. 63), el mini mercado EL DORADO es capaz de asumir el valor de la inversión que está considerando. Los recursos básicos que analizar son tiempo de ingeniería de requerimientos y tiempo de desarrollo. Para la realización de este proyecto se debe contar con un presupuesto aproximado a los $25.000.000 veinticinco millones de pesos. Esto contemplado todo el trabajo relacionado con el análisis, levantamiento de la información, ingeniería de requerimientos, modelamiento de datos y programación de software, lo que nos obedece a un

total de 983 horas de trabajo según el cronograma de trabajo detallado en el apartado de este trabajo.

Es claro que cada ámbito de trabajo mencionado tiene un costo más alto o más bajo dependiendo de la calidad y rapidez con que se necesiten los resultados de cada artefacto en cada momento del proyecto, pero basándonos en la experiencia laboral del mercado colombiano donde un ingeniero en promedio se gana $ 3.000.000 tres millones de pesos mensuales por (200) doscientas horas de trabajo en promedio mensual, se puede asumir que el costo de la hora ingeniería podría rondar los $25.000 veinticinco mil pesos, lo que nos da el valor inicialmente enunciado.

La viabilidad económica de este proyecto radica en el interés particular que el estudiante aspirante al grado tiene con el mini mercado EL DORADO, el cual está dispuesto a pagar en su totalidad por el trabajo realizado.

Para otros contextos, el costo promedio de una herramienta de software similar en el mercado colombiano ronda la suma de ($500.000) quinientos mil pesos, lo que hace que, para una inversión de tal magnitud, debe haber una venta mínima de 50 unidades del “sistema automatizado altamente usable para mini mercados”. Cifra que no es un hito difícil de alcanzar con un mercadeo apropiado y teniendo en cuenta el amplio rango de negocios en los que el sistema serviría.

1.11 Viabilidad de gestión

Es preciso contar con el aval del mini mercado EL DORADO para dar inicio al análisis y levantamiento de la información que serán la base para toda la planeación del proyecto, lo cual

es una necesidad principal del mini mercado para poder resolver sus necesidades inmediatas con el sistema POS.

Así mismo se debe contar con personal que este en capacidad de utilizar la herramienta informática, teniendo en cuenta la adaptabilidad que este tipo de implementaciones promueve, lo cual es un hecho en el mini mercado EL DORADO donde por más de 5 años se han utilizado diferente herramientas tipo POS. Teniendo en cuenta, que los usuarios han expresado la necesidad de un sistema que sea funcional y eficiente, las probabilidades de asegurar la viabilidad de la gestión son altas. Sin embargo, gran parte de la viabilidad de gestión recae en las interfaces de usuario elegidas (Kenneth E & Julie E., 2011).

1.12 Cronograma de actividades

1.12.1 Duración en horas

Tabla 3

Cronograma de actividades

ID	Title	Work Item Type	State	Effort	Value Area	Iteration Path	Tags	Duracion
1	Revision General del entorno y procesos del mini mercado EL Dorado	Product Backlog Item	Committed		Business	ProyectoUsta\Sprint 1		-
2	Entrevista con interesados por parte del mini mercado El Dorado	Task	In Progress			ProyectoUsta		4h
3	Visualización general del funcionamiento del negocio	Task	To Do			ProyectoUsta		8h
4	Chequeo de las herramientas de software actual	Product Backlog Item	Committed		Business	ProyectoUsta\Sprint 1		-
5	Verificar funcionalidad general y especifica de la herramienta de software actual para planear diseño de pantallas y funcionalidades	Task	To Do			ProyectoUsta		16h
6	Planificación general del sistema	Product Backlog Item	Committed		Business	ProyectoUsta\Sprint 1		-
7	Preview de la modularidad del sistema basada tanto en el analisis de la herramienta de software así como de la experiencia de creación de sistemas informaticos	Task	To Do			ProyectoUsta		24h
8	Levantamiento de información relacionada con el inventario	Product Backlog Item	Committed		Business	ProyectoUsta\Sprint 2		-
19	Analisis de el dominio del inventario en el sistema general	Task	To Do			ProyectoUsta		32h
20	Definir relación con los diferentes actores (vendedores, compradores, proveedores, sistema-ventas, sistema-inventario)	Task	To Do			ProyectoUsta		20h
10	Levantamiento de información con el flujo de llegada de pedidos, ingreso de items, ventas del producto.	Product Backlog Item	Committed		Business	ProyectoUsta\Sprint 2		-
22	Analisis del proceso general de entrada de items adquiridos al proveedor, su pago, su sistematización, su venta.	Task	To Do			ProyectoUsta		20h
23	Definir el flujo sistematico para la herramienta de software	Task	To Do			ProyectoUsta		20h
9	Levantamiento de información relacionada con los proveedores del mini mercado	Product Backlog Item	Committed		Business	ProyectoUsta\Sprint 3		-
21	Analisis del dominio en la aplicación de los proveedores, su relación respecto a la sistematización general del inventario, productos, ventas, pedidos	Task	To Do			ProyectoUsta		20h
11	Analisis de la viabilidad del proyecto	Product Backlog Item	Committed		Business	ProyectoUsta\Sprint 3		-
24	Análisis de la viabilidad técnica	Task	To Do			ProyectoUsta		10h
25	Análisis de la viabilidad económica	Task	To Do			ProyectoUsta		10h
26	Análisis de la viabilidad de gestión	Task	To Do			ProyectoUsta		10h
12	Realizar propuesta general y esperar validación por parte de interesados del mini mercado El Dorado	Product Backlog Item	Committed		Business	ProyectoUsta\Sprint 3		-
27	Entrega propuesta principal de trabajo y a espera de aval por parte del mini mercado El Dorado	Task	To Do			ProyectoUsta		10h
13	Ingenieria del proyecto Fase I	Product Backlog Item	New		Business	ProyectoUsta\Sprint 4		-
28	Modelo del dominio administrativo y base del sistema POS	Task	To Do			ProyectoUsta		15h
29	Modelo de casos de uso base administrativa sistema y base POS	Task	To Do			ProyectoUsta		24h
30	Modelo de análisis de secuencia del sistema base administrativa y base POS	Task	To Do			ProyectoUsta		24h

31	Diagrama de clases base administrativa sistema y base POS	Task	To Do			ProyectoUsta		20h
32	Diagrama de arquitectura base administrativa sistema y base POS	Task	To Do			ProyectoUsta		20h
33	Diagrama entidad relación base aministrativa sistema y base POS	Task	To Do			ProyectoUsta		20h
34	Diseño de interfaces gráficas base administrativa sistema y base POS	Task	To Do			ProyectoUsta		40h
14	Ingeniería del proyecto Fase II	Product Backlog Item	New		Business	ProyectoUsta\Sprint 5		-
35	Modelo de dominio módulos: Ventas, Inventario, Pedidos, Reportes	Task	To Do			ProyectoUsta		15h
36	Modelo de casos de uso módulos: Ventas, Inventario, Pedidos, Reportes	Task	To Do			ProyectoUsta		24h
37	Modelo de análisis de secuencia de módulos: Ventas, Inventario, Pedidos, Reportes	Task	To Do			ProyectoUsta		24h
38	Modelo de clases de dominio de módulos: Ventas, Inventario, Pedidos, Reportes	Task	To Do			ProyectoUsta		20h
39	Diagrama de arquitectura de módulos: Ventas, Inventario, Pedidos, Reportes	Task	To Do			ProyectoUsta		20h
40	Diagrama de entidad relación módulos: Ventas, Inventario, Pedidos, Reportes	Task	To Do			ProyectoUsta		20h
41	Diseño de interfaces gráficas de módulos: Ventas, Inventario, Pedidos, Reportes	Task	To Do			ProyectoUsta		40h
15	Desarrollo del proyecto Fase I	Product Backlog Item	New		Business	ProyectoUsta\Sprint 4		-
42	Desarrollo de capas de arquitectura de base administrativa y base POS	Task	To Do			ProyectoUsta		20h
43	Desarrollo de scripts relacionados con la implementacion de base de datos de base administrativa y base POS	Task	To Do			ProyectoUsta		40h
44	Desarrollo de capa de negocio de base administrativa y base POS	Task	To Do			ProyectoUsta		
45	Desarrollo de interfaces gráficas de base administrativa y base POS	Task	To Do			ProyectoUsta		80h
16	Desarrollo del proyecto Fase II	Product Backlog Item	New		Business	ProyectoUsta\Sprint 5		-
46	Desarrollo de capas de arquitectura de módulos: Ventas, Pedidos, Inventario, Reportes	Task	To Do			ProyectoUsta		20h
47	Desarrollo de scripts relacionados con la implementación de base de datos de módulos: Ventas, Pedidos, Inventario, Reportes	Task	To Do			ProyectoUsta		80h
48	Desarrollo de capa de negocios de módulos: Ventas, Pedidos, Inventario, Reportes	Task	To Do			ProyectoUsta		60h
49	Desarrollo de interfaces gráficas de módulos: Ventas, Pedidos, Inventario, Reportes	Task	To Do			ProyectoUsta		80h
17	Implementación y puesta en marcha	Product Backlog Item	New		Business	ProyectoUsta\Sprint 6		-
50	Despliegue de scripts en instancia de motor SGBD	Task	Done			ProyectoUsta		5h
51	Despliegue de aplicación web en servidor web de producción	Task	Done			ProyectoUsta		3h
52	Chequeo de funcionalidad del sistema	Task	In Progress			ProyectoUsta		5h

1.12.2 Duración en semanas

ID	Title	Duracion	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18
1	Revision General del entorno y procesos del mini mercado EL Dorado	-																		
2	Entrevista con interesados por parte del mini mercado El Dorado	4h																		
3	Visualización general del funcionamiento del negocio	8h																		
4	Chequeo de las herramientas de software actual	-																		
5	Verificar funcionalidad general y especifica de la herramienta de software actual para planear diseño de pantallas y funcionalidades	16h																		
6	Planificación general del sistema	-																		
7	Preview de la modularidad del sistema basada tanto en el analisis de la herramienta de software así como de la experiencia de creación de sistemas informaticos	24h																		
8	Levantamiento de información relacionada con el inventario	-																		
19	Analisis de el dominio del inventario en el sistema general	32h																		
20	Definir relación con los diferentes actores (vendedores, compradores, proveedores, sistema-ventas, sistema-inventario)	20h																		
10	Levantamiento de información con el flujo de llegada de pedidos, ingreso de items, ventas del producto.	-																		
22	Analisis del proceso general de entrada de items adquiridos al proveedor, su pago, su sistematización, su venta.	20h																		
23	Definir el flujo sistematico para la herramienta de software	20h																		
9	Levantamiento de información relacionada con los proveedores del mini mercado	-																		
21	Analisis del dominio en la aplicación de los proveedores, su relación respecto a la sistematización general del inventario, productos, ventas, pedidos	20h																		
11	Analisis de la viabilidad del proyecto	-																		
24	Análisis de la viabilidad técnica	10h																		
25	Análisis de la viabilidad económica	10h																		
26	Análisis de la viabilidad de gestión	10h																		
12	Realizar propuesta general y esperar validación por parte de interesados del mini mercado El Dorado	-																		
27	Entrega propuesta principal de trabajo y a espera de aval por parte del mini mercado El Dorado	10h																		
13	Ingenieria del proyecto Fase I	-																		
28	Modelo del dominio administrativo y base del sistema POS	15h																		
29	Modelo de casos de uso base administrativa sistema y base POS	24h																		
30	Modelo de análisis de secuencia del sistema base administrativa y base POS	24h																		

ID	Title	Duracion	S19	S20	S21	S22	S23	S24	S25	S26	S27	S28	S29	S30	S31	S32	S33	S34	S35	S36-S40	S41	S42-S46	S47-S50	S51-S55	S56
31	Diagrama de clases base administrativa sistema y base POS	20h																							
32	Diagrama de arquitectura base administrativa sistema y base POS	20h																							
33	Diagrama entidad relación base aministrativa sistema y base POS	20h																							
34	Diseño de interfaces gráficas base administrativa sistema y base POS	40h																							
14	Ingeniería del proyecto Fase II	-																							
35	Modelo de dominio módulos: Ventas, Inventario, Pedidos, Reportes	15h																							
36	Modelo de casos de uso módulos: Ventas, Inventario, Pedidos, Reportes	24h																							
37	Modelo de análisis de secuencia de módulos: Ventas, Inventario, Pedidos, Reportes	24h																							
38	Modelo de clases de dominio de módulos: Ventas, Inventario, Pedidos, Reportes	20h																							
39	Diagrama de arquitectura de módulos: Ventas, Inventario, Pedidos, Reportes	20h																							
40	Diagrama de entidad relación módulos: Ventas, Inventario, Pedidos, Reportes	20h																							
41	Diseño de interfaces gráficas de módulos: Ventas, Inventario, Pedidos, Reportes	40h																							
15	Desarrollo del proyecto Fase I	-																							
42	Desarrollo de capas de arquitectura de base administrativa y base POS	20h																							
43	Desarrollo de scripts relacionados con la implementacion de base de datos de base administrativa y base POS	40h																							
44	Desarrollo de capa de negocio de base administrativa y base POS																								
45	Desarrollo de interfaces gráficas de base administrativa y base POS	80h																							
16	Desarrollo del proyecto Fase II	-																							
46	Desarrollo de capas de arquitectura de módulos: Ventas, Pedidos, Inventario, Reportes	20h																							
47	Desarrollo de scripts relacionados con la implementación de base de datos de módulos: Ventas, Pedidos, Inventario, Reportes	80h																							
48	Desarrollo de capa de negocios de módulos: Ventas, Pedidos, Inventario, Reportes	60h																							
49	Desarrollo de interfaces gráficas de módulos: Ventas, Pedidos, Inventario, Reportes	80h																							
17	Implementación y puesta en marcha	-																							
50	Despliegue de scripts en instancia de motor SGBD	5h																							
51	Despliegue de aplicación web en servidor web de producción	3h																							
52	Chequeo de funcionalidad del sistema	5h																							

2- Marco de referencia

2.1 Antecedentes

El mini mercado EL DORADO ha utilizado tres diferentes sistemas tipo POS, que han sido adquiridos bajo la modalidad de alquiler anual y compra de licencia perpetua. Estos sistemas han sido Solvermedia ®, Software POS ® y tiendaTek ®. Se eligieron estas alternativas porque en el momento de su adquisición cada una ofrecía ventajas con respecto a su antecesor. La primera solución que se implementó fue SolverMedia ®, como producto de una recomendación personal. Este sistema prestaba los servicios de registro de ventas, inventario y reportes; pero dificultaba el ingreso de productos que carecían de codificación, ya que lo hacía por medio de un módulo adicional que imprimía códigos de barras alternativos para luego asignarlos, generando un proceso tedioso, ya que la variedad de estos productos es bastante amplia. Por esta razón en particular se decidió utilizar la herramienta Software POS ® desarrollada en Colombia, ya que esta solución además de contar con una funcionalidad más amigable para los productos no codificados permitía tener un conjunto de funcionalidades que mejoraban esta labor. En esta primera migración se evidencio la falta de un sistema abierto para la explotación de sus datos, ya que este inconveniente hizo que fuera un proceso manual y demorado. La herramienta Software POS ® tuvo un desempeño satisfactorio en su característica mejorada que ofrecía, pero aun así carecía de funcionalidades importantes como las ventas paralelas, abierto a la explotación de sus datos entre otros inconvenientes que entorpecían el proceso de venta. Al manifestar esta serie de inconvenientes con un proveedor importante del mini mercado EL DORADO, este recomendó la herramienta tiendaTek ® que estaba siendo objeto de pruebas, por lo que se permitían poder entregar la herramienta instalada, junto con todo el hardware involucrado (Tablet y lector de código de barras bluetooth) durante 6 meses en el mini mercado, y sin costo alguno.

La herramienta fue revolucionaria en cuanto a la facilidad de registro de ventas y libertad del lector de código de barras; sin embargo, lo que en algunos aspectos facilitaba esta herramienta, en otros aspectos adolecía de lo que las otras herramientas si proporcionaban. Con toda esta seria de experiencias y demás objetivos informáticos evidencias, se contempla para el mini mercado EL DORADO la adquisición de un sistema tipo POS a las medidas de sus necesidades.

Este tipo de sistemas denominados POS, han sido desarrollados desde la invención de las máquinas de cajeros o ECR (electronic cash register => registradoras de dinero electrónicas) mundialmente conocidas por firmas de tecnología como la NCR quien fue uno de los principales fabricantes en su tiempo de este tipo de herramientas y actualmente líder en la manufactura de ATM (Automatic transaction machine => máquina de transacciones automáticas o conocidos como cajeros electrónicos). (contributors, 2017)

Para el año 1986 y con la evolución de la computación gráfica programable GENE MOSHER introdujo el primer sistema POS grafico conocido. (contributors, 2017)

Para el año 1992 Martin Godwin y Bob Henry desarrollaron la solución de retail mundialmente conocida como IT-Retail que corría en un sistema Windows. (contributors, 2017)

Desde entonces existe una oferta extensa de soluciones POS desarrolladas para correr en diferentes plataformas y con diferentes utilidades, conectividad a diferentes estándares creados por la industria del hardware como buses de datos, impresoras, accesibilidad remota entre otros.

Los sistemas POS son catalogados como sistemas altamente complejos debido a la parametrización especifica que se requiere para cada tipo de negocio, pero en general lo que se demanda en este tipo de sistemas, es rapidez en la ejecución, conectividad, estabilidad y con un

avanzado conjunto de interfaces graficas que permita hacer más con menos (contributors, 2017).

Como se puede observar en este análisis previo de investigación del estado del arte en cuanto a sistemas de este tipo se aprecia una serie de características que son objeto de este proyecto.

2.2 Marco teórico

2.2.1 Usabilidad

Para Microsoft, el término "usabilidad" en el contexto de la creación de software representa un enfoque que ubica al usuario en vez del sistema como el centro del proceso. Existen varias características que caracterizan a un sistema altamente usable como lo son: la utilidad de la funcionalidad, la facilidad con la que un usuario puede aprender a utilizarla, la eficiencia con la que una tarea a través de una interfaz gráfica es realizada, la facilidad con que un usuario retiene una funcionalidad, que tan posible es que debido a la funcionalidad se incurran en errores de usuario y por último la satisfacción de utilización del usuario. "Microsoft Corporation. Usability in software design. United States. https://msdn.microsoft.com/en-us/library/ms997577.aspx"

La usabilidad mencionada en nuestro objetivo del trabajo principal está basada en la premisa de que un software o programa informático sea sencillo de utilizar, intuitivo, facilita el alcance de usuarios no aptos por ciertas discapacidades, garantizando accesibilidad y operatividad sencilla e intuitiva y demás atributos que hacen una interfaz de usuario junto con una necesidad de negocio encontrar un equilibrio perfecto.

Existe un tipo de aplicaciones que han tenido un gran auge en los últimos años debido a la masiva utilización de software web que caracteriza a las herramientas informáticas como aplicaciones

SPA (Single page application- Aplicaciones de única página), que no son más que una interfaz gráfica de usuario, altamente rica en contenido y usabilidad que le permiten al usuario operar el software de una manera más ágil y rápida.

2.2.2 Automatización

La automatización es el proceso mediante el cual haciendo uso de programas informáticos se pueden realizar tareas de forma parcial o completa, con o sin la intervención de un usuario. Con la automatización se logra aumentar la producción y disminuir la incertidumbre producto del error por factores humanos, así mismo la automatización permite realizar procesos en un tiempo considerablemente menor a un proceso manual y poder escalarlo a la medida de las necesidades. (Grunbacher, 2004).

La automatización de software es un proceso hecho a medida de la necesidad. Existen herramientas que permiten elaborar esquemas basados en asistentes gráficos para poder explotar o realizar tareas de manera repetitiva con poca o nula intervención de un usuario, pero en general es un proceso hecho a la medida de la necesidad puntual del proceso que se desea implementar apoyándose en el uso de algoritmos, programas o demás instrumentos que procesen entradas, realicen un proceso en ellas y finalicen con un resultado de ese proceso.

La automatización en el contexto de nuestro trabajo pretende realizar un análisis histórico del registro de ventas del sistema para poder crear los pedidos por proveedor de manera automática generando poca o nula intervención de análisis por parte del usuario administrador del sistema.

2.2.3 Conectividad

La conectividad es el estado o cualidad de ser conectado o conectivo. En las ciencias de la computación es la habilidad de comunicar y relacionar entre sistemas, dispositivos electrónicos, programas informáticos e incluso una red como la internet. (LLC, 2017).

El mundo informático globalizado ha ejercido gran impacto en lo que a conectividad en las tecnologías de la información se refiere. El uso masivo de internet, dispositivos móviles, gadgets personales y en general un amplio margen de elementos que hacen uso de la información de forma centralizada y descentralizada han promovido el uso de sistemas distribuidos, aplicaciones basadas en servicios, con repositorios o funcionalidades de acceso de datos consumibles, entre otras características de conectividad.

La conectividad se ha convertido en una característica de los sistemas informáticos con una relevancia bastantemente alta a la hora de la evaluación de las características de los sistemas informáticos, tanto que es crucial para tecnologías específicas como lo es la movilidad y sistemas distribuidos. “Ingeniería de software en el desarrollo de aplicaciones para dispositivos móviles. Argentina. (Thomas, y otros, 2013).

En el contexto del desarrollo de este trabajo se realizará un API (application program interface / interface de aplicación de programa) de consumo de información por medios informáticos programables para terceros en caso de que se requiera, así como una base de datos relacional de la información, accesible mediante la seguridad necesaria del caso con el ánimo de poder explotar este activo dentro del negocio o facilitar su migración.

2.2.4 Arquitectura de software

La arquitectura de software es el proceso de definir una solución estructurada que reúna todas las

necesidades técnicas y funcionales de un sistema, optimizando la calidad en atributos como el desempeño, la seguridad, escalabilidad, facilidad de mantenimiento y facilidad en la administración (Corporation, 2009).

La arquitectura de software no se puede definir sin incluir las inherentes metas que persigue y que son el resultado de la retroalimentación de ejecuciones de proyectos pasados, estandarización de soluciones que competen una tarea en particular y el objetivo de crear software altamente escalable, seguro, fácil de mantener y que sea altamente estructurado.

Técnicamente la arquitectura de software busca crear una brecha entre las responsabilidades intrínsecas de un sistema de software, por ejemplo, separando temas de índole técnica con los relacionados con el núcleo del negocio o dominio de la aplicación funcional.

Al realizar este tipo de separaciones se logran componentes de software con responsabilidades específicas asegurando el mantenimiento, escalabilidad, independencia entre otras bondades de cada componente de software.

Siempre existirá una relación implícita entre los requerimientos técnicos de la solución informática con su dominio o requerimiento funcional, por lo que la flexibilidad en estos ámbitos debe ser investigado de forma exhaustiva a la hora de la realización de la implementación de una arquitectura a un sistema.

Existe una premisa muy importante en la programación de software que no hay nada más fijo que el cambio, por ende, la aplicación de una arquitectura de software ideal pretende hacer flexible al cambio a los componentes de software.

Las buenas prácticas del desarrollo comprenden:

- Nombramiento de objetos codificables como clases, métodos, campos, parámetros, carpetas, recursos, procedimientos almacenados, tablas, y demás elementos de manera coherente y organizada para que sea deducible su ubicación, objetivo del mismo entre otros factores que a la larga son los que van a permitir un mantenimiento eficaz y modificación de la solución bajo parámetros establecidos y de una manera más sencilla.
- Ajustarse a un estándar de codificación dependiendo el lenguaje de programación a utilizar y despersonalizar el código adoptando buenas prácticas de programación (Pantaleo & Rinaudo, 2015, pág. 251).
- Creación de código que permita realizar más de una operación por transacción nos va permitir ser escalables en las soluciones de software ya que es importante prever que la concurrencia del sistema puede llegar a ser mayor de la esperada y la reescritura de código para manejar procesos en bloque debe desde un principio se codificada de tal manera incluso si en primera instancia se realizara solo una operación por transacción.
- Los lenguajes de programación de alto nivel permiten aplicar técnicas de desarrollo para poder reutilizar funcionalidades de manera óptima, como también de comprender de manera más fácil el código escrito anteriormente para poder atender un cambio a futuro, es proceso que los patrones de diseño son una alternativa bastante probada para solucionar problemas específicos de software.
- Aplicar pruebas unitarias automatizadas al código ha demostrado ser un filtro para código con falencias o aseguramiento de la calidad en el producto de software. Hoy en día entornos de desarrollo integrados o herramientas de software permiten hacer esto de manera más automatizada y clara con el ánimo de asegurar la calidad en el desarrollo final.

Como punto final se refiere al tema de la arquitectura de software, cabe resaltar que el enfoque

DDD (Domain Drive Design / Diseño basado en el dominio) que es una estructura de prácticas muy relacionadas a la arquitectura de software, centraliza al modelamiento de la arquitectura en el núcleo negocio, por lo que enfoca al uso del lenguaje ubicuo o paralelo en todos los procesos relacionados con la creación del software, así que de forma homogénea todo sea llamado igual por todas las partes interesadas de la arquitectura.

La arquitectura de software pretende promover el uso de separación de responsabilidades, mantenibilidad, escalabilidad entre otros beneficios para el desarrollo e implementación de un sistema informático. “Se entiende como arquitectura de un sistema las partes que lo componen y la forma en que estas reaccionan” (Pantaleo & Rinaudo, 2015, pág. 200).

Según Eric Evans, el aspecto más complicado de una aplicación de software que tenga una complejidad alta y una envergadura amplia no es la implementación, que a la final es un proceso que en un punto se puede volver mecánico o incluso automatizado, sino en cambio el verdadero problema radica en el domino de la aplicación en su contexto del mundo real. El dominio se define como la estructuración de todos las entidades, componentes, transversales, servicios y demás elementos claves del negocio en su representación en el sistema, y que deben o deberían ser tenidos en cuenta a la hora del diseño de la funcionalidad en sí. Metodologías de diseño arquitectónico de software como DDD enfocan al equipo de diseño y desarrollo a aproximarse a dominios altamente complejos utilizando técnicas de codificación y segmentación de código que enfocan bastante esfuerzo en la comprensión del dominio de la aplicación, y aplicando principios de desarrollo estructurados realizar un sistema más fácil de mantener, obteniendo un nivel de abstracción muy similar al dominio del negocio de la aplicación.

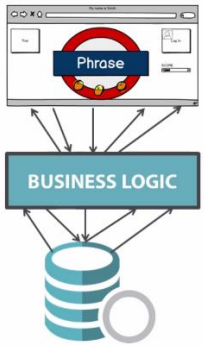
2.2.4.1 Persistencia

Una sólida arquitectura de software puede reducir el costo del desarrollo en el largo plazo. La elección de una metodología de acceso a datos está muy arraigada a las necesidades de la solución o proyecto informático, y es imprescindible ya que en general todos los sistemas informáticos persisten sus operaciones o datos del negocio, funcionalidad entre otros, en un sistema céntrico de base de datos o similar. Existen diferentes metodologías en particular para el acceso a datos, unas son prácticas para utilizar sacrificando dependencias, como otras tienen un componente de performance ideal, sacrificando practicidad a la hora de la implementación. La elección de un ORM (object relational mapper/objeto de mapeo relacional) es indispensable para poder tener un alcance medible para este proyecto en particular, ya que el acceso a datos representa un porcentaje importante de la codificación total del proyecto. Adicional a esto, un sistema altamente escalable, es posible con el uso indicado de este tipo de herramientas, ya que son altamente configurables y ampliables a las necesidades técnicas del proyecto.

2.2.4.2 User experience driven design (Diseño basado en la experiencia del usuario)

Como otro supuesto inherente a la producción de un producto de software y en especial los de tipo web, o que tienen una directa implementación para ser utilizado por usuarios, la interfaz gráfica juega un papel muy importante en el ciclo de la vida del desarrollo de la funcionalidad.

Todos los sistemas terminan con una segmentación como esta:



Donde una persistencia es paralelamente operada por una capa de negocio encargada de realizar los llamados, ingresos y procesos de datos que esta capa le envía y proyectarlos, alinearlos y controlarlos de su capa contraparte conocida como la presentación del sistema.

Imagen 55. Separación entre capas de negocio.

La arquitectura basada en el diseño o la presentación deberá planearse para brindar la mejor experiencia de usuario y la escalabilidad y mantenibilidad que esta capa promueve, debido a que igual debe existir un desacople entre las otras capas para poder realizar un cambio en la presentación de tal forma que pueda ser adaptable a diferentes tipos de presentación si es el caso.

2.2.4.3 Capas arquitectónicas

Con el ánimo de segmentar ordenadamente cada una de las responsabilidades de los diferentes servicios o funcionalidades del sistema se pretende desarrollar el software en capas arquitectónicas que definan un alcance para cada segmento de la aplicación. Un ejemplo muy claro es cuando un elemento de interfaz gráfica muestra información almacenada en un repositorio de datos. Aunque están relacionados los dos elementos, cada uno hace parte de una capa arquitectónica diferente e incluso puede haber otras capas arquitectónicas entre ellos. Por tal razón el elemento gráfico, deberá responsabilizarse por mostrar la información y demás funcionalidades de interfaz gráfica y la capa arquitectónica, responsabilizarse por el acceso a los datos. Código que no defina claramente estas responsabilidades, será muy difícil de separar, mantener, reutilizar, desplegar entre otros inconvenientes; es por eso por lo que en cuanto a arquitectura de software es preciso separar el código en capas y permitir una comunicación entre

ellas con el ánimo de asegurar mantenibilidad, escalabilidad y todas las premisas que la arquitectura de software abona.

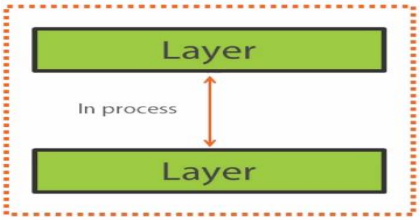


Imagen 59. Separación entre capas de la aplicación.

Realizando el código con esta premisa es más fácil ubicar donde se encuentra un segmento de código que se necesita expandir, mejorar.

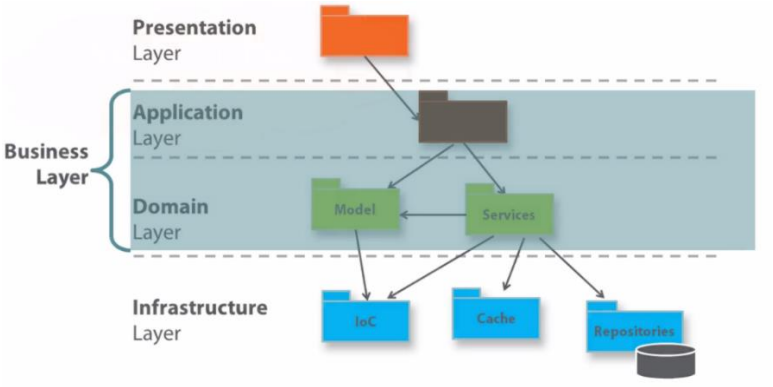


Imagen 60. Separación entre capas de la aplicación. Copyright www.pluralsight.com

El diseño de una aplicación desde cero permite aplicar las mejores prácticas y elección de metodologías que faciliten las premisas inicialmente mencionadas, como es el caso de este proyecto en particular.

2.2.5 Metodologías de desarrollo

En la ingeniera de software, un proceso de desarrollo de software consiste en varias etapas y fase de trabajo, como lo son el diseño, administración de producto y gestión del proyecto. Es también

conocido como el desarrollo de ciclo de vida del software, donde puede incluir desde la pre definición de artefactos entregables, contemplando actores con diferentes roles y su intervención en el ciclo de vida de desarrollo del software.

El desarrollo de software tiene bastantes focos de trabajo de las cuales un gerente de proyecto, programadores, usuarios y demás involucrados deben tener en cuenta ya que es un proceso bastante complejo y de bastante tiempo y análisis por parte de todos los involucrados. Teniendo clara la necesidad a suplir y anticipándose al cambio que es natural en los procesos empresariales y demás, es necesario aplicar las mejores prácticas a la ejecución del proyecto para poder llevar a el mejor término el producto final.

Las metodologías de desarrollo proporcionan una interactividad y mejores tiempos de entrega en el ciclo del desarrollo del software, proporcionando un modelo de pasos a seguir para obtener unos hitos en el proceso de la construcción de software. Existen varias metodologías agiles como lo son XP (extreme programming), Scrum, Crystal, Lean Development entre otras y aunque cada una de estas tiene un enfoque específico, todas ellas comparten una visión y objetivos generales basados en el manifiesto ágil que no es más que un documento de referencia, el cual está dado por la experiencia fallida en la utilización de metodologías conducidas por los planes (…) (Pantaleo & Rinaudo, 2015, pág. 101), el cual promueve una iteración continua en cada una de las fases involucradas como es el planeamiento, testing (pruebas), integración (integración continua) y otras formas de continua evolución del proceso del desarrollo del software buscando que todos los involucrados del sistema participen y puedan tomar decisiones en paralelo, rápidamente y efectivamente.

Diferentes equipos de software pueden adaptarse mejor a una metodología en particular y algunos

la adoptaran completamente o parcialmente, incluso hay casos donde no se implementa al pie de la letra cada estrategia proporcionada en el documento, sino que el equipo se amolda al contexto particular.

Un proyecto de software con un alcance amplio en particular y sin una metodología de desarrollo es un proyecto con una incertidumbre bastante amplia de lograrse (David Starr – Pluralsight – Scrum fundamentals), puede haber demasiado positivismo a la hora de iniciar el proyecto, pero en el camino puede determinarse incluso que el proyecto no es funcional y que deberá replantearse nuevamente.

Varios tipos de proyectos han sido no exitosos a través de la historia debido a la falta de una metodología específica que asegure o minimice el riesgo de un proyecto no exitoso.

Jeff Sutherland y Ken Schwaber fueron quienes desarrollaron esta metodología a mediados de los años 90 para un proyecto en particular estadounidense del departamento FBI denominado Sentinela y que alrededor de 10 años había sido objeto de fracaso en cuanto a su implementación debido a que el resultado era que no se tenía el software en el tiempo acordado y costaba mucho más de lo inicialmente planeado.

Después de gastarse alrededor de 640 millones de dólares y casi 10 años en los que el desarrollo de este software nunca se terminó, el gobierno estadounidense encamino el proyecto Sentinela, de la mano de Jeff y ken como CIO (Chief Information Officer/Líder de la gestión estratégica) y CTO (Chief technical officer/director de tecnología) del proyecto respectivamente, culminando el proyecto en un periodo de 2 años y costando alrededor de 30 millones de dólares para su producción. Actualmente es la base del sistema CRM de delitos del FBI a la fecha.

El secreto, la metodología que como creadores ellos mismos denominaron scrum la cual se

define como una herramienta de trabajo, mas no como una solución definitiva, ya que scrum no dice cómo resolver los problemas sino como incrementalmente resolver un proyecto de software.

Así mismo se puede ver según una encuesta realizada a 1208 profesionales del sector IT involucrado con el desarrollo ágiles dan un aval por scrum como la metodología a elección principal para los proyectos.

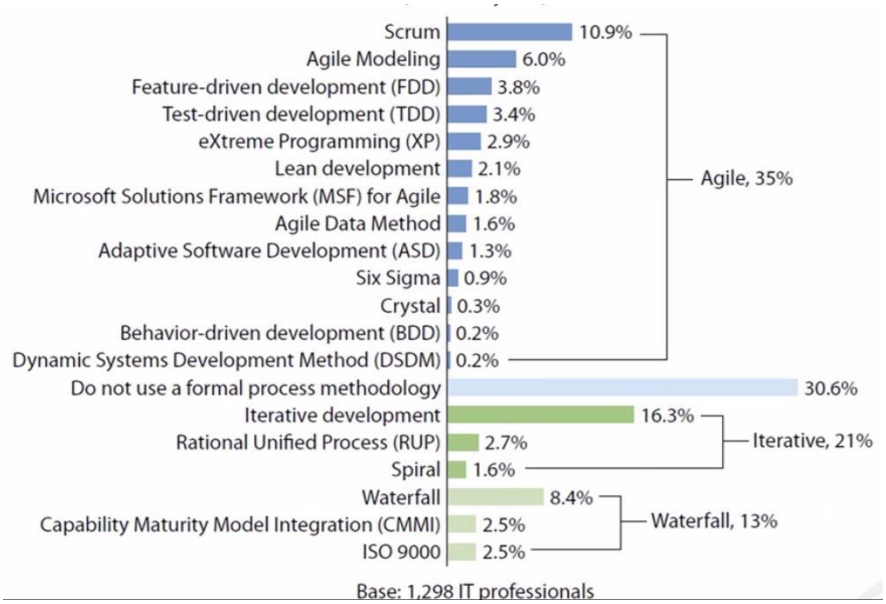


Imagen 1. Elección de metodologías basada en encuesta al sector IT. Copyright 2010 Forrester/Dr. Global Developer Techno graphics.

La metodología de desarrollo scrum debe su nombre a, “la formación de rugby donde los jugadores miembros de un equipo se agrupan y suman sus fuerzas empujando el obstáculo que en ese caso es el equipo adversario” (Pantaleo & Rinaudo, 2015, pág. 98) y que fomenta las siguientes premisas:

- Entregables incrementales que se ajustan a el desarrollo por fases de la herramienta.
- Características definidas a nivel de desarrollo complementadas con sus respectivos

product backlog

- Enfocar al equipo de trabajo, cliente, owner (dueño del producto), desarrollo y scrum master, en objetivos específico-entregables incrementales de forma tal que se pueda llevar a cabo la solución total del sistema de forma continua, entregando o produciendo bases significativas que retroalimentan el proyecto en general, generando confianza y compromiso entre las partes involucradas.

La metodología se compone de los siguientes artefactos, roles y eventos.

2.2.5.1 Product backlog

El producto backlog, es el listado priorizado de requerimientos del sistema a construir (Pantaleo & Rinaudo, 2015, pág. 98). Como product backlog se tendrá 14 hitos comprendidos entre 23 tareas. Se compone de los features (carateristicas) puntuales del sistema, que son un referente o agrupador para las actividades que lo conforman.

2.2.5.2 Scrum master

Es la persona encargada de promover la metodología de scrum en los integrantes del equipo. Más que velar por el cumplimiento de los Product backlog, features y task es el gestor encargado de correlacionar al equipo de trabajo con el ánimo de promover una buena comunicación basada en la metodología.

2.2.4.3 Product manager

Es una persona o rol, que de forma particular gestiona a un nivel gerencial el progreso de los Product backlog ítem y en particular recomienda de manera general la ejecución e incremento de

la producción orquestado con el cliente.

El product manager es una persona no un conjunto de personas, ya que es un rol de supervisión.

2.2.5.4 Development team (Equipo de desarrollo)

El equipo de desarrollo son todas las personas en capacidad de codificar cada una de las tareas para cumplir con la realización de los product backlog, altamente orientados al trabajo en equipo y con reuniones prácticamente diarias en las que la claridad, la honestidad y la colaboración son la piedra angular para el incremento en cada uno de los entregables propuestos.

Realmente es el development team, quien decide la ejecución y estimación de las tareas debido a su experticia teniendo en cuenta los requerimientos del cliente. Según el manifiesto ágil debe ser un grupo de entre 4 y 10 personas.

2.2.5.5 Evento scrum sprint backlog

Es el plan bien detallado de los entregables a nivel general de la metodología para un intervalo de tiempo definido.

La recomendación para cada sprint es de 1 mes y no más de 2 meses para poder sesgar de manera controlada cada uno de los features o entregables y así poder retroalimentar de manera organizada. Para un sprint pueden seleccionarse uno o más product backlogs que ellos a su vez tendrán una o varias tareas asociadas.

De parte del team development se tendrá un objetivo u objetivo a cumplir de manera muy coordinada por cada uno de los participantes y de parte del scrum master y el product owner se

tendrá una perspectiva de trabajo restante.

Pero en resumen es un recurso inherente al equipo de desarrollo.

2.2.5.6 Evento scrum sprint

El sprint es un tiempo aproximado a los 30 días y nunca superado por 60 días donde se debe ejecutar una planeación previa realizada en un capacity de equipo.

De su analogía de competencias atléticas, el sprint hace referencia a un ciclo rápido donde se pretende tener una retroalimentación rápida para minimizar los riesgos.

2.2.5.7 Evento scrum sprint planning

Hace referencia a la reunión principal entre los interesados y con base a una planeación de requerimientos funcionales o casos de uso realizar una planeación para el sprint.

Durante el sprint planning es que se llega a obtener un sprint backlog y es donde se tiene certeza de que se va a hacer y como para las metas del sprint.

2.2.5.8 Evento scrum daily scrum

Serie de reuniones por y para el equipo de desarrollo donde se crea un plan para el siguiente día y una serie de retroalimentaciones necesarias para poder dar prioridades en las tareas y product backlogs a obtener en el tiempo. El scrum master es un facilitador para de este tipo de encuentros, pero es tarea del equipo de desarrollo tener esta serie de interlocuciones para dar a conocer cada uno de los inconvenientes y generar un trabajo en equipo orquestado.

2.3 Marco conceptual

Basados en la información y documentación relacionada soportada en las bases teóricas expuestas en el marco de trabajo anterior, se delimitarán conceptos relacionados con la implementación específica para este proyecto.

2.3.1 Sistemas POS

También conocidos como sistemas TPV (Terminales de punto de venta) son programas de cómputo que corren en una computadora personal o de escritorio y que permiten tener funcionalidades específicas para negocios comerciales de diferentes tamaños tales como: manejo de inventario, registro de ventas, ingreso de mercancías, registro de empleados, reportes de ventas, registro de ubicaciones, registro de proveedores, registro de clientes, entre otras funcionalidades. Son herramientas bastante útiles para llevar ordenada y digitalmente toda la información de un establecimiento comercial, o actividad económica, incluido entre ellos los mini mercados.

2.3.2 Rapidez en la operación del sistema

Asociado a la usabilidad, definida en el marco teórico, comprende la carga eficiente del contenido ofrecido por un sistema informático, facilidad y rapidez de la operación de las pantallas ofrecidas, así como un inicio de sesión eficiente. Igualmente está relacionado con el modo en que se desarrolle una funcionalidad específica, en la que sea necesario procesar grandes volúmenes de información, lo que puede llevar a tener que utilizar procesos asíncronos para garantizar la rapidez.

2.3.3 Simplicidad

Asociado a la usabilidad, definida en el marco teórico, la elaboración de cada una de las

funcionalidades, así como de la interfaz gráfica de usuario relacionada se orientará a brindar la menor saturación de operación para garantizar los objetivos buscados en la usabilidad como la retención del proceso por parte del usuario entre otros beneficios expuestos en el marco teórico con respecto a la usabilidad.

Sistemas bastante complejos en cuanto a su uso o su interfaz gráfica saturada, promueven a desalentar al usuario en el uso de la herramienta o simplemente termina siendo un obstáculo en la operación, en vez de serle de utilidad.

2.3.4 Algoritmos informáticos

Los sistemas informáticos están compuestos en su gran mayoría por componentes compilados que se interrelacionan con otros elementos programados compilados o del sistema, para la manipulación, entrada, procesamiento y salida de datos. Estos componentes en su gran mayoría son escritos utilizando un lenguaje y semántica estructurada que permite realizar ciertas acciones sobre el flujo de información. Los programas informáticos son eso, instrucciones que se ejecutan una a una, en la mayoría de los casos en forma secuencial, en otros de forma paralela, enfocados a procesar información y entregar resultados para otra fase del proceso o como punto de finalización del flujo de información.

Un proceso sistemático con estas características, que se ejecuta una y otra vez para cada petición o evento que llame a la ejecución de este proceso, podría pensarse como un sistema automatizado, que realiza acciones basadas en unas reglas, procesa información basado en unas reglas, y presenta información basado también en unas reglas o criterios previamente definidos.

Este apartado hace referencia a la automatización, previamente referenciada en el marco teórico.

En el contexto de este proyecto, utilizando esta herramienta de un programa informático, se pretenderá entregar los pedidos a realizar de manera automatizada por proveedor, basado en una serie de algoritmos que analizan la cantidad registrada de ventas comparado con la cantidad en stock de uno o más productos asociados al proveedor, generando un resultado automático listo para ser procesado o con la asistencia del usuario poder modificar esa sugerencia automatizada.

2.3.5 Requerimientos funcionales y no funcionales

Los requerimientos no funcionales son todos esos componentes o funcionalidades implícitas que permitan garantizar el aseguramiento de la información pero que no contemplan una necesidad propia del negocio. Pueden diferenciarse tipos de requerimientos no funcionales como las validaciones en las interfaces gráficas de entrada de datos de intervención del usuario, con el ánimo de asegurar el diligenciamiento correcto de la información, así mismo como requisitos no funcionales se pueden clasificar a ese tipo de requerimientos que proporcionen la alta disponibilidad, correcto funcionamiento y escalabilidad de la herramienta.

Por el contrario, los requisitos funcionales, son los requerimientos que pretenden dar solución a las necesidades sistemáticas que han emergido y son la base del proyecto de software. En estos se ubican todos los procesos críticos de la operación, así como los que se quieren crear o desarrollar y que son el núcleo del negocio. De cara al usuario son los principales procesos que espera poder obtener de la herramienta informática.

Para este proyecto en particular son las necesidades puntuales en cuanto a usabilidad, automatización y conectividad del sistema las que representan el eje principal de la solución informática.

2.3.6 Independización de responsabilidades

Es un concepto relacionado con la arquitectura de software, donde se pretende diseñar y programar los componentes de la aplicación de forma que sean autónomos con el ánimo de poder independizar funcionalidades, reutilizar código, poder realizar pruebas de forma independiente entre otros beneficios. En este proyecto se realizará modelamiento por capas orientadas a escribir funcionalidades específicas de forma agrupada para generar una estructura que sea fácil de mantener y sea altamente escalable en términos de ordenamiento del proyecto.

Así mismo al independizar comportamientos y responsabilidades de cada proceso permitirá garantizar la escalabilidad y aseguramiento de la calidad en el momento de generar pruebas al código escrito.

2.3.7 Levantamiento de la información

Relacionado con el elemento teórico de metodologías de desarrollo expuesto en el marco correspondiente del proyecto, es el proceso en sitio mediante el cual se analiza el estado del proceso desde una perspectiva crítica y analítica con aras de recopilar la mayor información posible y relevante para la definición de la funcionalidad o etapa de un proceso.

Se parte de definir componentes de entrada, actores, variables del entorno, constantes, procesamiento de datos y resultados que proveen una perspectiva del contexto.

El principal método utilizado para este proyecto será la observación del proceso y la entrevista informal para poder obtener el punto de vista del proceso del usuario final y actores involucrados y así utilizar esta información para la documentación del requisito.

2.3.8 Requisitos de software

Es preciso basar toda esta funcionalidad en el problema específico de la ingeniería de requerimientos desarrollada, ya que es sumamente importante junto con el conocimiento técnico de infraestructura, codificación, análisis y desarrollo, operar todas estas variables en conjunto para poder resolver de la manera más pertinente la necesidad de software. Es por ello por lo que aun contemplando una arquitectura de software con un alcance amplio, algoritmos eficientes y experiencias de usuario optimas, el objetivo principal está orientado a suplir las necesidades funcionales del negocio basada en esa ingeniería de requerimientos

2.3.9 Compatibilidad

Condición que caracteriza a un sistema informático, sistema de hardware o demás componentes que cumplen una función específica poder compartir, recibir, entablar una comunicación con un componente de igual o similares características mediante un medio por donde se comparte esa información. Hace referencia al apartado de conectividad del marco teórico del proyecto, ya que la compatibilidad es el medio o garantía que se tiene al poder tener un sistema conectable para la explotación de los datos. La compatibilidad se da en muchos varios aspectos, tanto físicos, como por ejemplo lo es el hardware, así como lógicos o de software como son tecnologías de acceso a bases de datos. Diferentes fabricantes se encaminan a generar sus propios sistemas de comunicación, pero de la mano de los consorcios, se logra tener un camino de compatibilidad entre las partes a interactuar.

Es una cualidad muy extendida en la informática en diferentes contextos gracias en parte a la estandarización de protocolos de trabajo y el alto uso de programas informáticos.

2.4 Marco metodológico

El desarrollo de la investigación del presente trabajo será exploratorio en campo, utilizando observación directa del proceso, entrevistas a los usuarios del sistema, indagando por las necesidades planteadas en la formulación del problema, teniendo como base los objetivos a desarrollar y soportando en toda la información documental y de experiencia, relacionada con los marcos de trabajo del proyecto.

La metodología a utilizar partirá de obtener la mayor información relacionada con los objetivos de este trabajo, contemplando la revisión de las herramientas previas, que no cumplieron con los objetivos principales del mini mercado EL DORADO, obteniendo una visión particular y general de la problemática expuesta y corroborar las necesidades del usuario. Ir reuniendo todos los documentos que definan la problemática y sirvan de base para generar la ingeniería de requerimientos pertinente para dar paso a la elaboración de una arquitectura de software y posterior codificación de la solución.

En la tabla 4, se detallarán cada una de las estrategias a realizar para conseguir el objetivo específico mencionado y la variable a obtener a la cual hace referencia.

Tabla 4

Especificación de artefactos obtenibles en el proceso del proyecto

Actividad Especifica	Estrategia	Objetivo Del proyecto
Investigación inicial	Se realizará una investigación de campo que contemplará entrevistas con los interesados, así como de un análisis exhaustivo de todos los procesos del mini mercado EL DORADO.	- Usabilidad - Conectividad - Automatización
Comprobación de herramientas POS adquiridas por el mini	Se analizarán cada una de las herramientas informáticas adquiridas por el establecimiento con el ánimo de obtener de primera mano la experiencia de usuario y corroborar las	- Usabilidad - Conectividad

mercado EL DORADO.	necesidades específicas.	- Automatización
Levantamiento de la información principal	Se realizará un análisis de cada uno de los procesos funcionales y de dominio de la aplicación	- Usabilidad - Conectividad - Automatización
Análisis	Se realizará un análisis exhaustivo de la funcionalidad del negocio vs la arquitectura y requisitos técnicos de la herramienta	- Usabilidad - Conectividad - Automatización
Diseño	Se realizará el modelamiento y toda la ingeniería del proyecto para soportar la funcionalidad esperada, así como los requisitos técnicos y de dominio del sistema apoyada en esquemas de modelo, de procesos y diferentes artefactos de la ingeniería de requerimientos	- Usabilidad - Conectividad - Automatización
Creación artefactos iniciales	Se creará un proyecto tipo solución que albergará diferentes tipos de proyectos de librerías y el aplicativo web en cuestión utilizando el entorno de desarrollo integrado Visual Studio	- Conectividad
Programar funcionalidades generales para el sistema MVC (Modelo Vista Controlador).	Se crearán clases que proporcionarán servicios transversales para ser consumidos desde diferentes áreas en la aplicación.	- Conectividad
Login del sistema	Se deberá codificar toda la funcionalidad de ingreso seguro al aplicativo por medio de una pantalla donde se ingresen las credenciales de acceso.	- Usabilidad
Seguridad del sistema	Se realizará la codificación de pantallas, clases, controladores, vistas, procedimientos almacenados y demás objetos relacionados con la funcionalidad administrativa (usuarios, roles, administración de accesos) del sistema.	- Conectividad
Definir la arquitectura del sistema	Basado en las mejores prácticas del desarrollo de software se realizarán la inclusión de todos los elementos, artefactos, componentes, códigos y demás ítems relacionados con el proyecto de manera organizada, con nombramientos que sean dicientes de lo que se pretende realizar en cada uno de los segmentos de código del sistema.	- Usabilidad - Conectividad - Automatización
Definir la estructura de la base de datos	Se crearán esquemas específicos para cada bloque funcional del sistema. Para este caso en particular se creará el esquema de nombre ADM (Siglas de Administración) en el motor de base de datos junto con las tablas conservando una normalización ideal y que permita aprovechar el espacio y el rendimiento del motor de una manera eficaz.	- Conectividad

Crear modelamiento o capa de persistencia de datos.	Utilizando el framework de desarrollo ORM Entity Framework de Microsoft se creará una capa de acceso a datos a través de objetos y clases del Framework de trabajo .NET de Microsoft.	- Conectividad
Crear funcionalidad específica para módulos de venta	Teniendo en cuenta el marco teórico con respecto a la usabilidad y la experiencia adquirida y documentación adicional, crear la funcionalidad de ventas del sistema	- Usabilidad
Crear funcionalidad correspondiente a un sistema informático de tipo POS	Teniendo en cuenta el marco conceptual, donde se especifica las particularidades de un sistema informático tipo POS, desarrollar cada una de las funcionalidades que tengan lugar, como lo son: registro de proveedores, registros de clientes, registro de usuarios del sistema, entre otros.	- Conectividad
Crear la funcionalidad de pedidos automática	Analizando la documentación especifica recolectada en el levantamiento de la información realizar el diseño de la funcionalidad que permita automatizar pedidos en el sistema	- Automatización

3- Ingeniería del proyecto

3.1 Modelo del dominio

Hace referencia al conjunto de entidades del negocio representadas en módulos o componentes dentro del sistema. En la imagen 2 se puede observar cada uno de los elementos principales del dominio de la aplicación, donde también se puede evidenciar un flujo de proceso entre las diferentes entidades.

Un punto de partida para este modelo se puede evidenciar con la entidad de dominio proveedor, donde regularmente él, realiza visitas al mini mercado EL DORADO, con el ánimo de programar el pedido a entregar. Generalmente este proceso tiene un lapso de entrega inmediato para algunos casos hasta cuatro días de anticipación para otros. El motivo de la visita es verificar que productos desea pedir el mini mercado, así como ofrecer promociones o productos nuevos. Como se puede evidenciar, existe una relación directa con lo que es proveedor-inventario y proveedor-usuario del sistema, ya que es quien informa al proveedor cuales son los productos necesitados y concuerdan con el proveedor que se compromete a recibir en la entrega del pedido.

Una entidad muy importante del dominio de la aplicación son los usuarios. Por usuarios entendemos a los vendedores y administradores del sistema. Son quienes hacen uso de la herramienta informática y como se mencionó anteriormente, tienen relación directa con los proveedores, como también se puede evidenciar según la imagen 2, tienen relación directa con el módulo de reportes del sistema, el módulo de ventas, el módulo de inventario, a través de la entidad o esquema de seguridad de la aplicación. Haciendo uso de este esquema de seguridad, tienen relación con la entidad clientes del dominio de la aplicación, en el proceso de registro de las ventas, créditos y demás funcionalidades expuestas por la herramienta informática.

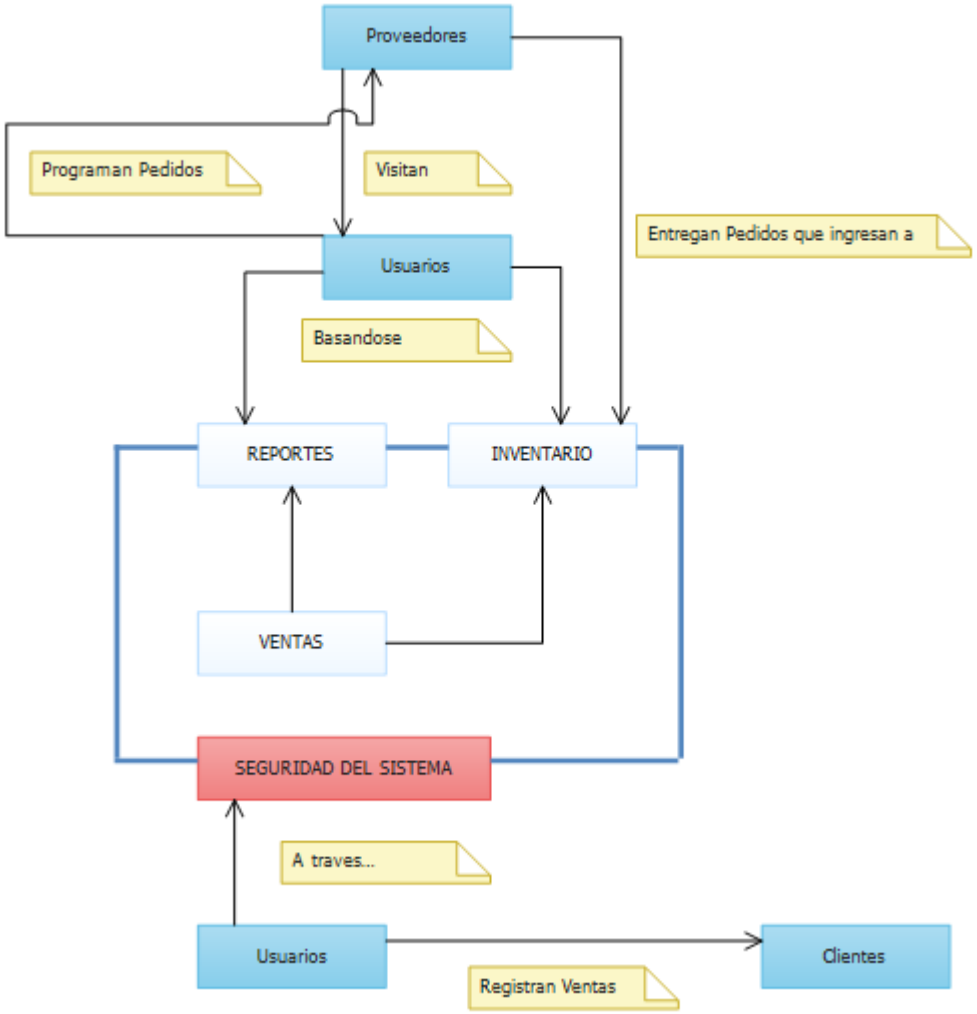


Imagen 2. Modelo de conceptos de dominio del sistema

3.2 Modelo de casos de uso

Este modelamiento permite detallar individualmente la funcionalidad específica de los módulos y componentes del sistema, para su desarrollo, documentación e implementación en el proceso del ciclo de vida del software. Se ha utilizado la notación ADM ("Administración"), para todos los casos de uso relacionados con la administración de la herramienta y componentes transversales

que proporcionan una funcionalidad general de cualquier sistema informático, como lo es un sistema de ingreso, registro y dada de alta de usuarios, permisos de acceso a los diferentes módulos y funcionalidades del sistema, como manejo de roles. Adicionalmente se ha utilizado la notación BUS ("Business/Negocio"), para todos los casos de uso relacionados con componentes y funcionalidades específicas del negocio como lo son: el registro de proveedores, registro de clientes, registros de ventas, registro de productos, inventario y reportes del sistema.

La elección de este tipo de documentación tiene como objetivos:

- Unificar los conceptos entre los diferentes actores del equipo de trabajo.

- Encontrar un común acuerdo entre las definiciones expuestas por el mini mercado EL DORADO junto con la ingeniería de requerimientos y propuesta de desarrollo

- Definir un alcance detallado de la funcionalidad requerida esencial por la herramienta, así como definir que esta por fuera del alcance del proyecto.

- Entender los procesos de negocios relacionados con la funcionalidad del caso de uso.

- Priorizar las tareas.

- Descubrir brechas entre la funcionalidad requerida y la desarrollada.

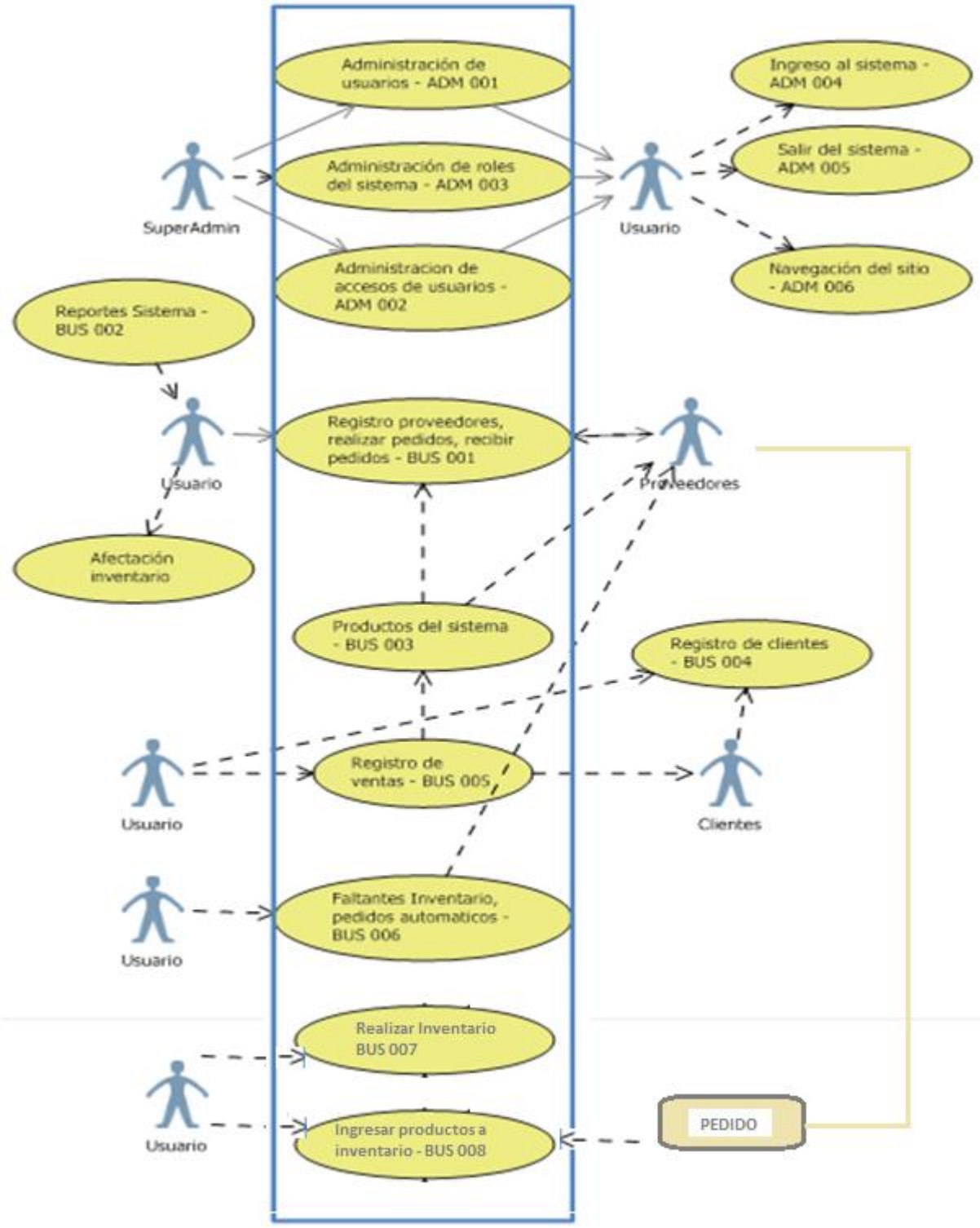


Imagen 3. Modelo de casos de uso.

3.2.1 Caso de uso: ADM – 001

ID	ADM-001
Titulo	Administración de usuarios del sistema
Descripción	Mediante una pantalla donde inicialmente se verán los actuales usuarios del sistema, el administrador o el usuario que cuente con el rol especificó, podrá ingresar nuevos usuarios al sistema, eliminar actuales y actualizar datos relacionados con uno existente.
Actores	Administrador del sistema
Precondiciones	Es preciso contar con el esquema de base de datos previamente creado e incluido correctamente al modelo ORM. Igualmente se debe contar al menos con un rol del sistema 'administrador' con las páginas de administración habilitadas para este (relacionado con caso de uso ADM-002), un usuario ingresado en el sistema con el rol de administrador asociado para que pueda realizar operaciones de ingreso de usuarios nuevos. El usuario que utiliza esta página debe ser un usuario existente del sistema, por lo general, en el primer uso se deberá ingresarse este registro por base de datos para garantizar la usabilidad de la página, ya que es una página de la administración del sistema y la precondición para su uso es que únicamente pueda ser accedida por un rol tipo administrador.
Postcondiciones	Una vez el usuario administrador utilice esta funcionalidad y realice operaciones de guardado, pueden presentarse los siguientes eventos como resultado de esa operación: • Se Ingresa un usuario nuevo al sistema • Se elimina un usuario del sistema • Se actualiza la información de un usuario del sistema
Escenario Principal	El usuario administrador ingresa al sistema con sus credenciales. Siendo estas validadas el administrador por lo general estando dentro de la parametrización estándar del sistema podrá acceder a todas las páginas de administración del sistema como lo es la pantalla de administración de usuarios. En este punto el usuario administrador podrá visualizar todos los usuarios actuales del sistema y podrá ejecutar las acciones de: actualización, eliminación.

	Además de estos flujos anteriormente mencionados el usuario podrá ingresar un nuevo usuario, este flujo, que es el principal del caso de uso tendrá las siguientes características: En la página en cuestión existirá un botón con el cual el administrador podrá ingresar un nuevo usuario, al ejecutar esta acción el usuario podrá visualizar un formulario en el cual podrá diligenciar la información del usuario nuevo a ingresar al sistema. En este punto de este flujo el usuario puede cancelar esta opción y el sistema lo redirigirá a la vista donde se visualizan todos los usuarios nuevamente. Si el administrador decide ingresar al usuario deberá diligenciar los siguientes campos: • Nombre de usuario (campo de texto) • Rol (campo de selección única) • Lenguaje (campo de selección única) Todos los campos son necesarios y existe una precondición para el campo de nombre de usuario y es que este debe ser único, es decir, el sistema realizará la verificación si el nombre de usuario ingresado por el administrador del sistema no exista previamente y en caso tal deberá avisar al usuario que se debe elegir un nombre de usuario diferente. Una vez validado este flujo y realizado se ingresará un nuevo usuario al sistema.
Escenario(s) Alternativo(s)	Así como se puede agregar un usuario nuevo al sistema, es posible actualizar los datos de un usuario existente del sistema. Este escenario contempla que el flujo se inicia cuando el administrador del sistema da click en el botón actualizar de la lista de usuarios existentes del sistema lo que el administrador visualizara una pantalla similar a la de crear un usuario nuevo, pero con los datos diligenciados del usuario en edición junto con el campo de nombre de usuario deshabilitado, para impedir colapso por nombramientos. En este flujo el usuario podrá modificar el lenguaje y el rol del sistema y podrá tomar dos acciones que son: actualiza la información nueva diligenciada del usuario en edición o cancela la acción actualmente en operación.

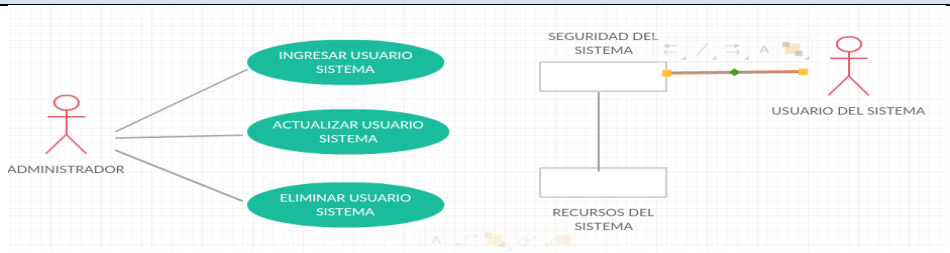
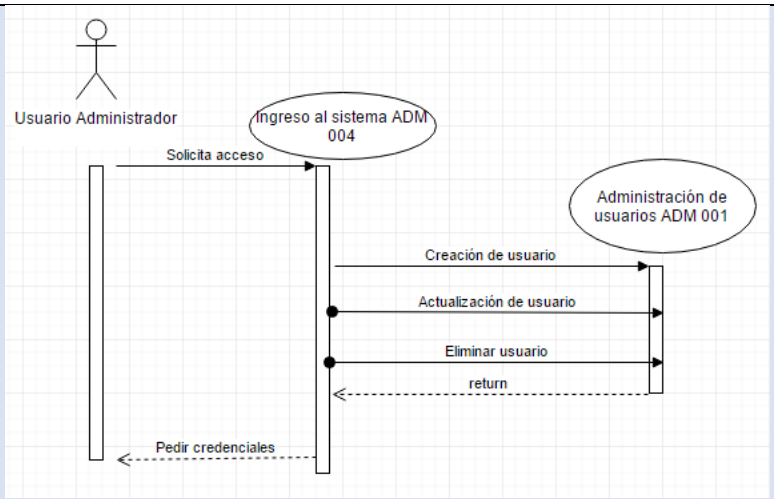
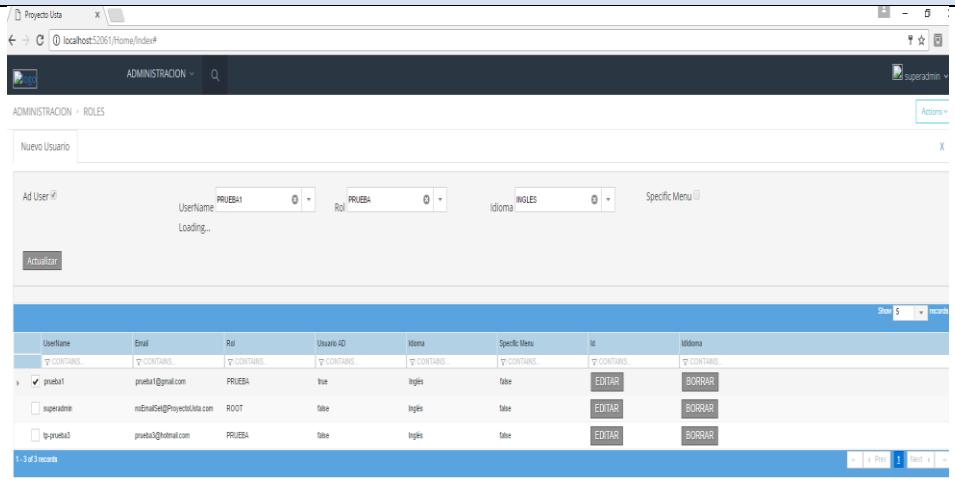
	Los campos para la edición del usuario existente son: • Username: (Solo lectura) • Rol (campo de selección única) • Lenguaje (campo de selección única) Todos los campos son obligatorios y es de comentar que el username es de solo lectura y no modificable. Otro flujo realizable en este caso por el actor administrador del sistema es el de la eliminación de usuario. Esto se realizará mediante la pantalla principal de usuarios activos donde en una columna específica para cada usuario habrá un botón que permita la eliminación. Al dar click en este punto el usuario podrá aceptar la eliminación o cancelarla, es una confinación necesaria debido a la naturaleza de la operación. En este punto es preciso aclarar que el usuario no será borrado físicamente de la base de datos salvo el punto en el que el usuario no tenga registros o huellas en el sistema. Para el caso donde exista una marca en otros registros del sistema para el usuario en eliminación, este se pondrá en un estado de inactivo tanto para no alterar los registros historiales de este, así como no dejar huérfanos ningunos datos correspondientes a este usuario.
Prioridad	1
Ámbito	Administración del sistema
Frecuencia de Uso	Baja
Ilustración	 *Imagen 4. Ilustración caso de uso ADM-001*

Diagrama de Secuencia	 *Imagen 5. Diagrama de secuencia caso de uso ADM-01*
Mockup	 *Imagen 6. Mockup pantalla caso de uso ADM-01.*

3.2.2 Caso de uso: ADM – 002

ID	ADM – 002
Titulo	Administración de opciones del sistema
Descripción	Mediante una pantalla donde inicialmente se verán los actuales módulos y páginas del sistema, el administrador o el usuario que cuente con el rol específico de administrador, podrá ingresar nuevos módulos y/o paginas al sistema, eliminar módulos y/o paginas actuales y actualizar datos relacionados con un módulo y/o página existente como lo son la ruta física, pagina o controlador que este iniciara, el nombre del módulo en el idioma español o inglés así como el orden es que este aparecerá en el menú principal del

	sistema.
Actores	Administrador del sistema
Precondiciones	Es preciso contar con el esquema de base de datos previamente creado e incluido correctamente al modelo ORM. Igualmente se debe contar al menos con un rol del sistema 'administrador' con las páginas de administración habilitadas para este (relacionado con caso de uso ADM-002), así como deben existir previamente los registros de páginas de administración creadas junto con el módulo que las contiene, adicional un usuario ingresado en el sistema con el rol de administrador asociado para que pueda realizar operaciones de ingreso de usuarios nuevos. El usuario que utiliza esta página debe ser un usuario existente del sistema, por lo general, en el primer uso se deberá ingresarse este registro por base de datos para garantizar la usabilidad de la página, ya que es una página de la administración del sistema y la precondición para su uso es que únicamente pueda ser accedida por un rol tipo administrador.
Postcondiciones	Una vez el usuario administrador utilice esta página y haga modificación en los datos pueden presentarse los siguientes eventos como resultado de esa operación: • Se Ingresa un módulo o página nuevo al sistema • Se elimina un módulo o página existente del sistema Se actualiza la información de un módulo o página existente tal como el nombre del módulo o página en inglés o español, así como el orden en que este aparecerá en el menú principal.
Escenario Principal	El usuario administrador ingresa al sistema con sus credenciales. Siendo estas validadas el administrador por lo general estando dentro de la parametrización estándar del sistema podrá acceder a todas las páginas de administración del sistema como lo es la pantalla de opciones. En este punto el usuario administrador podrá visualizar todos los módulos y páginas actuales del sistema y podrá ejecutar varias acciones con estos como son: (actualización, eliminación). Además de estos dos flujos anteriormente mencionados el usuario podrá ingresar un nuevo módulo o página al sistema, este flujo que es el principal del caso de uso tendrá las siguientes características: En la página en cuestión existirá una caja de texto para módulo nuevo y otra

	caja de texto para página nueva junto con el botón crear y eliminar. Dependiendo la selección actual del usuario administrador tendrá una u otra habilitada. Cuando no hay selección alguna la caja de texto de nuevo módulo estaría habilitada y para cuando hay un módulo o una página seleccionada el usuario administrador encontrará la opción de página nueva únicamente habilitada. En este punto de este flujo el usuario puede crear una nueva opción o simplemente navegar los módulos actuales y sus propias páginas. Sí el administrador decide ingresar al usuario deberá diligenciar los siguientes campos: • Nombre del Módulo (campo de texto) • Nombre de la página (campo de texto) El campo es necesario y único para cada caso en particular y existe una validación general para ambos y es que no permite que existan nombre de módulos o página duplicados literalmente por lo que el nombre que se ingrese debe ser único, es decir, el sistema realizará la verificación si el nombre de modulo o página ingresado por el administrador del sistema no exista previamente y en caso tal deberá avisar al usuario de que se debe elegir un nombre de usuario diferente. Una vez validado este flujo y realizado se ingresará un nuevo módulo o página al sistema y para el cual se realizará la selección automática y así dando visualización a las opciones existentes para este módulo o página. En este punto se podrán realizar las asignaciones de nombres para el idioma inglés o español del módulo y el orden es que este aparecerá en el módulo principal. En caso de haberse realizado una creación de página nueva se mostrará la visualización de las opciones relacionadas con la página como son, la página o el controlador junto con la acción que esta opción redirigiera en el sistema, así como los nombres en inglés y español de la página en cuestión.
Escenario(s) Alternativo(s)	Así como se puede agregar un nuevo módulo o página al sistema, es posible actualizar los datos de un módulo o página existente del sistema. Este escenario contempla que el flujo se inicia cuando el administrador del sistema da click en el ítem en cuestión. Se diferenciará al módulo porque este tendrá la opción de despliegue cuando contenga páginas o porque en la tabla de visualización no se tendrá para esta opción una configuración de controlador y página. Al seleccionar el ítem tipo módulo el sistema visualizará

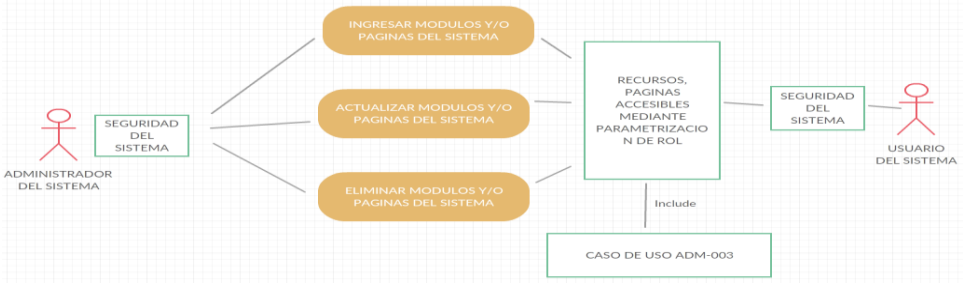
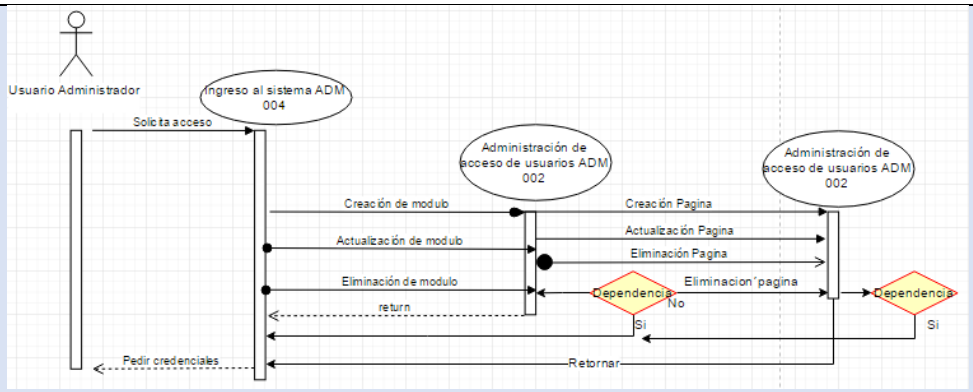
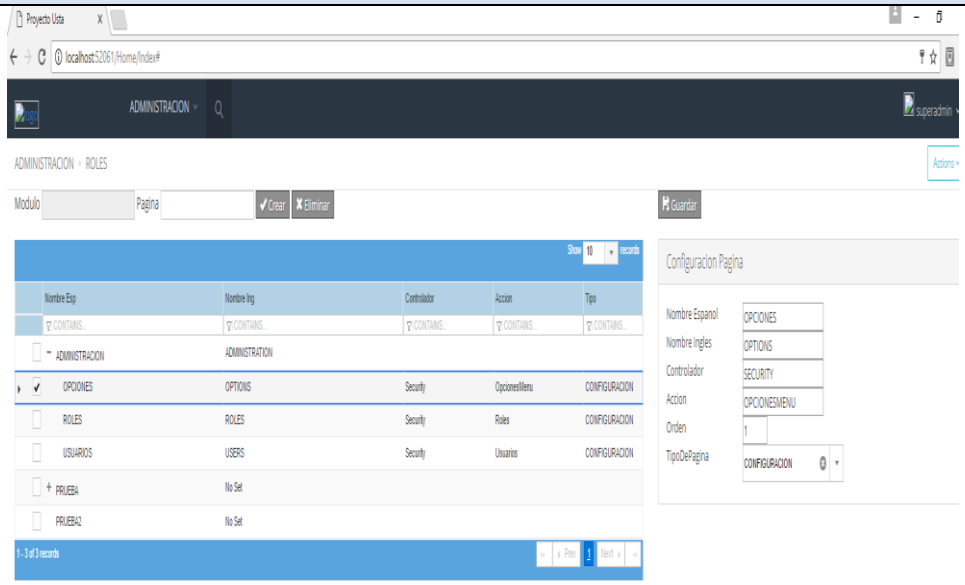
	los campos concernientes a actualizar los datos del módulo como lo son el nombre del módulo en inglés o español, así como el orden de aparición de este en el menú principal. Para cuando sea la selección de un ítem de tipo página, se visualizará los campos concernientes a la página que son el nombre en inglés y español de esta opción tipo página, así como el controlador y la acción que el sistema iniciara, así como el orden es que aparecerá esta página en el menú principal tanto como el tipo de opción de página que esta es. Adicional a los flujos anteriormente mencionados esta la opción de que el usuario administrador puede realizar la eliminación de una página de un módulo especifico, así como un módulo completo junto con sus páginas que la contienen. Para realizar esta labor es preciso seleccionar el ítem en cuestión bien sea un módulo o una página específica y el sistema activara el botón eliminar de la pantalla. En este momento del flujo al dar click en el botón eliminar de la pantalla el usuario será preguntado una vez más si desea realizar la operación solicitada y en podrá continuarla o cancelarla. Para el primer caso el sistema realizará la eliminación de la página o módulo junto con todas sus páginas del esquema de base de datos del menú principal del sistema y de permisos. En caso de realizar una eliminación, el sistema validará que no existan roles que contengan parametrizadas opciones que se estén eliminando en el momento.
Prioridad	2
Ámbito	Administración del sistema
Frecuencia de uso	Baja
Ilustración	 *Imagen 7. Ilustración caso de uso ADM-002.*

Diagrama de Secuencia	 *Imagen 8. Diagrama de secuencias caso de uso de ADM-002.*
Mockup	 *Imagen 9. Mockup pantalla caso de uso ADM-002.*

3.2.3 Caso de uso: ADM – 003

ID	ADM – 003
Titulo	Administración de roles del sistema
Descripción	Mediante una pantalla donde inicialmente se verán los actuales roles del sistema, el administrador o el usuario que cuente con el rol específico de administrador, podrá ingresar nuevos roles al sistema, eliminar actuales y actualizar datos relacionados con uno existente como lo es ingresar usuarios al rol seleccionado de manera ágil y asignar a que paginas tendrá acceso el rol

	seleccionado.
Actores	Administradores del sistema.
Precondiciones	Es preciso contar con el esquema de base de datos correspondiente previamente creado e incluido correctamente al modelo ORM. Igualmente se debe contar al menos con un usuario ingresado en el sistema y configurado como administrador del mismo para que pueda realizar operaciones de ingreso de usuarios nuevos. El usuario que utiliza esta página debe ser un usuario existente del sistema, por lo general, en el primer uso se deberá ingresarse este registro por base de datos para garantizar la usabilidad de la página, ya que es una página de la administración del sistema y la precondición para su uso es que únicamente pueda ser accedida por un rol tipo administrador.
Postcondiciones	Una vez el usuario administrador utilice esta página y haga modificación en los datos pueden presentarse los siguientes eventos como resultado de esa operación: • Se Ingresa un rol nuevo al sistema • Se elimina un rol del sistema Se actualiza la información de un rol específico tal como los usuarios que tienen configurado este rol, así como los módulos y páginas a las que este rol tiene acceso.
Escenario Principal	El usuario administrador ingresará al sistema con sus credenciales. Siendo estas validadas el administrador por lo general estando dentro de la parametrización estándar del sistema podrá acceder a todas las páginas de administración del sistema como lo es la pantalla de usuarios. En este punto el usuario administrador podrá visualizar todos los roles actuales del sistema y podrá ejecutar varias acciones con estos como son: (actualización, eliminación). Además de estos dos flujos anteriormente mencionados el usuario podrá ingresar un nuevo rol, este flujo que es el principal del caso de uso tendrá las siguientes características: En la página en cuestión existirá un botón con el cual el administrador podrá ingresar un nuevo rol, al ejecutar esta acción el usuario podrá visualizar un formulario en el cual podrá diligencia la información del rol nuevo a ingresar al sistema. En este punto de este flujo el usuario puede cancelar esta opción y

	el sistema lo redirigirá a la vista donde se visualizan todos los roles nuevamente. Sí el administrador decide ingresar al usuario deberá diligenciar los siguientes campos: • Nombre del rol (campo de texto) El campo es necesario y existe una validación que no permite que existan nombre de roles duplicados literalmente por lo que el nombre que se ingrese debe ser único, es decir, el sistema realizará la verificación si el nombre de rol ingresado por el administrador del sistema no exista previamente y en caso tal deberá avisar al usuario de que se debe elegir un nombre de usuario diferente. Una vez validado este flujo y realizado se ingresará un nuevo rol al sistema y para el cual se podrán realizar las acciones de asociar usuarios previamente creados en el sistema en el rol que se tiene seleccionado, así como seleccionar de manera jerárquica que modulo y a que páginas de que modulo tiene acceso el rol que se está editando. Para estas acciones cada una tiene un flujo diferente como por ejemplo para el caso de las opciones a escoger estas debería haber previamente creadas por el caso de uso ADM-002 donde se creará una estructura de menú y agrupamiento de páginas por menú, por lo que el administrador al rol en cuestión de edición de este caso de uso le asociara paginas o módulos de acceso según la necesidad. La visualización de esta estructura preferiblemente debería ser en modo jerárquico donde al seleccionar un módulo en particular, este debería asociar todas las páginas contenidas en este y un comportamiento similar cuando la opción es la de deseleccionar. Adicional a este flujo esta la opción de asociar usuarios y desasociarlos dentro de la misma página de roles. Para el ingreso es necesario colocar el nombre en una caja de texto con función de autocompletar para ubicar al usuario que se desee agregar. Para quitar a un usuario se deberá dar click en el botón eliminar de la grid de visualización de usuarios del rol.
Escenario(s) Alternativo(s)	Así como se puede agregar un rol nuevo al sistema, es posible actualizar los datos de un rol existente del sistema. Este escenario contempla que el flujo se inicia cuando el administrador del sistema da click en el botón seleccionar rol de la lista roles existentes del sistema con lo que el administrador visualizara las grid y segmentos visuales de los controles HTML relacionados con los usuarios relacionados a ese rol y

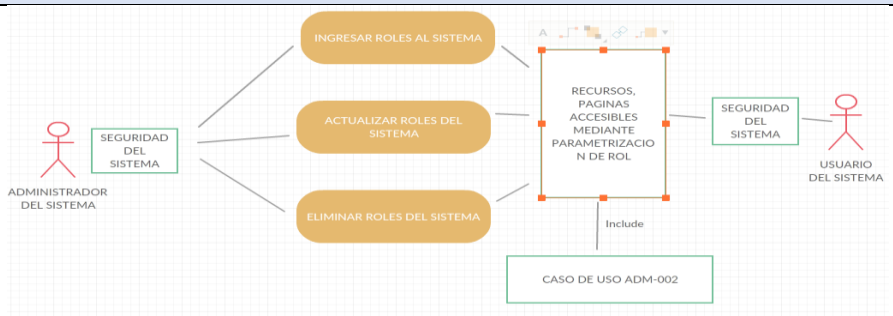
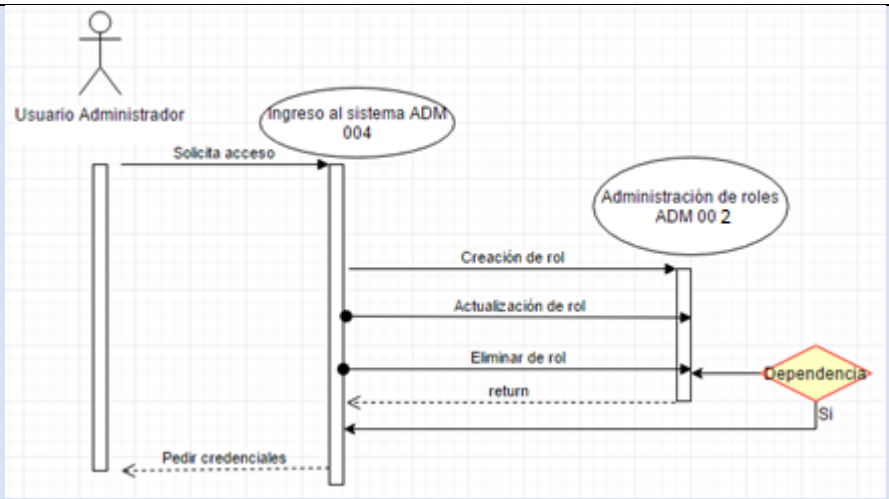
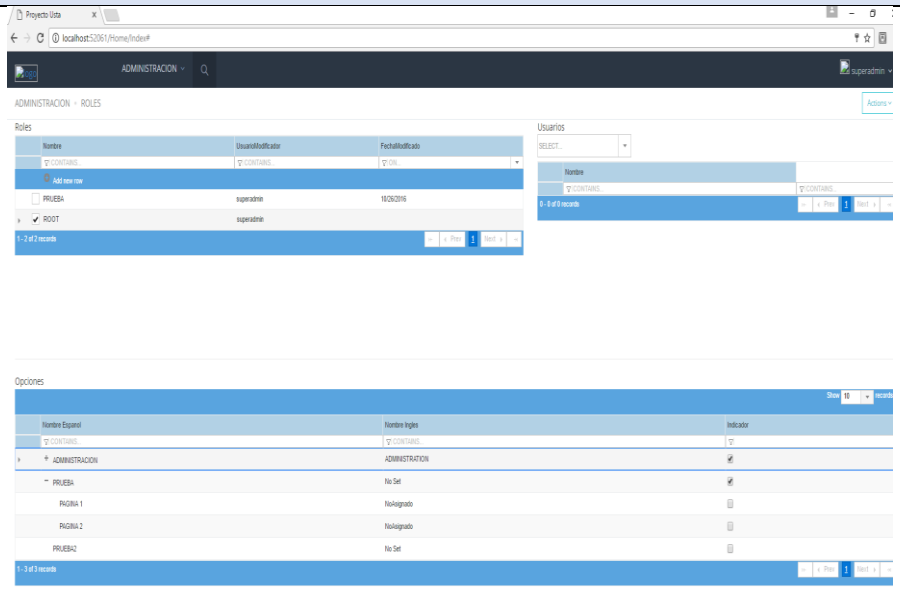
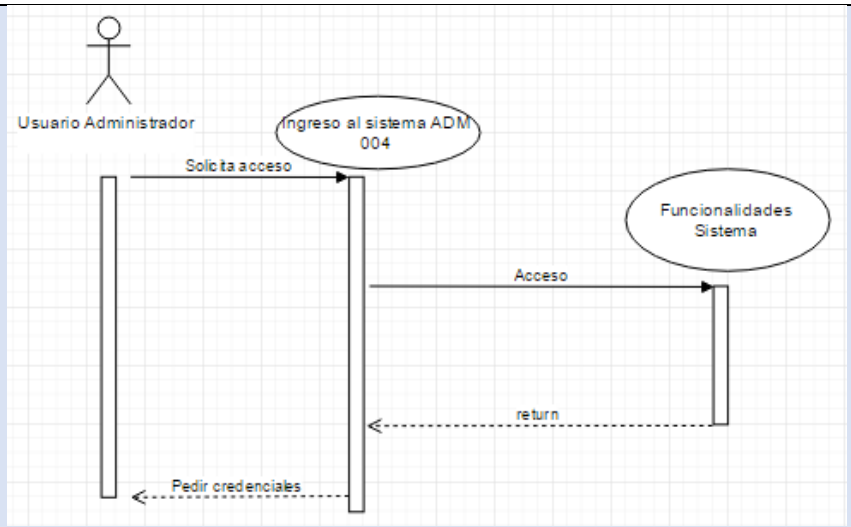
	los módulos y paginas asociados a este. Una vez seleccionado el rol se podrán realizar las acciones de asociar usuarios previamente creados en el sistema en el rol que se tiene seleccionado, así como seleccionar de manera jerárquica que módulo y a que páginas de que módulo tiene acceso el rol que se está editando. Para estas acciones cada una tiene un flujo diferente como por ejemplo para el caso de las opciones a escoger estas debería haber previamente creadas por el caso de uso ADM-002 donde se creará una estructura de menú y agrupamiento de páginas por menú, por lo que el administrador al rol en cuestión de edición de este caso de uso le asociara paginas o módulos de acceso según la necesidad. La visualización de esta estructura preferiblemente debería ser en modo jerárquico donde al seleccionar un módulo en particular, este debería asociar todas las páginas contenidas en este y un comportamiento similar cuando la opción es la de deseleccionar. Otro flujo realizable en este caso por el actor administrador del sistema es el de la eliminación del rol. Esto se realizará mediante la pantalla principal de roles activos donde en una columna específica para cada rol habrá un botón que permita la eliminación. Al dar click en este punto el usuario podrá aceptar la eliminación o cancelarla, es una confinación necesaria debido a la naturaleza de la operación. Una vez aceptado la confirmación de eliminación, el sistema borrará todos los metadatos de configuración asociados al rol y colocará a todos los usuarios en relación con el rol a eliminar con el rol básico del sistema.
Prioridad	3
Ámbito	Administración del sistema.
Frecuencia de uso	Baja
Ilustración	 *Imagen 10. Ilustración caso de uso ADM-003*

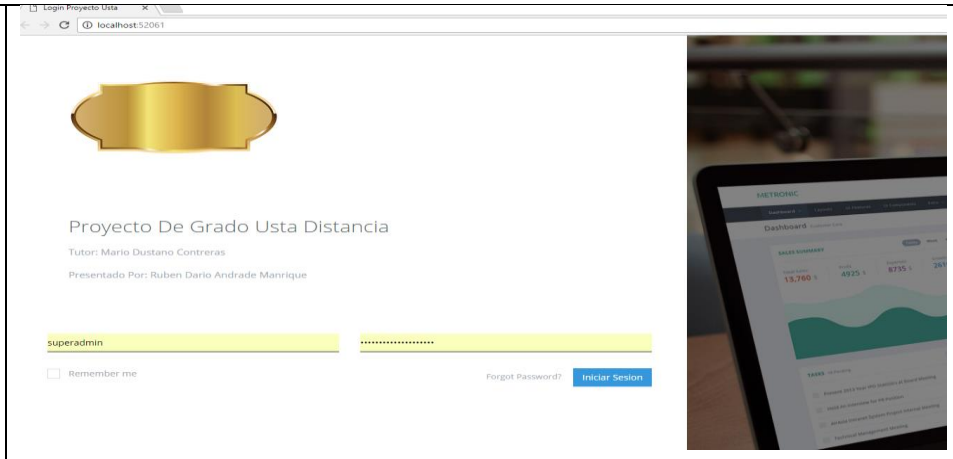
Diagrama de Secuencia	 *Imagen 11. Diagrama de secuencia caso de uso ADM-003.*
Mockup	 *Imagen 12. Mockup pantalla caso de uso ADM-003.*

3.2.4 Caso de uso: ADM – 004

ID	ADM – 004
Titulo	Login (Funcionalidad de ingreso) del sistema
Descripción	Mediante una pantalla de login donde al usuario del sistema se le pedirán sus credenciales de acceso se le permitirá ingresar o ser informado de cualquier eventualidad a un interesado en ingresar al sistema.

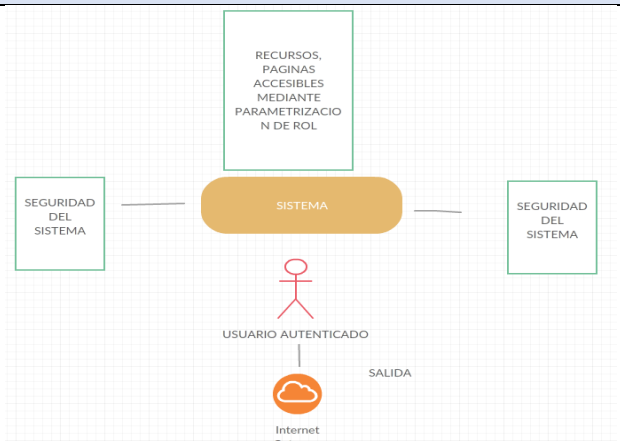
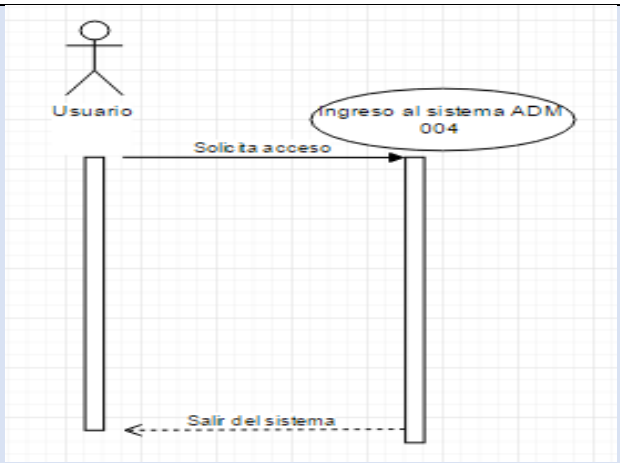
Actores	Usuarios del sistema.
Precondiciones	Es preciso contar con el esquema de base de datos previamente creado de usuarios del sistema e incluido correctamente al modelo ORM. Igualmente se debe contar al menos con un usuario dentro del sistema para validar el correcto acceso del mismo. Es un supuesto que, al momento de la instalación de los artefactos iniciales del sistema, este script incluirá en base de datos los registros necesarios para poder realizar el ingreso del administrador del sistema. El usuario que utiliza esta página debe ser un usuario existente del sistema, por lo general, en el primer uso se deberá ingresarse este registro por base de datos para garantizar la usabilidad de la página, ya que es una página de la administración del sistema y la precondición para su uso es que únicamente pueda ser accedida por un rol tipo administrador.
Postcondiciones	Una vez el usuario del sistema ingrese sus credenciales de acceso puede presentar los siguientes flujos de eventos resultados de su operación: • Realiza un ingreso exitoso al sistema y el motor de enrutamiento redirigirá a la página principal y cargara en pantalla las opciones a las que el usuario tiene acceso. Se realizará un intento de ingreso fallido por credenciales erróneas o inexistentes para el sistema.
Escenario Principal	El usuario ingresa a cualquier dirección, vinculo o demás recurso del sistema lo cual hará que el motor de asp.net atienda la petición Http y la procese para analizar su estado general y puede de que se trate de una petición valida, actualmente activa o de una petición no valida lo cual redirigirá a la pantalla de login comentada en cuestión en este caso de uso. Los campos solicitados por esta pantalla son: • Usuario (Campo de texto) (obligatorio) • Contraseña (campo de texto oculto) (obligatorio) Para este ingreso de datos los dos campos son obligatorios y formas las credenciales con las que el usuario se inscribió en el sistema. Es esta parte del flujo el usuario luego de ingresar sus respectivas credenciales estas se validarán por medio de las clases de seguridad junto con los recursos de almacenamiento de base de datos y así validará o negará la entrada al usuario.
Escenario(s)	Así como se pueden ingresar las credenciales correctas de un usuario valido

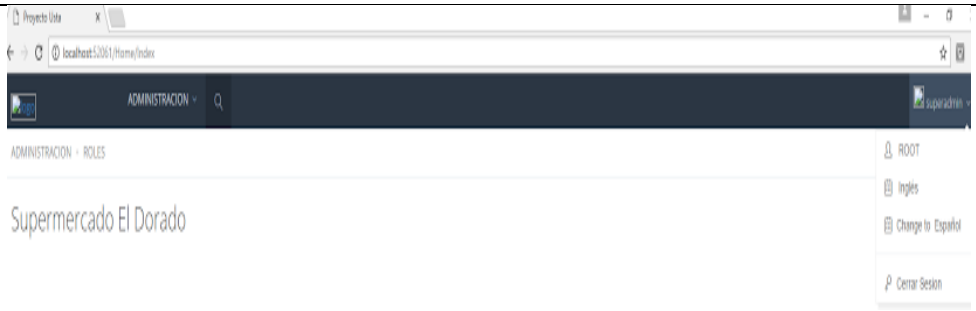
Alternativo(s)	para el sistema es posible también que se incurra en un error tipográfico o se trate de un intento de ingreso no autorizado. Para lo cual la respuesta de este flujo será tener una respuesta genérica que le indique al usuario de su error sin ser explícito en cuál de las equivocaciones ha cometido con el ánimo de fortalecer un sistema contra un ataque de fuerza bruta.
Prioridad	4
Ámbito	Administración del sistema
Frecuencia de uso	Alta
Ilustración	*Imagen 13. Ilustración caso de uso ADM-004.*
Diagrama de Secuencia	 *Imagen 14. Diagrama de secuencia caso de uso ADM-004.*

Mockup	 *Imagen 15. Mockup pantalla caso de uso ADM-004.*

3.2.5 Caso de uso: ADM – 005

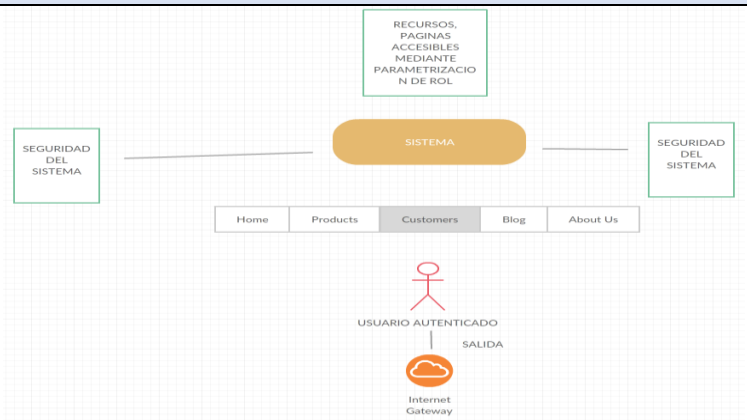
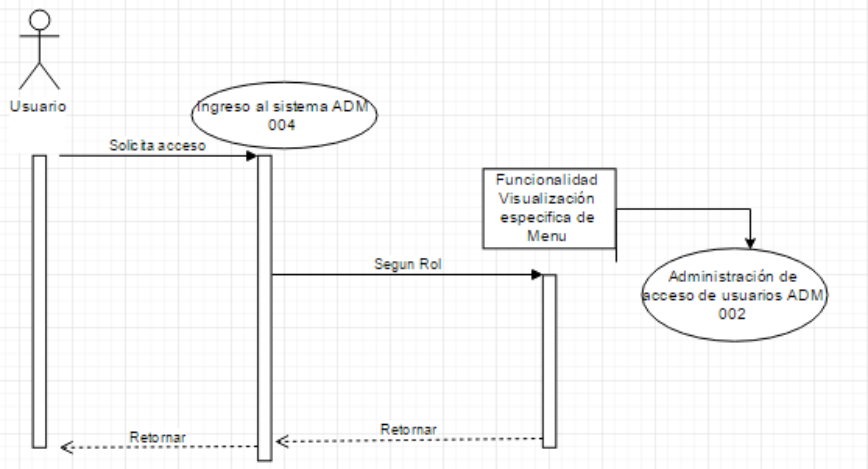
ID	ADM – 005
Titulo	Log Out (Funcionalidad de salida) del sistema.
Descripción	Mediante un vínculo que se ubicará en la ventana principal de trabajo una vez se ingresa al sistema, permitir al usuario finalizar sesión de manera segura.
Actores	Usuarios del sistema
Precondiciones	Es preciso tener al usuario del sistema logueado en el aplicativo y que pueda visualizar correctamente la pantalla de trabajo. El usuario que utiliza esta página debe ser un usuario existente del sistema, por lo general, en el primer uso se deberá ingresarse este registro por base de datos para garantizar la usabilidad de la página, ya que es una página de la administración del sistema y la precondición para su uso es que únicamente pueda ser accedida por un rol tipo administrador.
Postcondiciones	Una vez el usuario del sistema de click en el botón de logout se presentan los siguientes eventos en el flujo del caso: Se realizará un des logueo del sistema por parte del usuario, a nivel técnico el motor de ASP.NET liberará los recursos de sesión que el usuario o la sesión de este hacia uso tanto como recursos de memoria en el servidor, como recursos de cliente como cookies y demás objetos de seguridad. Luego se redirigirá a la pantalla de login del sistema
Escenario	El usuario se encuentra navegando el sistema, con una sesión abierta

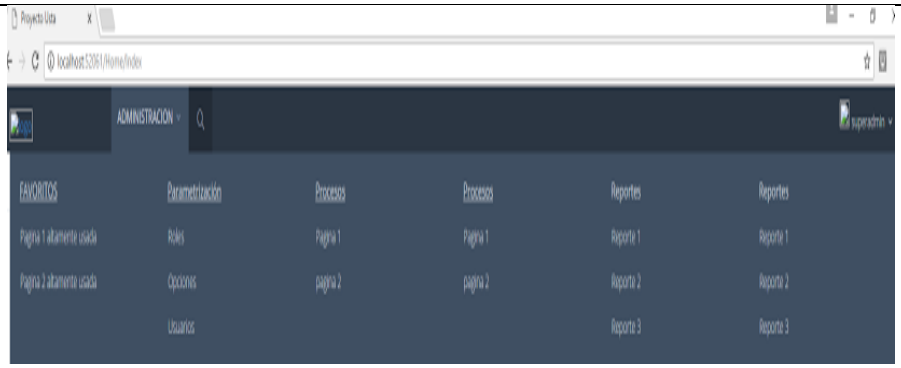
Principal	previamente otorgada por haberse logueado en el sistema con las credenciales de ingreso, por lo que si se desea terminar la sesión de trabajo del sistema el programa le habilitará la opción de salir del sistema por medio de un botón de logout.
Escenario(s) Alternativo(s)	El usuario puede abandonar la aplicación olvidando el proceso de log out detallado en este caso de uso. Por lo cual el sistema deberá cerrar la sesión del usuario después de un determinado tiempo de no recibir peticiones desde el cliente.
Prioridad	5
Ámbito	Administración del sistema.
Frecuencia de uso	Alta
Ilustración	 *Imagen 16. Ilustración caso de uso ADM-005.*
Diagrama de Secuencia	 *Imagen 17. Diagrama de secuencia caso de uso ADM-005.*

Mockup	 *Imagen 18. Mockup pantalla caso de uso ADM-005.*

3.2.6 Caso de uso: ADM – 006

ID	ADM – 006
Titulo	Menú principal del sistema
Descripción	Mediante un menú principal se le proporcionara acceso al usuario a los módulos y páginas disponibles dependiendo su rol en el sistema.
Actores	Usuarios del sistema
Precondiciones	Es preciso tener al usuario del sistema logueado en el aplicativo y que pueda visualizar correctamente la pantalla de trabajo. El usuario que utiliza esta página debe ser un usuario existente del sistema y contar con un rol obligatoriamente como el caso ADM-001 lo menciona. En el primer uso se deberá ingresar los registros relacionados con el menú habilitado para las páginas de administración por parte del usuario administrador.
Postcondiciones	Una vez el usuario sea validado para el ingreso al sistema, se visualizará un menú correspondiente al rol que este tiene configurado.
Escenario Principal	Al usuario que ha realizado el login satisfactorio del sistema se le redirigirá a la pantalla principal donde se visualizara un segmento de código HTML correspondiente a el menú de vínculos de acceso al sistema el cual dependerá específicamente del rol que este tenga configurado y la configuración de accesos de este rol a las diferentes opciones del sistema ADM-002.
Escenario(s) Alternativo(s)	Es posible ingresar a las opciones mediante el ingreso explícito de la dirección URL (uniform resource locator) en la caja de texto de entrada del navegador web. Esto requiere un conocimiento técnico de acceso a recursos web, lo cual no está contemplado como una opción de acceso de usuario final

	del sistema.
Prioridad	6
Ámbito	Administración del sistema.
Frecuencia de uso	Alta
Ilustración	 *Imagen 19. Ilustración caso de uso ADM-006.*
Diagrama de Secuencia	 *Imagen 20. Diagrama de secuencia caso de uso ADM-006.*

Mockup	 *Imagen 21. Mockup pantalla caso de uso ADM-006*

3.2.7 Caso de uso: BUS- 001

ID	BUS- 001
Titulo	Administración de proveedores
Descripción	Mediante acceso a una página contemplada dentro del módulo de BASE POS se proporcionará acceso al usuario del sistema para que ingrese a los proveedores y configure los días en los que visitan al mini mercado, así como los días que entregan los pedidos previamente programados.
Actores	Usuarios del sistema autorizados.
Precondiciones	Es preciso tener al usuario del sistema logueado en el aplicativo y que pueda visualizar correctamente la pantalla de trabajo. El usuario que utiliza esta página debe ser un usuario existente del sistema y contar con un rol obligatoriamente como el caso ADM-001 lo menciona. En la pantalla principal de este requerimiento funcional se proporcionará al usuario una pantalla para el ingreso/actualización y eliminación de proveedores del mini mercado El DORADO. Se proporcionará una caja de chequeo por cada día de la semana tanto para la visita como para la entrega del pedido previamente programado. Es preciso validar que el nombre del proveedor sea diligenciado obligatoriamente, así como que sea único dentro de la tabla o fuente de datos de persistencia de los mismos.
Postcondiciones	Una vez el usuario sea validado para el ingreso al sistema, se visualizará un menú correspondiente al rol que este tiene configurado.

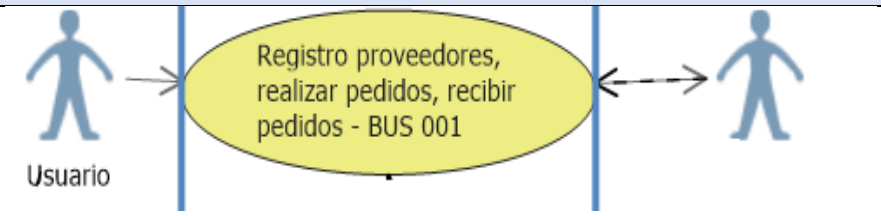
Escenario Principal	Al usuario que ha realizado el login satisfactorio del sistema y que tenga el rol con los permisos suficientes para poder visualizar la pantalla de proveedores se le proporcionará la pantalla donde se visualizaran los proveedores existentes registrados en el sistema. El usuario podrá actualizar un proveedor, eliminarlo y crear uno nuevo teniendo en cuenta las validaciones del caso de uso. El menú de acceso a la funcionalidad se ubicará dentro del módulo llamado "Base Pos" ilustrado en la imagen 23 del Mockup de este caso de uso. El grid o tabla de HTML que contenga la visualización de los proveedores, proporcionará un botón de editar para modificar proveedores existentes en el sistema y el botón nuevo para crear un nuevo proveedor, ilustrado en la imagen 23 del Mockup de este caso de uso. Al estar creando/actualizando, (ilustrado en la imagen 24 del Mockup de este caso de uso), se deben mantener las reglas de validación expuestas en el apartado de precondiciones. El sistema deberá permitir actualizar el nombre del proveedor contemplando las validaciones del caso de uso.
Escenario(s) Alternativo(s)	La eliminación y actualización del proveedor será posible contemplando las especificaciones de las precondiciones de este caso de uso.
Prioridad	7
Ámbito	Negocio del sistema.
Frecuencia de uso	Media
Ilustración	 *Imagen 22. Ilustración caso de uso BUS-001*

Diagrama de Secuencia

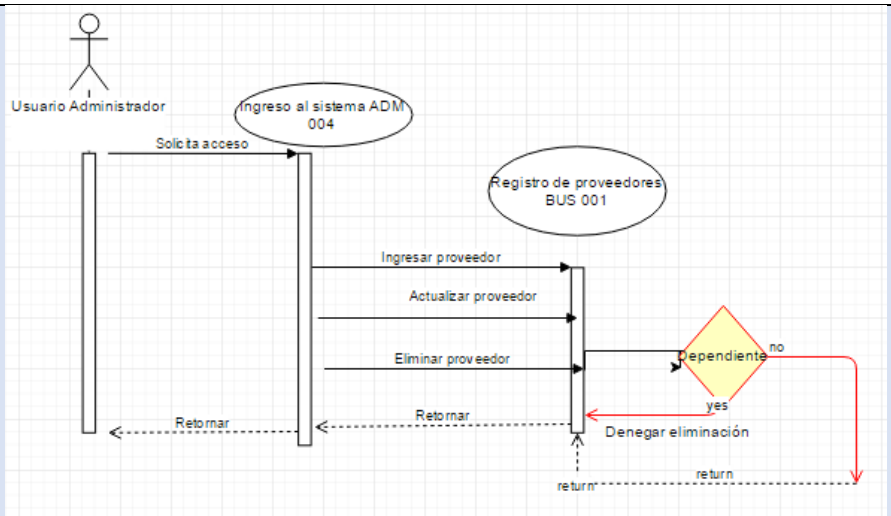


Imagen 23. Diagrama de secuencia caso de uso BUS-001.

Mockup

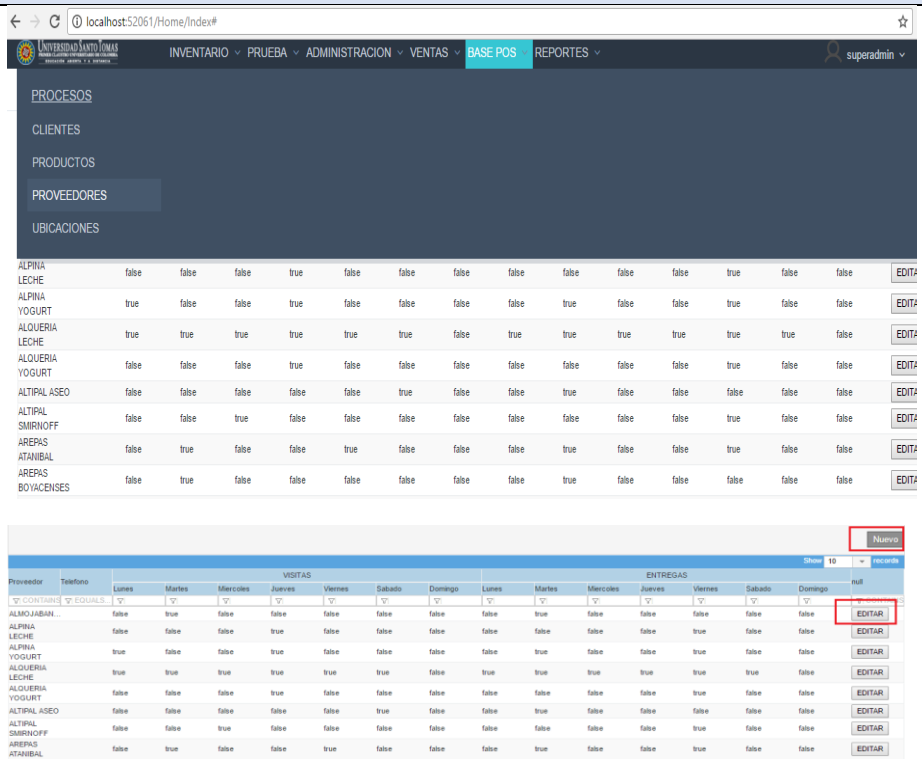


Imagen 24. Mockup pantallas caso de uso BUS-001

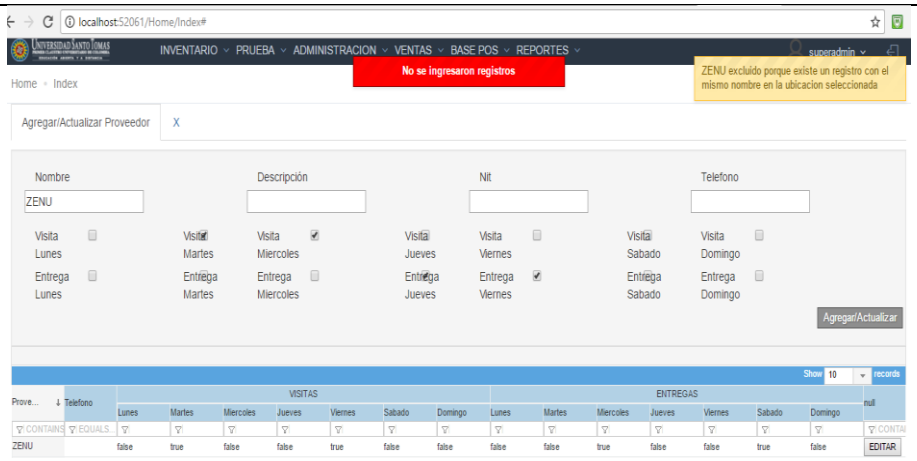


Imagen 25. Mockup pantalla caso de uso BUS-001

3.2.8 Caso de uso: BUS- 002

ID	BUS- 002
Titulo	Reportes del sistema
Descripción	Mediante acceso a una página contemplada dentro del módulo de reportes se proporcionará acceso al usuario del sistema para que tenga acceso a una variedad de reportes indicadores de créditos, ventas, inventario y movimientos de caja.
Actores	Usuarios del sistema autorizados.
Precondiciones	Es preciso tener al usuario del sistema logueado en el aplicativo y que pueda visualizar correctamente la pantalla de trabajo. El usuario que utiliza esta página debe ser un usuario existente del sistema y contar con un rol obligatoriamente como el caso ADM-001 lo menciona. En la pantalla principal de este requerimiento funcional se proporcionará al usuario una pantalla para la visualización de reportes relacionados con el registro y movimientos dentro del sistema. Las fechas de los filtros siempre deberán ser diligenciadas ambas para cuando se desee segmentar el resultado y deberá siempre ser fecha inicial menor a la fecha final.
Postcondiciones	Una vez el usuario sea validado para el ingreso al sistema, se visualizará un

	menú correspondiente al rol que este tiene configurado.
Escenario Principal	Al usuario que ha realizado el login satisfactorio del sistema y que tenga el rol con los permisos suficientes para poder visualizar la pantalla de proveedores se le proporcionara la pantalla donde se visualizaran los siguientes reportes del sistema: • Reporte de créditos (créditos actuales, log de esos créditos) • Reporte de movimientos de caja (con filtro para poder visualizar periodos de tiempo específicos) • Reporte de ventas (con filtro para poder visualizar periodos de tiempo específicos) • Total, inventario => El costo monetario total del inventario El usuario autenticado y con el rol específico podrá tener acceso a la funcionalidad de reporte de créditos. Los créditos son registrados al momento de realizar una venta y son específicamente tratados en el caso de uso BUS 004 y BUS 005. La pantalla de reporte de créditos proporcionara interface expuesta en la imagen 27, en el apartado de Mockup de este caso de uso. Al lado izquierdo de la pantalla se tendrá una previsualización general en modo resumen de los créditos actuales del sistema y al lado derecho se tendrá un log de esos créditos, contemplando el valor del crédito, el cliente al que se le otorgo ese crédito, el vendedor que lo realizó y la fecha de registro de la operación.
Escenario(s) Alternativo(s)	No aplica.
Prioridad	14
Ámbito	Negocio del sistema.
Frecuencia de uso	Alta.

Ilustración

Imagen 26. Ilustración caso de uso BUS-002.

Diagrama de Secuencia

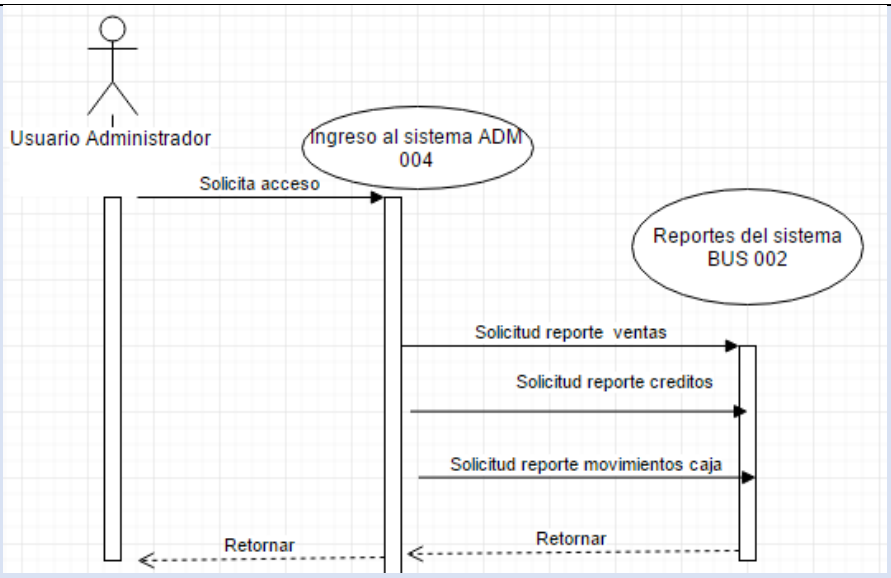


Imagen 27. Diagrama de secuencia caso de uso BUS-002.

Mockup

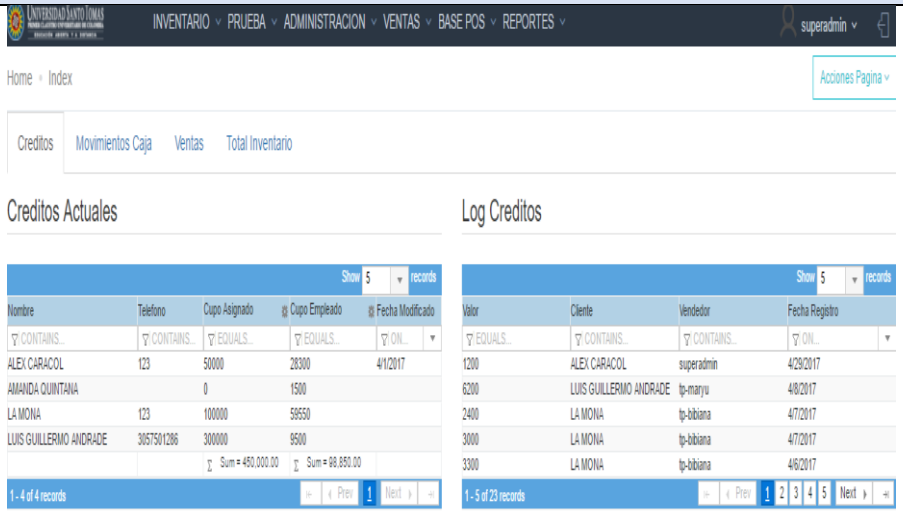
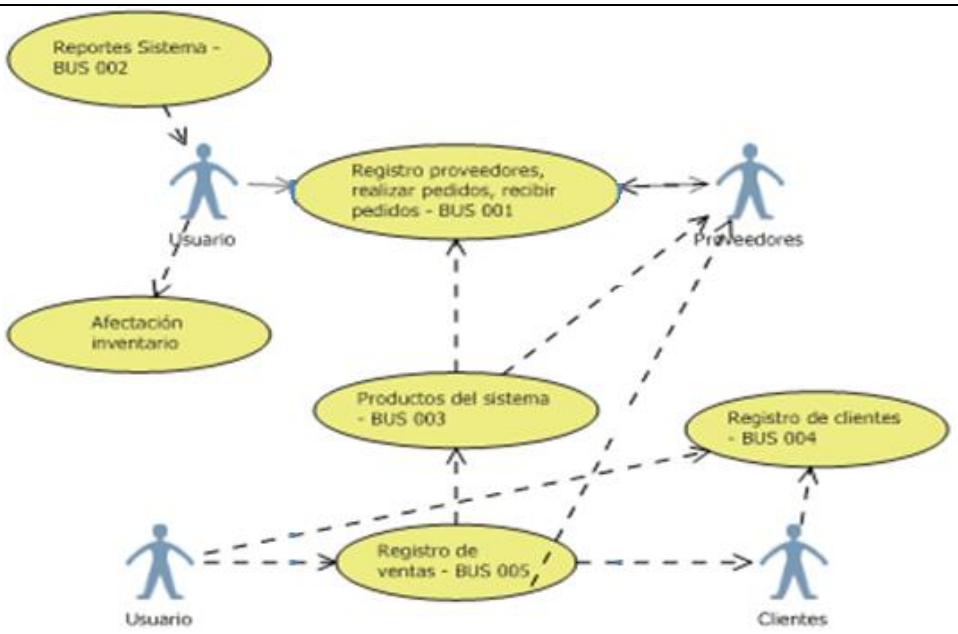
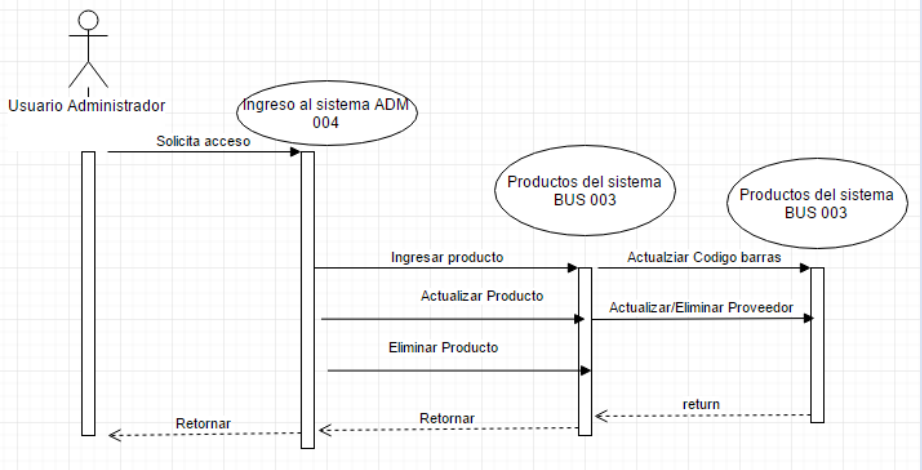


Imagen 28. Mockup pantalla caso de uso BUS-002

3.2.9 Caso de uso: BUS- 003

ID	BUS- 003
Titulo	Administración de productos
Descripción	Mediante acceso a una página contemplada dentro del módulo de “base pos” se proporcionará acceso al usuario del sistema para que tenga acceso a una página con la funcionalidad de ingresar, actualizar y eliminar productos del sistema.
Actores	Usuarios del sistema autorizados.
Precondiciones	Es preciso tener al usuario del sistema logueado en el aplicativo y que pueda visualizar correctamente la pantalla de trabajo. El usuario que utiliza esta página debe ser un usuario existente del sistema y contar con un rol obligatoriamente como el caso ADM-001 lo menciona. En la pantalla principal de este requerimiento funcional se proporcionará al usuario una pantalla para la visualización de productos en una tabla paginable por servidor ya que pueden llegar a ser demasiados productos en el sistema. Así mismo se proporcionará la funcionalidad de edición para cada registro creado, la funcionalidad de ingresar productos nuevos al sistema, así como eliminar existentes. Los campos obligatorios para la creación o actualización de un producto son: • Nombre • Precio • Qty (Cantidad)
Postcondiciones	Una vez el usuario sea validado para el ingreso al sistema, se visualizará un menú correspondiente al rol que este tiene configurado, mediante el cual en el módulo creado de “base pos” se encontrara con la página de productos.
Escenario Principal	La funcionalidad se accederá por medio del módulo “Base Pos”, ilustrado en la imagen 29, en la sección Mockup de este caso de uso. Al ingresar a la página principal podemos ver los productos que están registrados dentro del sistema, paginarlos, ordenarlos y filtrarlos mediante la tabla de navegación. De igual manera se pueden editar los valores de cada registro, así como eliminar el producto del sistema. Al seleccionar la opción de crear un nuevo producto (imagen 30), el sistema proporcionará la sección HTML de ingreso de un nuevo producto (imagen 31) que ofrecerá los siguientes campos:

	• Nombre (obligatorio) - Alfanumérico • Precio – Numérico • Cantidad Actual – Numérico • Cantidad Umbral – Numérico • Ubicación en el negocio – Selección • Ubicación en el stock o bodega – Selección • Proveedores (grid) • Códigos de barras (grid) Las validaciones pertinentes deberán informarse al usuario por medio de un mensaje como se ilustra en la imagen 32.
Escenario(s) Alternativo(s)	Es posible realizar operaciones de actualización y eliminación de productos desde la funcionalidad expuesta por esta página. Al actualizarse un producto, se ejecutan las mismas validaciones expuestas en la sección de precondiciones y escenario principal. En cuanto a la eliminación del producto, si el producto tiene un código de barras asociado, este no se eliminará, sino que se desactivará, y al momento de ingresar este código de barras a un producto que se esté intentado crear, se avisará al usuario de que el producto existe en base de datos, se encuentra desactivado y ofrece la opción de reactivación.
Prioridad	7
Ámbito	Negocio del sistema.
Frecuencia de uso	Alta

Ilustración	 *Imagen 32. Ilustración caso de uso BUS-003*
Diagrama de Secuencia	 *Imagen 13. Diagrama de secuencia caso de uso BUS-003*

Mockup

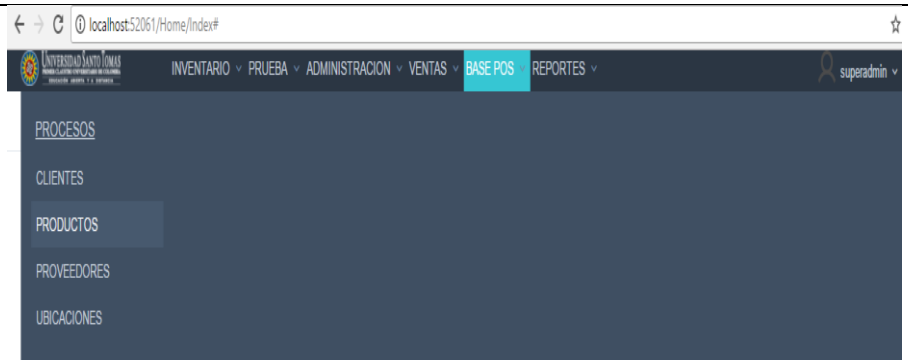


Imagen 29. Mockup pantalla caso de uso BUS-003

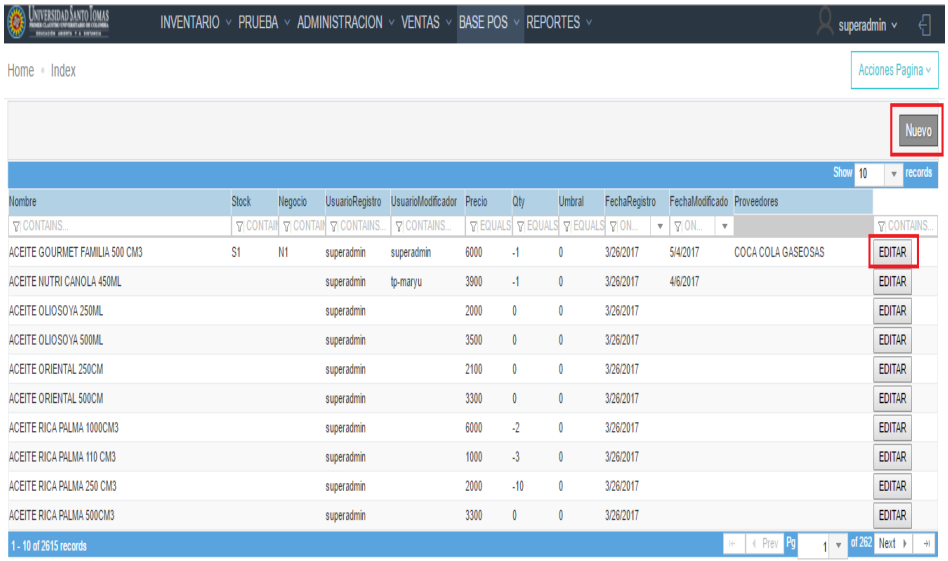


Imagen 30. Mockup pantalla caso de uso BUS-003

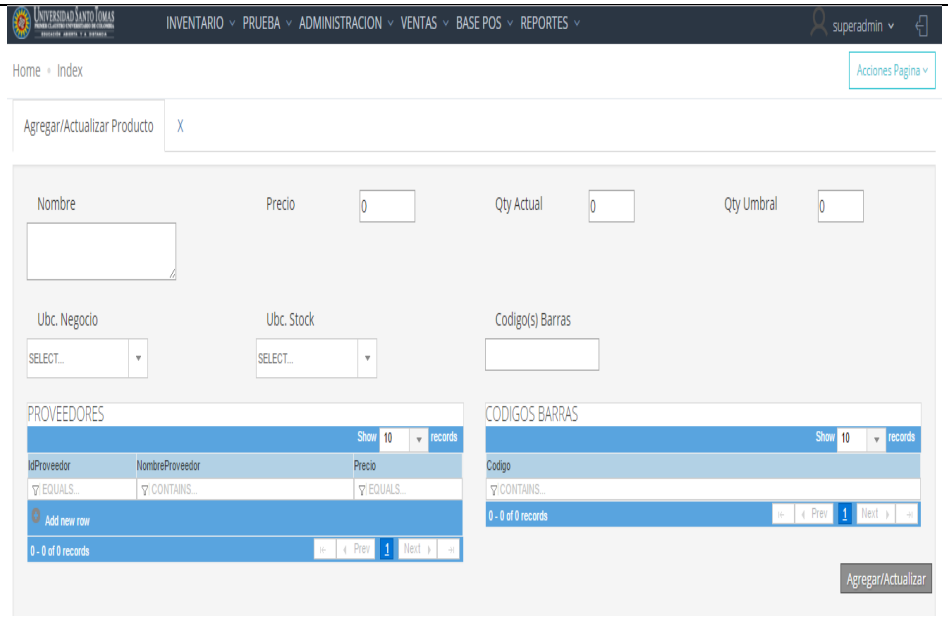


Imagen 32. Mockup pantalla caso de uso BUS-003

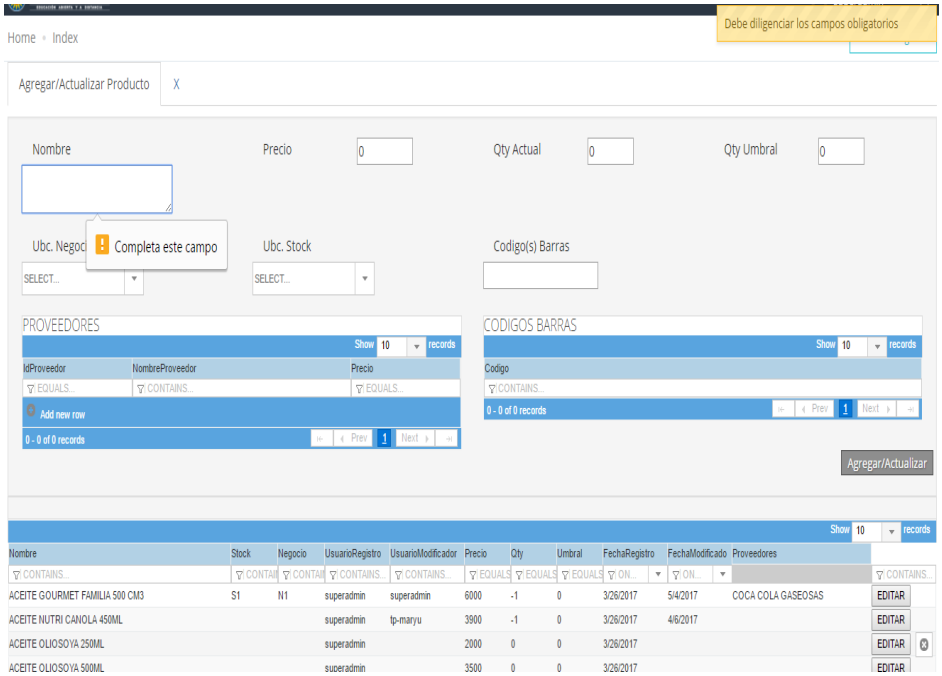
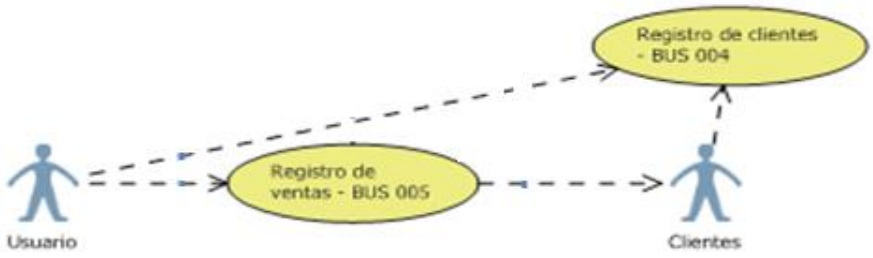
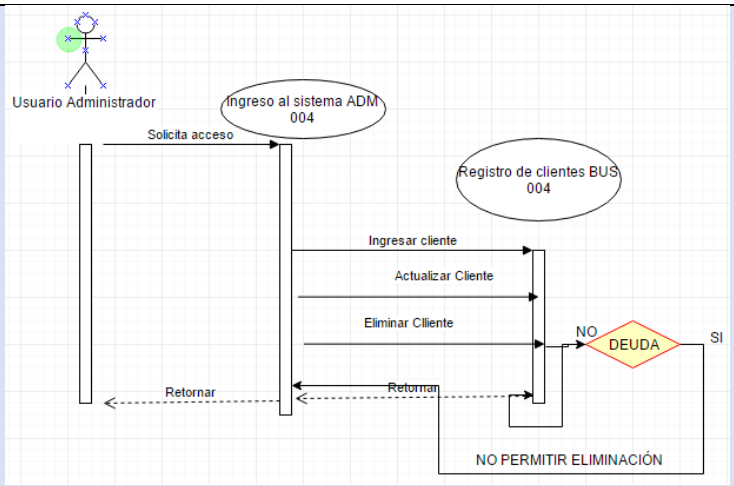


Imagen 33. Mockup pantalla caso de uso BUS-003

3.2.10 Caso de uso: BUS- 004

ID	BUS- 004

Titulo	Administración de clientes del sistema
Descripción	Mediante acceso a una página contemplada dentro del módulo de “base pos” se proporcionará acceso al usuario del sistema para que tenga acceso a una página con la funcionalidad de ingresar, actualizar y eliminar clientes del sistema.
Actores	Usuarios del sistema autorizados.
Precondiciones	Es preciso tener al usuario del sistema logueado en el aplicativo y que pueda visualizar correctamente la pantalla de trabajo. El usuario que utiliza esta página debe ser un usuario existente del sistema y contar con un rol obligatoriamente como el caso ADM-001 lo menciona. En la pantalla principal de este requerimiento funcional se proporcionará al usuario una pantalla para la visualización de clientes del sistema en una tabla paginable por servidor ya que pueden llegar a ser demasiados registrados. Así mismo se proporcionará la funcionalidad de edición para cada registro creado, la funcionalidad de ingresar clientes nuevos al sistema, así como eliminar existentes. Los campos obligatorios para la inserción/actualización de un producto son: • Nombre (alfanumérico) • Cupo Asignado (numérico)
Postcondiciones	Una vez el usuario sea validado para el ingreso al sistema, se visualizará un menú correspondiente al rol que este tiene configurado, mediante el cual en el módulo creado de “Base Pos” se encontrara con la página de clientes.
Escenario Principal	La página se accederá por medio del módulo “base pos” en su vínculo llamado clientes. Una vez se ingresa se deberán ver todos los clientes registrados en el sistema, así como su deuda actual el sistema como su cupo máximo asignado (imagen 34). Desde esta tabla podemos hacer uso de la funcionalidad de saldar total o parcialmente la deuda que el cliente tiene con el mini mercado, dando click en el botón saldar (imagen 35). Se presentará una pantalla informando cual es el valor de la deuda y se le proporcionará al usuario un campo de texto numérico donde podrá ingresar valores numérico-positivos (imagen 36). Una vez se ingrese un valor este deberá descontarse de la deuda vigente con el sistema. Para el ingreso de un cliente nuevo al sistema se deberá proporcionar una

	funcionalidad de ingreso de los campos con su respectiva validación (imagen 37).
Escenario(s) Alternativo(s)	Debe existir la funcionalidad relacionada junto con la historia de usuario BUS-005, donde un cliente podrá ser seleccionado para la venta actual, detallada en la historia de usuario BUS-005
Prioridad	8
Ámbito	Negocio del sistema.
Frecuencia de uso	Media
Ilustración	 *Imagen 29. Ilustración caso de uso BUS-004*
Diagrama de Secuencia	 *Imagen 34. Diagrama de secuencia caso de uso BUS-004*

Mockup

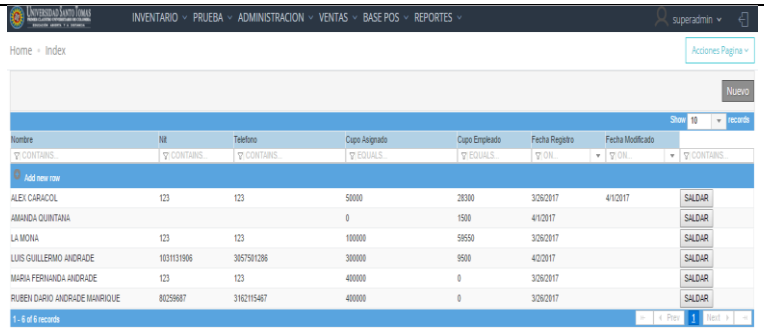


Imagen 35. Mockup pantalla caso de uso BUS-004

Figure 36. Mockup pantalla caso de uso BUS-004

Imagen 37. Mockup pantalla caso de uso BUS-004

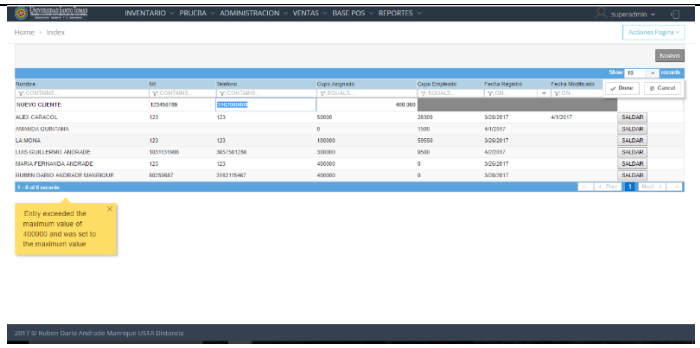
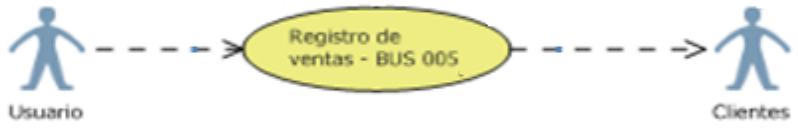
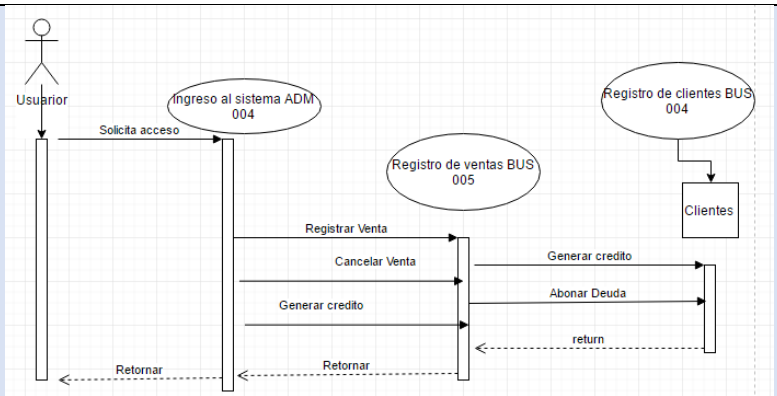


Imagen 38. Mockup pantalla caso de uso BUS-004

3.2.11 Caso de uso: BUS- 005

ID	BUS- 005
Titulo	Registro de ventas
Descripción	Mediante acceso a una página contemplada dentro del módulo de “ventas” se proporcionará acceso al usuario del sistema para que tenga acceso a una página con la funcionalidad de registrar las ventas del sistema.
Actores	Usuarios del sistema autorizados.
Precondiciones	Es preciso tener al usuario del sistema logueado en el aplicativo y que pueda visualizar correctamente la pantalla de trabajo. El usuario que utiliza esta página debe ser un usuario existente del sistema y contar con un rol obligatoriamente como el caso ADM-001 lo menciona. En la pantalla principal de este requerimiento funcional se proporcionará al usuario una pantalla para la visualización de cuatro espacios de ventas para poder cumplir con uno de los objetivos fundamentales de la implementación de este proyecto, la alta usabilidad, y poder realizar ventas paralelas fácilmente. Los campos obligatorios para la realización de una venta son registro de ítems mínimo una unidad.
Postcondiciones	Una vez el usuario sea validado para el ingreso al sistema, se visualizará un menú correspondiente al rol que este tiene configurado, mediante el cual en el módulo creado de “ventas” se encontrará con la página de “realizar ventas”.
Escenario Principal	La funcionalidad se accederá por medio del módulo “Ventas” en su vínculo llamado Realizar Ventas.

	Una vez el usuario ingresa a la página contará con una pantalla que le permitirá realizar hasta cuatro ventas simultáneamente sin ninguna ser incluyente de la otra (imagen 39). En cada casilla de venta se tendrán las siguientes funcionalidades • Realizar venta • Cancelar venta • Botón genérico de ventas especificas • Entrada de texto y de código de barras para la inserción del producto a la venta • Tabla de productos de la venta con la funcionalidad de eliminar, adicionar y restar unidades • Paginación en cliente de la funcionalidad • Selección de cliente quien realiza la venta Para cada casilla se tendrá la posibilidad de registrar esa venta para un cliente en particular seleccionándolo de la sección de clientes (imagen 40). Al seleccionar un cliente y realizar una venta se puede destinar total o parcialmente una parte a crédito como también se puede recibir el efectivo de la venta completamente sin necesidad de crear un registro de crédito sino únicamente el récord asociado a la venta del cliente (imagen 41). En caso de ser un crédito se le da la opción al usuario de poder asignarlo total o parcialmente con la idea de un abono (imagen 42). Así mismo se tendrá un prompt de finalización de la venta el cual se puede aceptar o cancelar (imagen 43). Al ejecutar cualquier venta la pantalla informará del éxito de la operación (imagen 44). Para las entradas de datos o procesos de registro, que generan un movimiento en el inventario o en la caja del sistema, incluso una anotación de crédito a un cliente sea completa o parcial, se notificara vía correo al administrador del sistema. Este correo esta parametrizado en la sección 3.8 “implementación” (imagen 80)
Escenario(s) Alternativo(s)	No aplica. El único escenario para realizar ventas en el sistema es por medio de esta funcionalidad.
Prioridad	10

Ámbito	Negocio del sistema.
Frecuencia de uso	Alta
Ilustración	 *Imagen 30. Ilustración caso de uso BUS-005*
Diagrama de Secuencia	 *Imagen 45. Diagrama de secuencia caso de uso BUS-005.*

Mockup

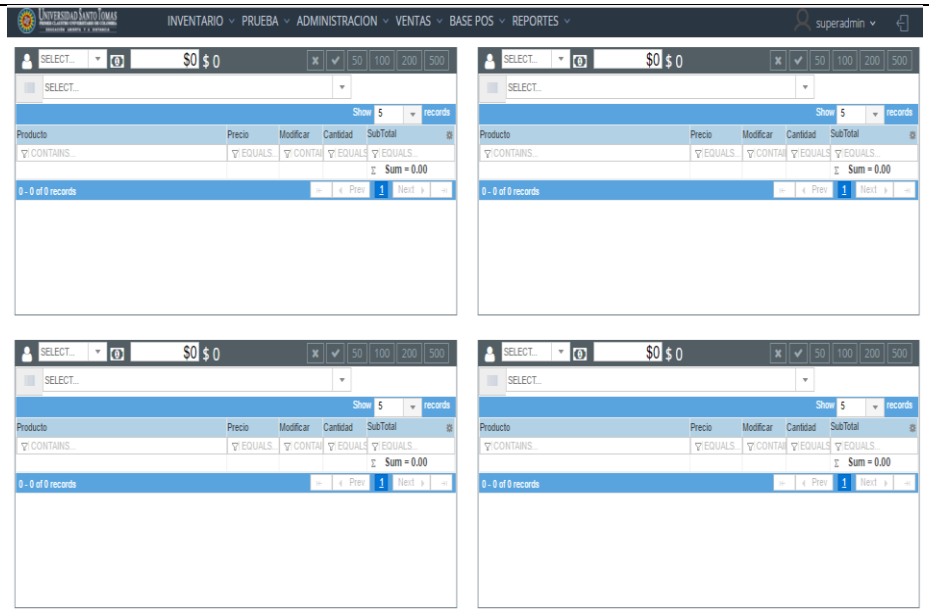


Imagen 39. Mockup pantalla caso de uso BUS-005

Imagen 40. Mockup pantalla caso de uso BUS-005

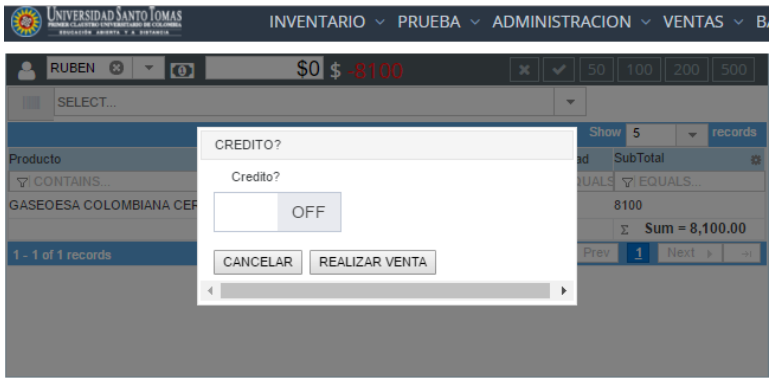


Imagen 410. Mockup pantalla caso de uso BUS-005

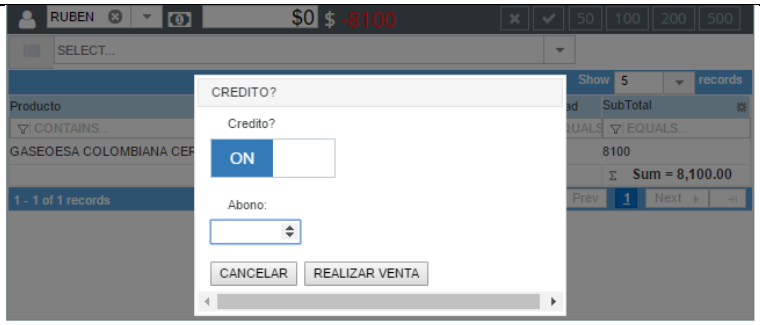


Imagen 42. Mockup pantalla caso de uso BUS-005

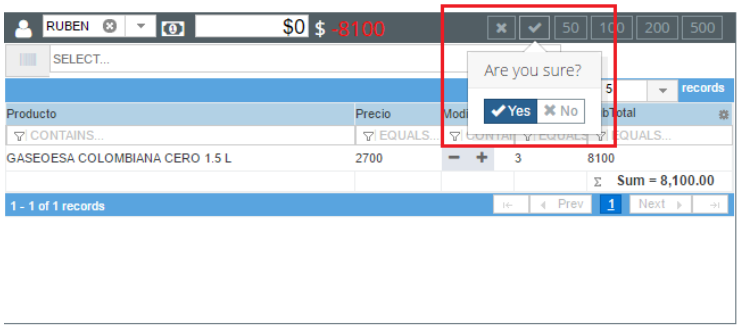


Imagen 43. Mockup pantalla caso de uso BUS-005

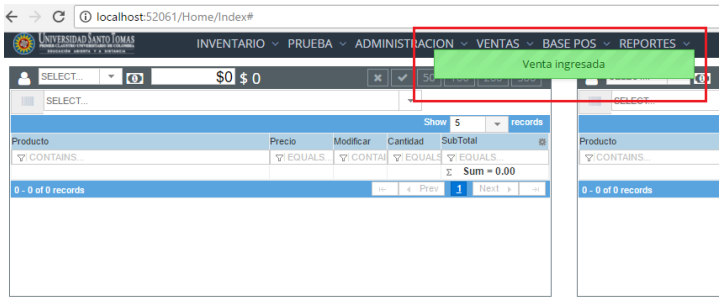


Imagen 44. Mockup pantalla caso de uso BUS-005

3.2.12 Caso de uso: BUS- 006

ID	BUS- 006
Titulo	Programación de pedidos
Descripción	Mediante acceso a una página contemplada dentro del módulo de inventario, se proporcionará acceso al usuario del sistema para que tenga acceso a una página con la funcionalidad de ingresar, actualizar y eliminar pedidos del sistema.

Actores	Usuarios del sistema autorizados.
Precondiciones	Es preciso tener al usuario del sistema logueado en el aplicativo y que pueda visualizar correctamente la pantalla de trabajo. El usuario que utiliza esta página debe ser un usuario existente del sistema y contar con un rol obligatoriamente como el caso ADM-001 lo menciona. En la pantalla principal de este requerimiento funcional se proporcionará al usuario una pantalla para la visualización de los productos faltantes en una tabla paginable por servidor ya que pueden llegar a ser demasiados. Así mismo se proporcionará la funcionalidad de selección de cada uno o la totalidad de estos productos faltantes y asociarlos a un proveedor previamente ingresado en el caso BUS 001.
Postcondiciones	Una vez el usuario sea validado para el ingreso al sistema, se visualizará un menú correspondiente al rol que este tiene configurado, mediante el cual en el módulo creado "inventario", se encontrara con la página de "Productos Faltantes". Al dar click en cualquiera de los botones de funcionalidad se debe seleccionar uno o varios elementos a los que aplicar la acción.
Escenario Principal	La página se accederá por medio del módulo inventario en su vínculo llamado productos faltantes. En esta página se tendrá una tabla con un resumen de las unidades faltantes en el sistema basado en la cantidad actual de ítems y su umbral previamente configurado. Así aparecerán los productos cuantas opciones disponibles tengan de crear el pedido por cada proveedor o en el caso particular que no tengan asociado a ningún proveedor en absoluto (imagen 46). Se puede en esta página realizar la funcionalidad de asociar, desasociar, y programar productos faltantes individualmente o tantos productos se seleccionen en la tabla de visualización. Así mismo se debe contar con la funcionalidad de ver los días de visita y entrega de los proveedores para poder realizar una selección basada en esa información (imagen 47). Una vez tengamos certeza de que productos elegir para el pedido y con qué proveedor los seleccionamos y se pasaran a una segunda tabla que es la de la programación de los pedidos (imagen 48). Y el sistema notificara que el producto ha sido programado para pedir en su visita.
Escenario(s) Alternativo(s)	Por parte de los usuarios vendedores del sistema, habrá una funcionalidad relacionada con la confirmación del pedido con el proveedor y el acuse de recibido del pedido. Una vez un pedido ha sido programado para realizarse, este se pondrá a disposición para que el usuario vendedor del sistema se lo comunique al

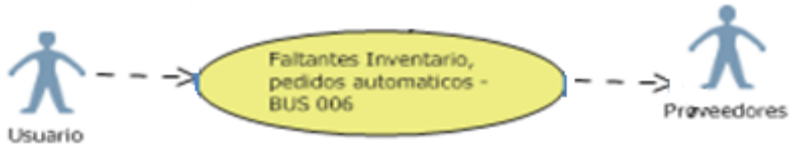
	vendedor visitante. Para poder acceder a la funcionalidad el usuario del sistema podrá dar click en la parte superior derecha donde se le desplegará una venta lateral donde hay dos secciones indicando las visitas y los pedidos. Para que salgan los pedidos deben aprobarse las visitas (imagen 49a), estos son los productos que el administrador del sistema previamente programo para que el usuario vendedor se los comunique al proveedor. En este punto, el proveedor podrá validar la disposición del producto o la inexistencia, por lo que el usuario vendedor, chequeará los ítems confirmados (imagen 49b) y dará click en el botón de confirmar. Una vez este proceso ha sido realizado, se le enviara una notificación vía correo electrónico al administrador del sistema confirmando que productos han sido validados por el proveedor y cuales no (imagen 49c). Una vez este flujo se ha completado se dispondrá de la funcionalidad de acuse de recibido del pedido para el proveedor (imagen 49d), donde el usuario vendedor, podrá chequear lo que se ha pedido por el proveedor y diligenciara el valor del pedido que afectara los reportes de movimientos del sistema e informara al administrador del sistema de la operación (imagen 49e).
Prioridad	11
Ámbito	Negocio del sistema.
Frecuencia de uso	Alta
Ilustración	 *Imagen 31. Ilustración caso de uso BUS-006.*

Diagrama de Secuencia

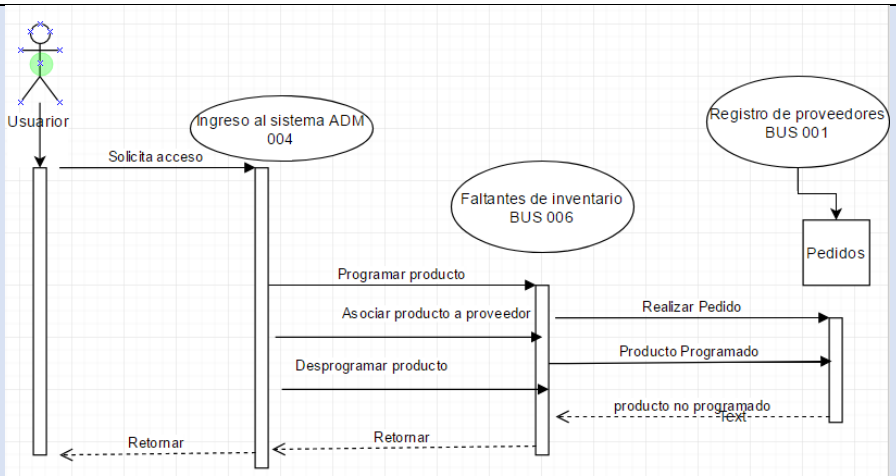


Imagen 49. Diagrama de secuencia caso de uso BUS-006.

Mockup

Imagen 46. Mockup pantalla caso de uso BUS-006.

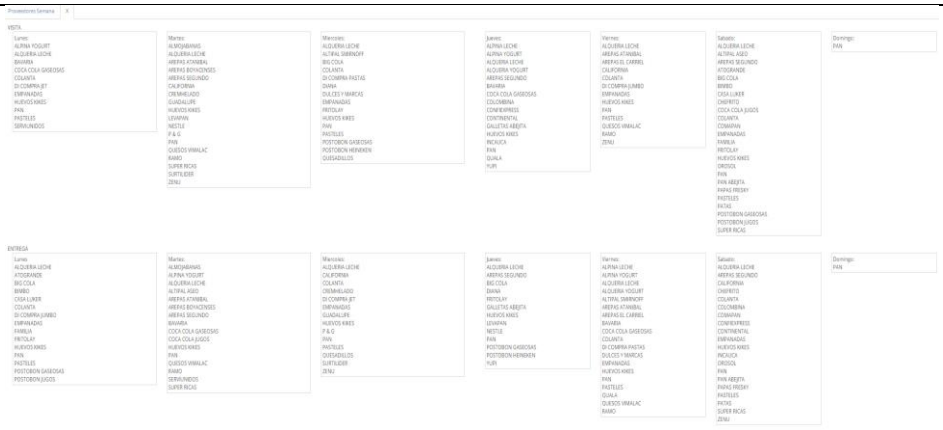


Imagen 47. Mockup pantalla caso de uso BUS-006.

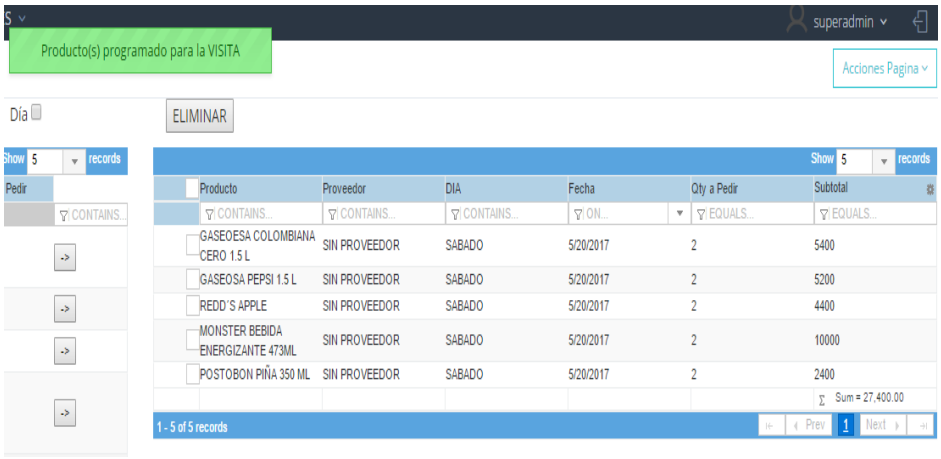


Imagen 48. Mockup pantalla caso de uso BUS-006.

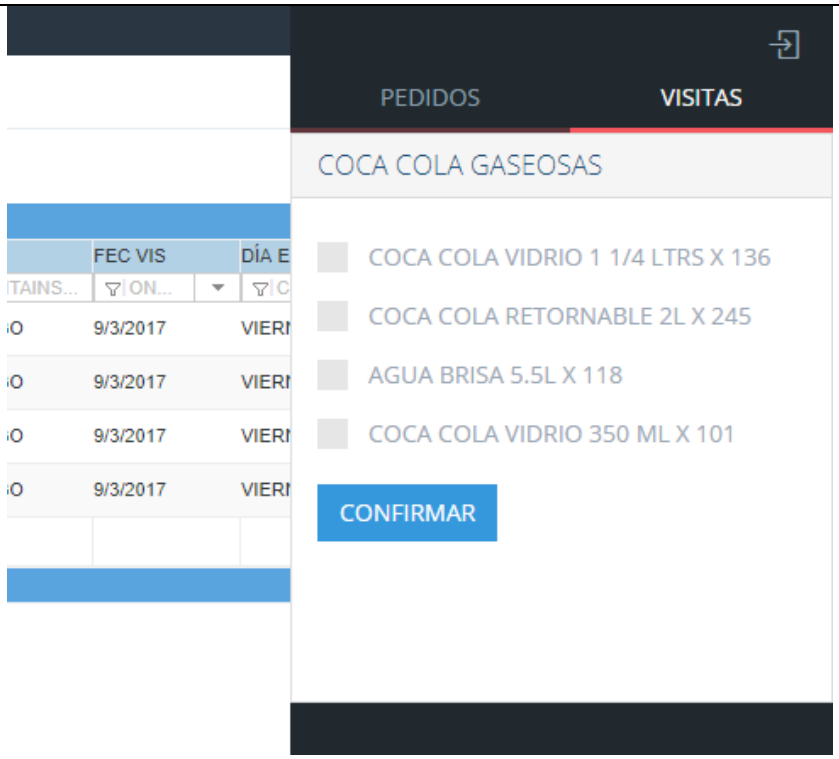


Imagen 49a. Mockup pantalla caso de uso BUS-006.

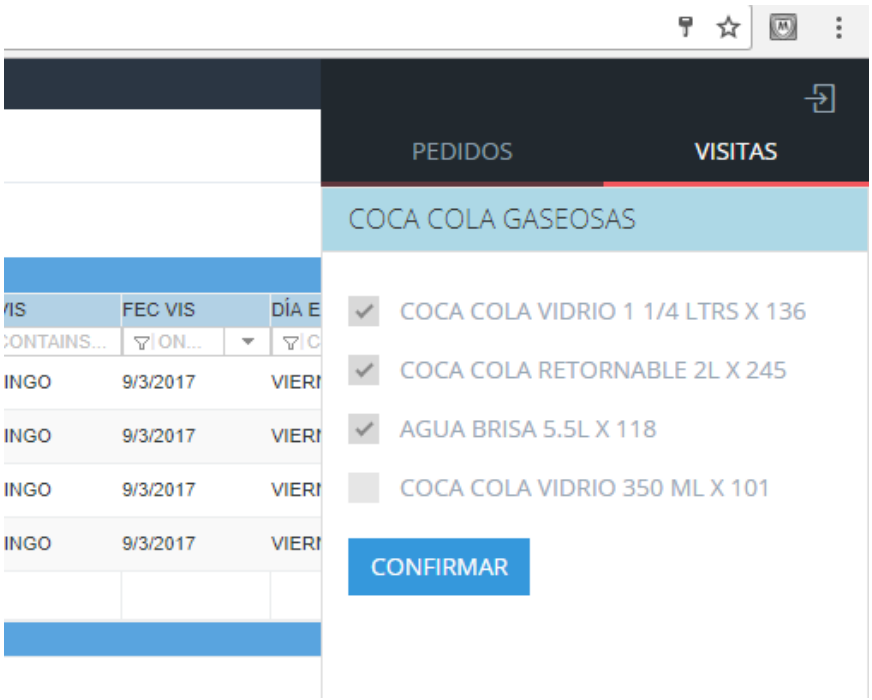


Imagen 49b. Mockup pantalla caso de uso BUS-006.

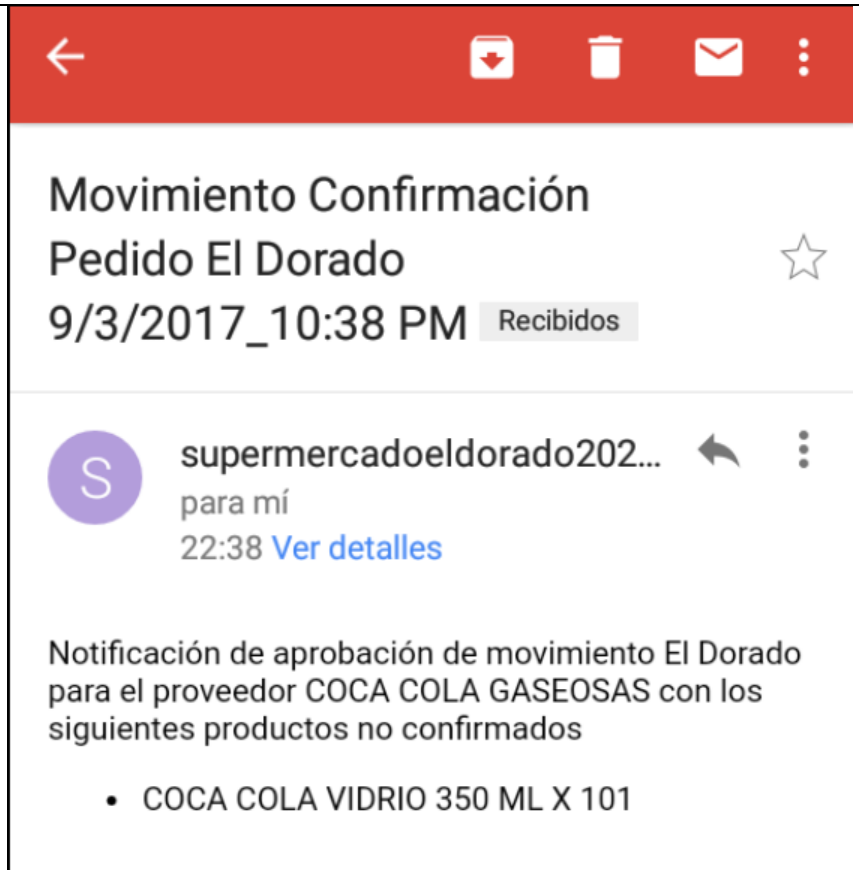


Imagen 49c. Mockup pantalla caso de uso BUS-006.

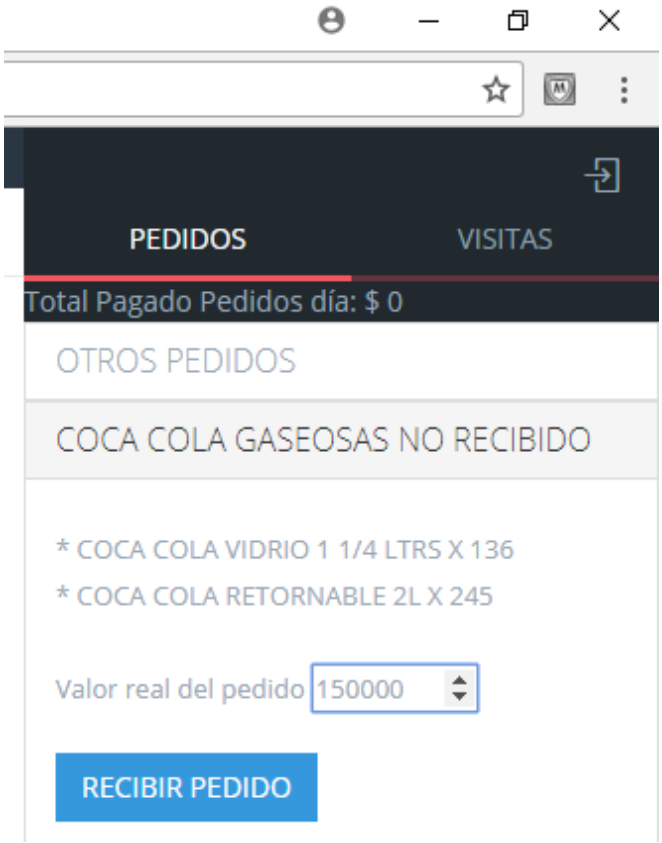


Imagen 49d. Mockup pantalla caso de uso BUS-006.

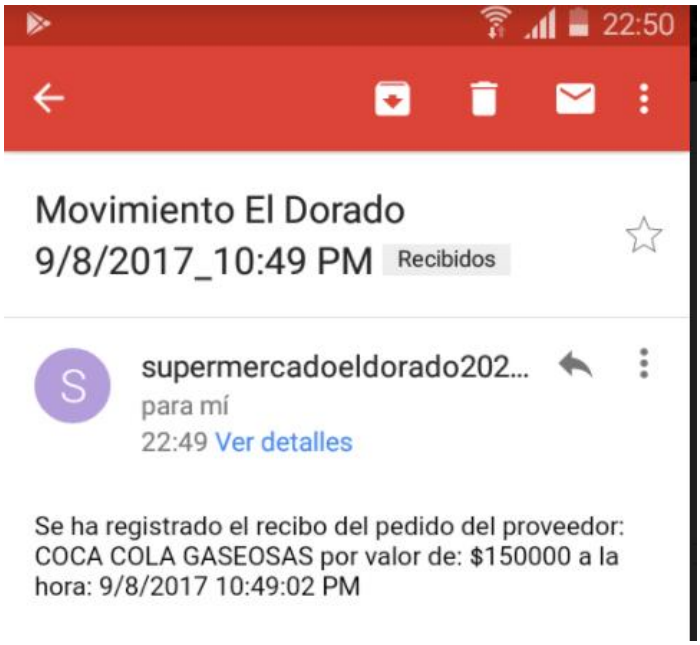
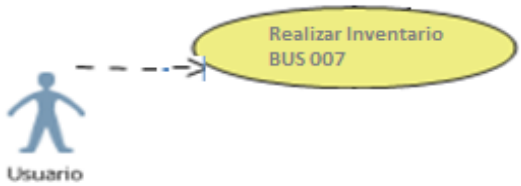
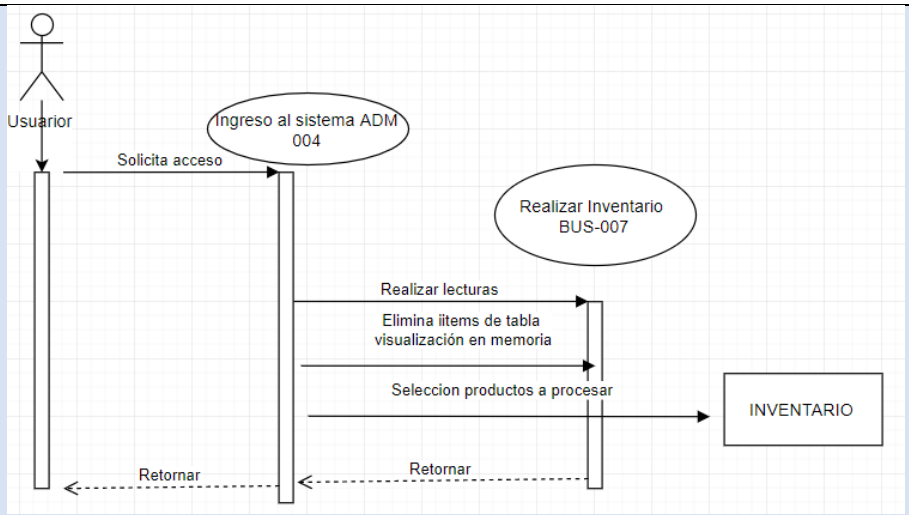


Imagen 49e. Mockup pantalla caso de uso BUS-006.

3.2.13 Caso de uso: BUS- 007

ID	BUS- 007
Titulo	Realizar inventario
Descripción	Mediante el acceso en el vínculo “realizar inventario” del módulo “inventario” del sistema, se proporcionará la funcionalidad para poder llevar a cabo un inventario por ubicaciones definidas del negocio.
Actores	Usuarios del sistema autorizados.
Precondiciones	Es preciso tener al usuario del sistema logueado en el aplicativo y que pueda visualizar la opción correctamente en el menú. El usuario que utiliza esta página debe ser un usuario existente del sistema y contar con un rol, como el caso ADM-001 lo menciona. La funcionalidad se basa en el conteo físico de las unidades actuales, para actualizarlas en el sistema. Los productos deben estar ingresados en el sistema como se indica en el caso de uso BUS-003.
Postcondiciones	Una vez realizado el ingreso a la funcionalidad por el usuario autorizado y luego de seleccionada la ubicación física del negocio a la que se está realizando el inventario, los productos leídos y registrados quedaran actualizadas sus cantidades en el sistema.
Escenario Principal	La funcionalidad se accederá por medio del vínculo “realizar inventario” del módulo “inventario” (imagen 49s). Una vez se ingresa, se presentará la pantalla (imagen 49t), con un selector de las ubicaciones existentes en el sistema, a lo que se deberá seleccionar una opción, y una vez seleccionada, se visualizará un campo de texto el cual servirá de punto de entrada para las lecturas realizadas bien sea por código de barras o mediante el texto del producto y su selección (imagen 49u). Una vez el producto ha sido leído o ingresado por selección del texto, este se cargará en la tabla de visualización (imagen 49w). Para el caso de las lecturas realizadas por código de barras, el sistema permitirá lecturas continuas sobre los productos de la ubicación en inventario, por lo que el sistema estará en capacidad de cargar las unidades correspondientes a las lecturas que vaya realizando y sumar las unidades encontradas al producto (imagen 49x). Una vez se han terminado de realizar las lecturas correspondientes, el usuario tendrá la opción de revisar o modificar las cantidades de un producto específico, por motivos de error en lectura o facilidad de digitación y podrá procesar los registros que desee seleccionándolos y dando click en el botón “procesar” (imagen 49y). En este paso del proceso el sistema pedirá la autorización del usuario, advirtiendo que los productos seleccionados y a procesar serán objeto de actualizaciones en el inventario (imagen 49z.), y en caso de aceptación por parte del usuario, el sistema notificara al usuario de la actualización del inventario (imagen

	49aa).
Escenario(s) Alternativo(s)	El usuario podrá remover productos que han sido registrados o leídos y que se encuentran en la tabla de visualización de productos leídos como proceso del inventario y que han sido objeto de lecturas erróneas o digitación inexacta. Para poder realizar esta acción, el usuario seleccionará los productos que desea descartar (imagen 49ab) y con la aceptación final del usuario (imagen 49ac), la tabla de visualización ya no tendrá los elementos eliminados.
Prioridad	13
Ámbito	Negocio del sistema.
Frecuencia de uso	Baja
Ilustración	 *Imagen 49ad. Ilustración caso de uso BUS-007.*
Diagrama de Secuencia	 *Imagen 49ae. Diagrama de secuencia caso de uso BUS-007.*

Mockup

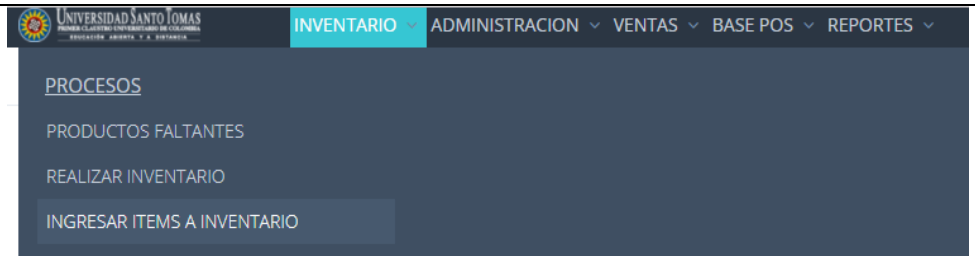


Imagen 49s. Mockup caso de uso BUS-007.

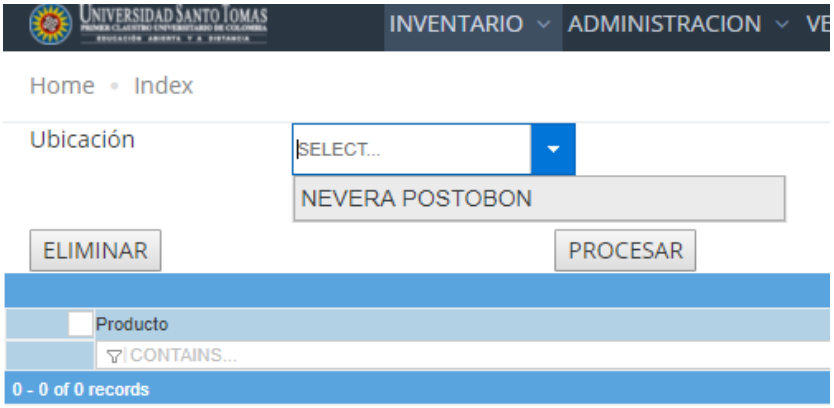


Imagen 49t. Mockup caso de uso BUS-007.

Imagen 49u. Mockup caso de uso BUS-007.

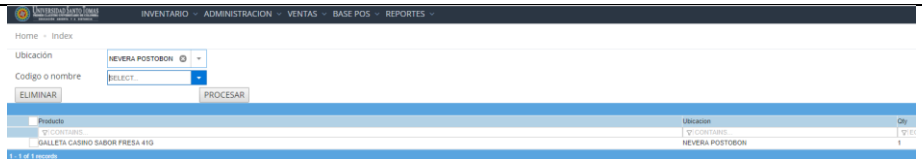


Imagen 49w. Mockup caso de uso BUS-007.

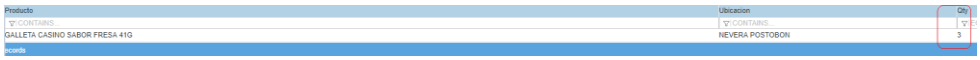


Imagen 49x. Mockup caso de uso BUS-007.

Imagen 49y. Mockup caso de uso BUS-007.

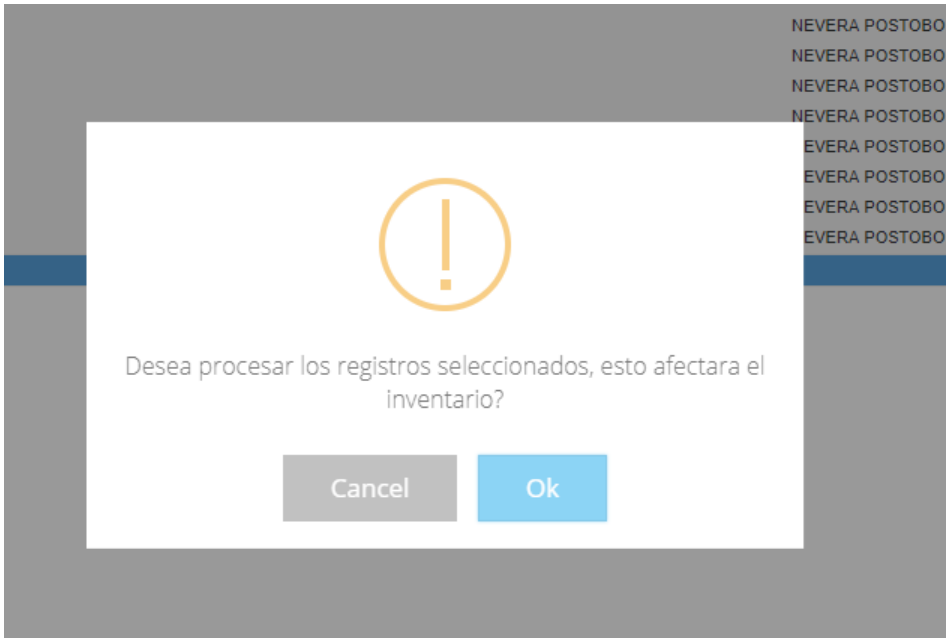


Imagen 49z. Mockup caso de uso BUS-007.

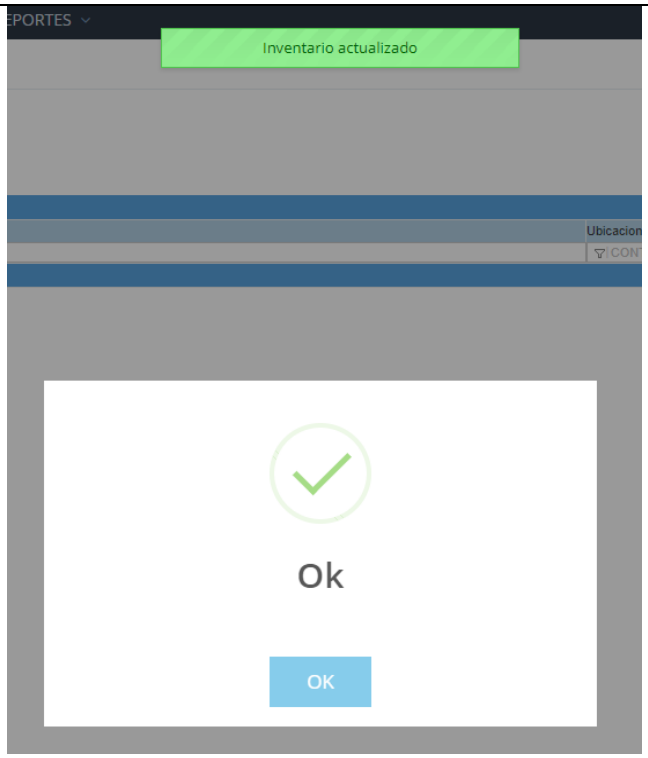


Imagen 49aa. Mockup caso de uso BUS-007.

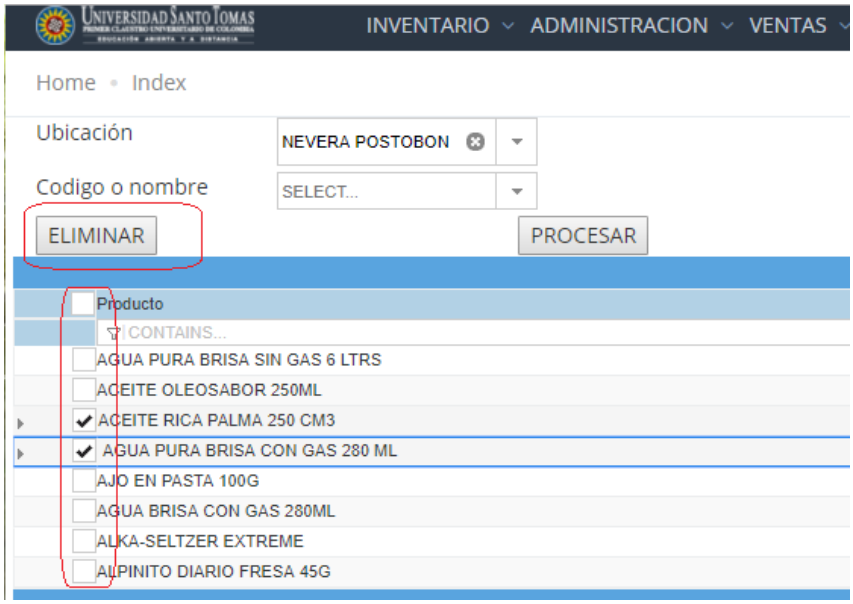


Imagen 49ab. Mockup caso de uso BUS-007.

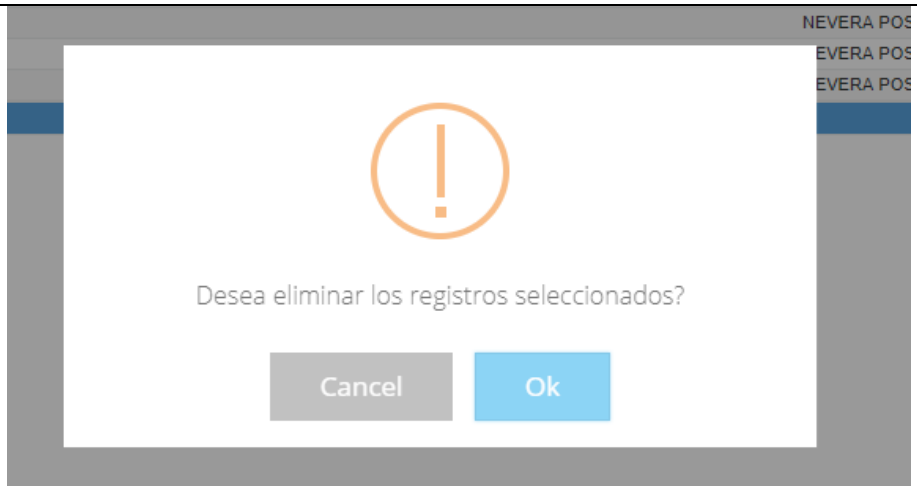


Imagen 49ac. Mockup caso de uso BUS-007.

3.2.14 Caso de uso: BUS- 008

ID	BUS- 008
Titulo	Ingresar ítems al inventario
Descripción	Mediante el acceso en el vínculo "ingresar ítems a inventario" del módulo de inventario del sistema, se proporcionará la funcionalidad de ingresar los ítems entregados por el proveedor y así mantener el inventario actualizado y con precios acorde a las actualizaciones de precios.
Actores	Usuarios del sistema autorizados.
Precondiciones	Es preciso tener al usuario del sistema logueado en el aplicativo y que pueda visualizar la opción correctamente en el menú. El usuario que utiliza esta página debe ser un usuario existente del sistema y contar con un rol, como el caso ADM-001 lo menciona. La funcionalidad se basa en el ingreso de ítems por proveedor, por lo que deben existir en el sistema los proveedores (caso de uso BUS-001).
Postcondiciones	Una vez realizado el ingreso a la funcionalidad por el usuario autorizado, se realice la selección del proveedor y el ingreso de los ítems con sus cantidades y costos específicos, los productos en cuestión quedaran actualizados en el inventario del sistema.
Escenario Principal	La funcionalidad se accederá por medio del módulo "inventario" en el vínculo "ingresar ítems a inventario" (imagen 49h). Una vez en la página, se nos presentará un selector con todos los proveedores disponibles, del cual procederemos a seleccionar al cual se le realizar el ingreso de ítems (imagen

	49i). Una vez se seleccione al proveedor al cual se le realizará el ingreso de ítems producto de un pedido, se mostrará la tabla donde se ingresarán los ítems (imagen 49j). Para el ingreso de cada uno de los ítems se contará con una caja de texto que ofrece la doble funcionalidad altamente usable, de poder ingresar los productos tanto por su código de barras, como por su nombre. Una vez ingresado el ítem se puede visualizar en la tabla (imagen 49l) donde nos aparecerá información del producto a ingresar con los siguientes campos: • Nombre producto. • Cantidad de unidades pedidas. • Cantidades actuales del producto. • % Porcentaje de ganancia actual. • Valor del producto contemplando ganancia del 20%. • Valor del producto contemplando ganancia del 25%. • Valor del producto contemplando ganancia del 30% • Cantidad por ingresar (modificable). • Precio de venta (modificable). • Costo del ítem (modificable). El campo “cantidad por ingresar” se refiere al valor de unidades por ingresar al sistema del producto. El campo “costo del ítem” hace referencia al valor unitario que le cuesta al establecimiento adquirir el producto. Una vez se ingresa este valor, y teniendo en cuenta el valor del campo “precio de venta” se actualizarán los campos % de ganancia actual, valor del producto contemplando ganancia al 20%, valor del producto contemplando ganancia al 25% y valor del producto contemplando ganancia al 30%. Esto con el ánimo de ofrecer una visión del margen de ganancias actual del producto, así como de ofrecer valores finales de venta que deberían dejarse al producto, contemplando un margen de ganancias en los valores estipulados de 20%, 25% y 30% (imagen 49m). Adicionalmente, si actualizamos el valor del campo “costo del producto” o el valor del campo “precio de venta”, sé actualizaran los valores de los campos: % ganancia actual, valor del producto contemplando ganancia del 20%, valor del producto contemplando ganancia del 25%, valor del producto contemplando ganancia del 30%. Esto con el ánimo de ayudar al usuario del sistema a configurar el precio del producto dentro de un margen que le genere rentabilidad sin tener un precio de venta elevado. De igual manera, existirá una advertencia visual cuando el costo del producto, junto con el valor de venta del producto, genere un margen de ganancia menor al 20%. Para este

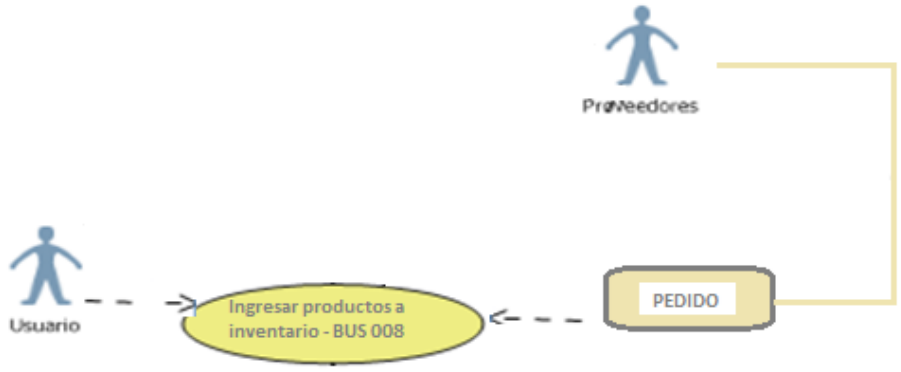
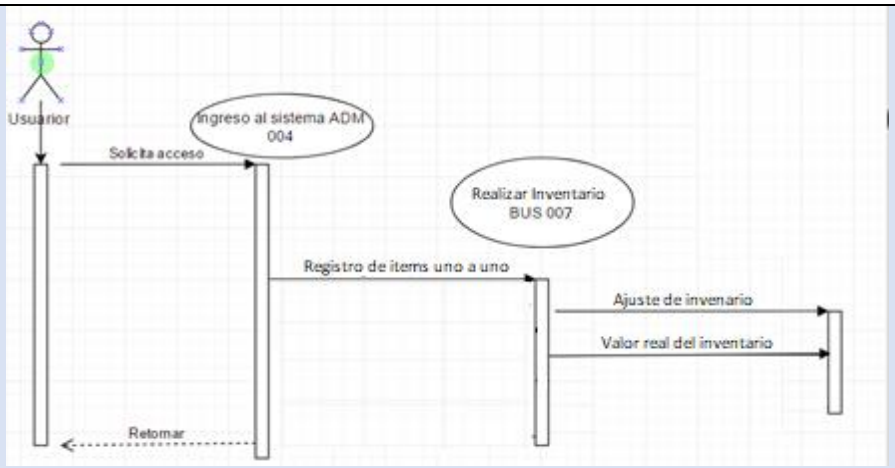
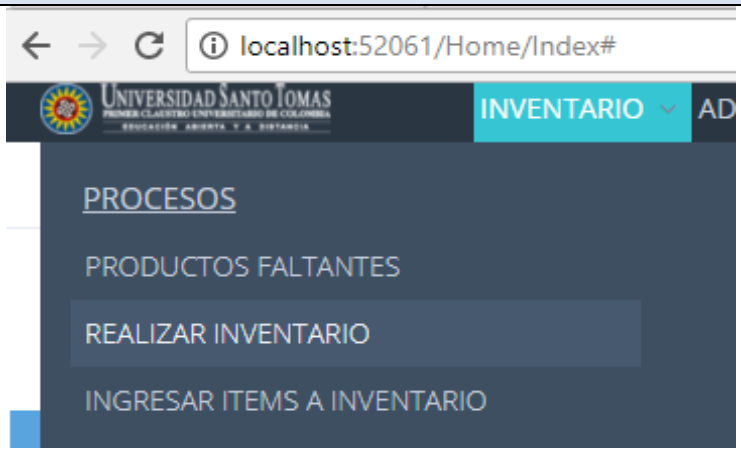
	caso el campo “% ganancia actual” se pondrá de color rojo (imagen 49m). Una vez ingresados productos a la tabla, todos estos valores, están siendo actualizados en memoria y aún no son persistidos en la base de datos del sistema. Esto con el ánimo de que el usuario valide la información y seleccione los productos que desea procesar. Para este proceso final, el deberá seleccionar los productos que desee o en su defecto todos, para que sean procesados y cargados al inventario, dando click en el botón procesar (imagen 49n), a lo que el sistema realizara una última solicitud de aceptación por parte del usuario (imagen 49o), y en caso de seleccionar “ok” el sistema actualizara los siguientes valores del producto: la cantidad de ítems, el precio de venta, el costo de compra por parte del proveedor, se informara al usuario del proceso exitoso y se finalizara el proceso (imagen 49p).
Escenario(s) Alternativo(s)	Es probable que en el paso relacionado con la imagen 49k, donde se lee el código de barras del producto a ingresar o se escribe su nombre para su selección, se comentan errores por parte del usuario. Por lo que el usuario tendrá la funcionalidad de remover de la tabla el producto, seleccionando la caja de texto correspondiente y dando click en el botón “descartar” (imagen 49q), a lo que el sistema una vez más realizara una última solicitud de aceptación por parte del usuario (imagen 49r) y en caso de aceptación, el producto o los productos seleccionados serán borrados de la taba de visualización.
Prioridad	12
Ámbito	Negocio del sistema.
Frecuencia de uso	Alta
Ilustración	 *Imagen 49f. Ilustración caso de uso BUS-008.*

Diagrama de Secuencia	 *Imagen 49g. Diagrama de secuencia caso de uso BUS-008.*
Mockup	 *Imagen 49h. Mockup caso de uso BUS-008.*

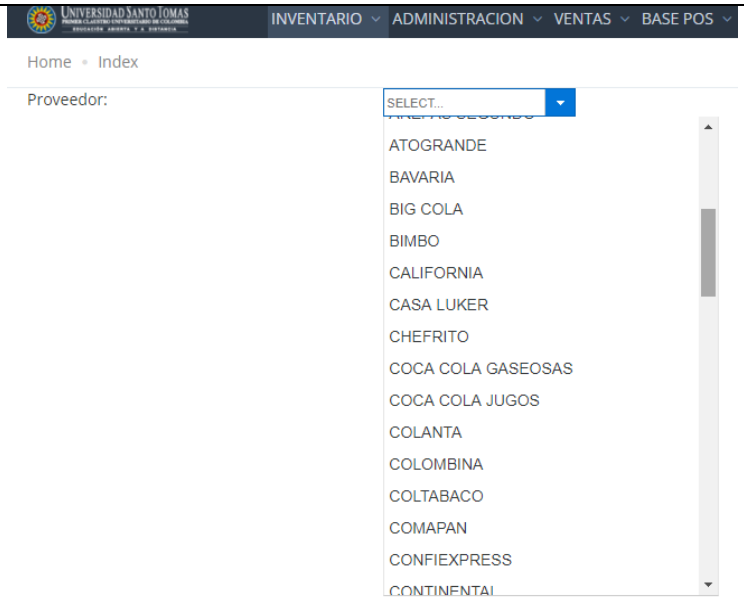


Imagen 49i. Mockup caso de uso BUS-008.

Imagen 49j. Mockup caso de uso BUS-008.

Imagen 49k. Mockup caso de uso BUS-008.

Imagen 49l. Mockup caso de uso BUS-008.

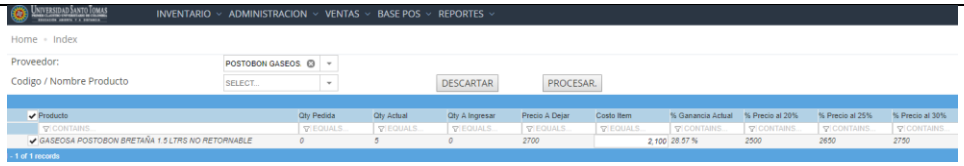


Imagen 49m. Mockup caso de uso BUS-008.

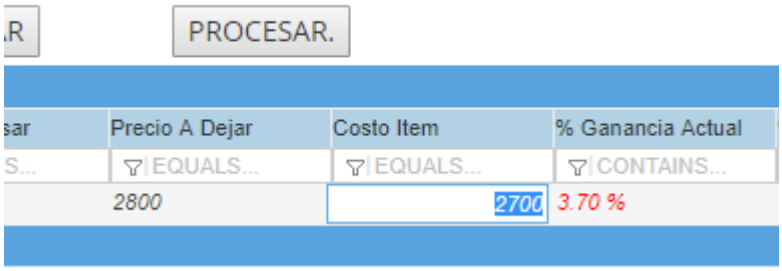


Imagen 49m. Mockup caso de uso BUS-008.

Imagen 49n. Mockup caso de uso BUS-008.

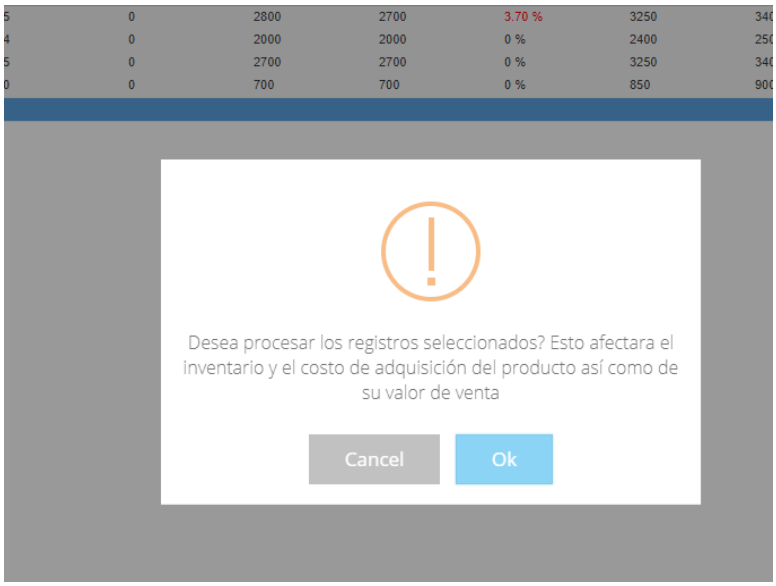


Imagen 49o. Mockup caso de uso BUS-008.

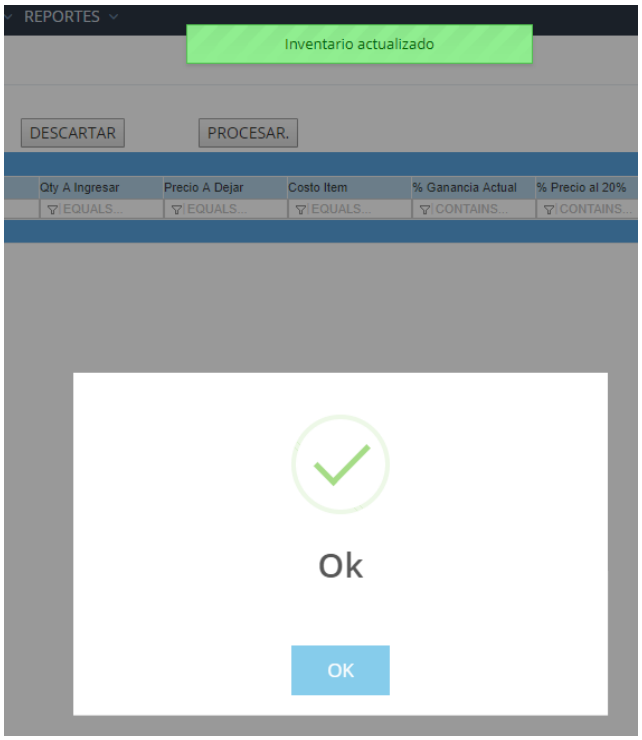


Imagen 49p. Mockup caso de uso BUS-008.

Imagen 49q. Mockup caso de uso BUS-008.

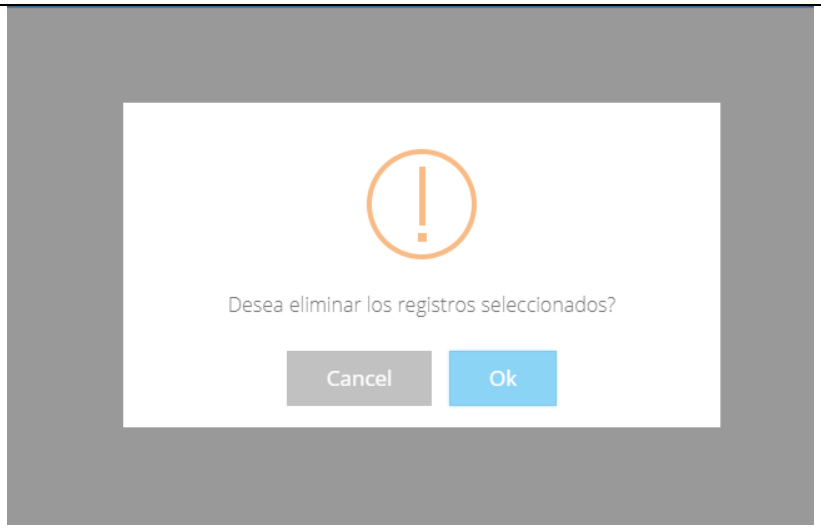


Imagen 49r. Mockup caso de uso BUS-008.

3.3 Modelo entidad relación

El elemento central del modelo relacional es la relación. Una relación tiene un nombre, un conjunto de atributos que representan sus propiedades y un conjunto de tuplas de incluyen los valores que cada uno de los atributos toma para cada elemento de la relación. (Cuadra, y otros, 2008, pág. 183)

Se ha realizado la segmentación del modelo relacional en dos bloques distintivos que son: el administrativo y el de negocio. El modelo administrativo hace referencia a todas las tablas donde se persistirán las entidades relacionadas con la seguridad, usuarios, paginas, tipos de páginas, roles, tipos de páginas e idiomas del sistema (imagen 50). El modelo de negocio hace referencia a todas las tablas correspondientes a las entidades del negocio como son: clientes, productos, proveedores, ubicaciones, ventas entre otros (imagen 51).

Se ha decidido realizar esta segmentación con la creación de los esquemas ADM (Administrativo) y BUS (Negocio), con el ánimo de separar o dar una similitud entre distintos

objetos de la base de datos, asegurar su mantenibilidad, promover una ubicación para cada objeto y tratar de que no se duplique funcionalidad en diferentes partes. Adicional a esto, la mayoría de las definiciones de cada una de las tablas, contienen los campos: IdUsuarioRegistro (int), FechaRegistro (DateTime), IdUsuarioModificador (int, null), FechaModificado (Datetime, null), con el objetivo de tener una trazabilidad en el registro de la base de datos del sistema.

3.3.1 Modelo entidad relación administración del sistema

El modelamiento de la administración del sistema es un único conjunto de tablas que funcionan en conjunto para garantizar la accesibilidad, autorización, seguridad, navegabilidad y registro de usuarios, para el funcionamiento del sistema.

En la siguiente tabla se describe cada uno de los elementos de este modelo junto con una descripción.

Tabla 5. Elementos base de datos, administración del sistema

Tabla	**Descripción**
OpcionesProyectoUsta	Son las opciones o vínculos de acceso del sistema, contemplando también contener registros que actúen a su vez como contenedores de los vínculos o módulos para el caso específico de este proyecto. Este tipo de modelamiento permite en una misma tabla, proporcionar una jerarquía de modulo, vinculo, que se ajusta a las necesidades de navegación de este sistema. Tiene campos como el “orden”, que permite parametrizar el orden de renderizado de la opción.
TipoOpcionProyectoUsta	Es una clasificación que se le da a la opción de tipo vínculo para organizar su renderizado.
OpcionesUsuarioProyectoUsta	Son las opciones específicas a las que un usuario tiene permiso y, por consiguiente, serán renderizadas para cuando acceda al sistema.

OpcionesRolProyectoUsta	Son las opciones específicas a las que un rol tiene permiso de visualización dentro del sistema. Se tiene esta opción para poder tener una parametrización de entrada y navegación general, ya que pueden muchos usuarios tener un mismo rol, por ende, la configuración a la que se accederá en ese caso particular será la misma, evitando replicación de configuraciones innecesarias.
RolesProyectoUsta	Son los roles que existen dentro del sistema. Se cuenta con los campos de nombre y estado, ya que un rol puede darse de baja o desactivarse.
Webpages_membership	Es un recurso de utilidad propia del api de seguridad ASP.NET Membership de Microsoft, que se utiliza en conjunto con un motor de base de datos relacional. Se encarga de toda la parte de manejo de contraseñas, registro de accesos, nivel de contraseñas, hash de contraseñas y demás componentes de seguridad relacionados con una aplicación persistida en base de datos.
IdiomasProyectoUsta	Es la tabla en la que se guardan el idioma del sistema, que por defecto es español.
UsuariosRolesProyectoUsta	Es la tabla donde se relacionan los roles con los usuarios del sistema.
UsuarioProyectoUsta	Es la tabla donde se guarda toda la información relacionada con los usuarios del sistema, entre estos se encuentran: administradores, vendedores y demás roles que se necesiten. Adicional en esta tabla se pueden almacenar los datos del usuario como correo, email de contacto y teléfono.

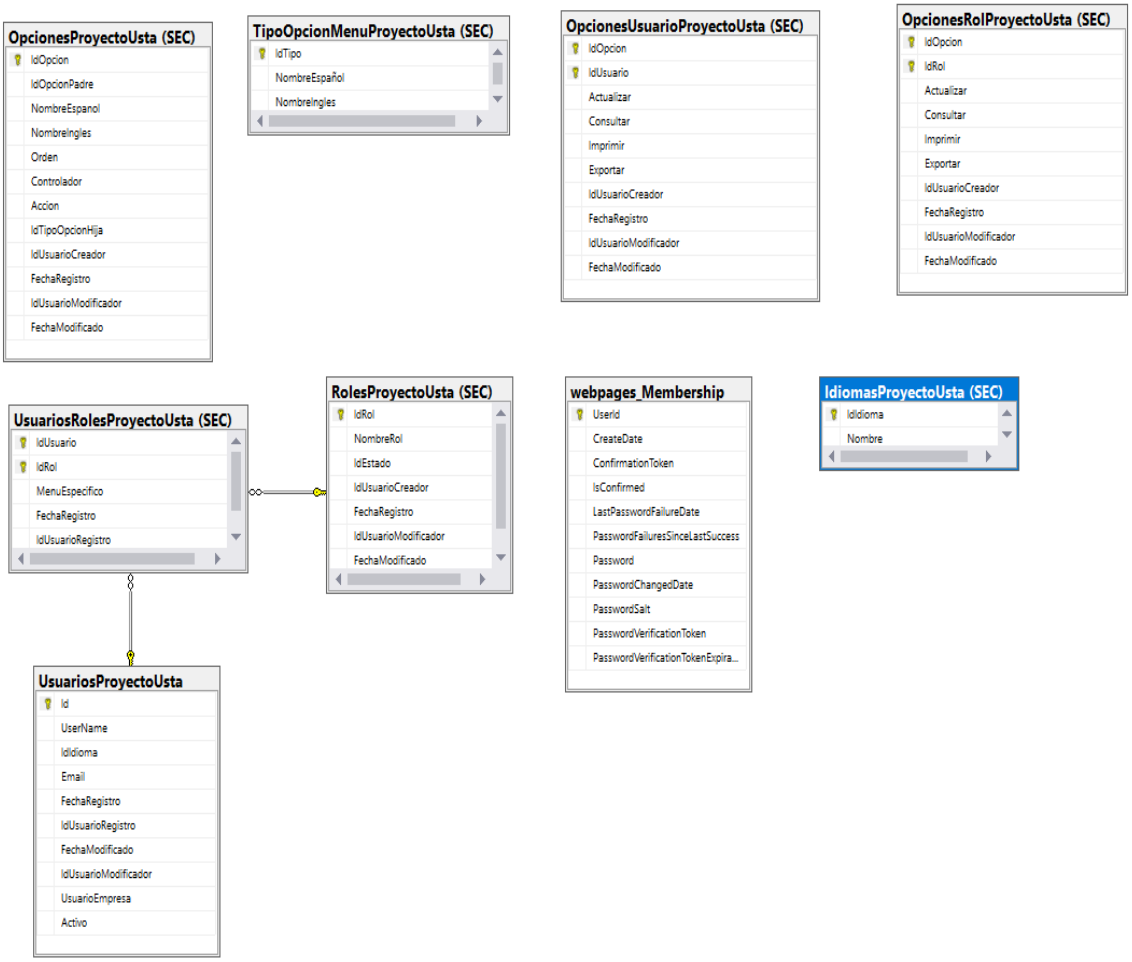


Imagen 50. Modelo entidad relación administración del sistema

3.3.2 Modelo entidad relación negocio del sistema

El modelamiento del negocio del sistema se compone de diferentes grupos de tablas las cuales están relacionadas por funcionalidad del sistema y se definen como: inventario, ubicaciones, caja, automatización de pedidos y clientes. En la siguiente tabla se muestra cada uno de los elementos de este modelo, junto con la descripción.

Tabla 6. Elementos base de datos, negocio sistema.

Grupo	Tabla	Descripción

INVENTARIO	ProductosProyectoUsta	Tabla donde se persistirá cada uno de los productos del sistema, será el repositorio principal para obtener cantidades actuales y umbrales del producto, así como su precio de venta.
	ProductosCodigoDeBarrasProyectoUsta	Tabla donde se persistirán cada uno de los códigos de barras del sistema, junto con el identificador del producto al que se asocia.
	ProveedoresProyectousta	Tabla donde se persistirán cada uno de los proveedores registrados, en el sistema. Cuenta con los campos destinados para guardar los días en que el proveedor visita el establecimiento y los días que realiza entrega de pedidos.
	ProductoProveedorProyectoUsta	Tabla donde se persistirán cada una de las relaciones existentes entre un producto del sistema y el proveedor que lo distribuye. En este modelo hay un campo "PrecioCompra" que es de bastante utilidad para el proceso de programación de los pedidos y que tiene como objetivo cumplir con el objetivo de brindar un sistema automatizado, ya que, con este valor, se pueden tomar decisiones de tiempo vs costo, donde se puede realizar la programación de un pedido basado en los faltantes o en el costo de obtener el ítem en un espacio de tiempo diferente entre proveedores.
CAJA	VentasProyectoUsta	Es el repositorio principal de todos los movimientos de ventas del sistema. En el quedaran persistidos el cliente, el usuario vendedor del sistema, el producto, la cantidad de productos vendidos, el total de la línea de venta, así como la fecha en la que se realizó el movimiento. Atendiendo al objetivo de ventas paralelas, este modelo de persistencia es general para cualquier tipo de proceso de venta, por ende, el objetivo de la alta usabilidad está desarrollado en la capa de presentación.
	MovimientosCajaProyectousta	Adicionalmente existe otra tabla donde se agrupa cada uno de los movimientos del sistema los cuales están representados en la tabla

<table>
<tr><td></td><td></td><td>TiposMovimientoProyectoUsta y que se describirá más adelante. El objetivo de este tipo de modelamiento en la capa de datos es la de separar los registros relacionados con la venta, con registros relacionados con movimientos de pagos, salidas de dinero y demás movimientos del sistema.</td></tr>
<tr><td></td><td>LogCreditosProyectoUsta</td><td>El modelo de esta tabla hace referencia a todos los registros de crédito del sistema, con el ánimo de tener un log, de que valor y que fecha, vendido o dado en crédito por un usuario vendedor del sistema. Es acumulativo y sirve de referencia para el cliente que realiza un crédito con el establecimiento EL DORADO.</td></tr>
<tr><td></td><td>TipoMovimientoCajaProyectoUsta</td><td>Es la tabla con valores constantes que definen que tipos de movimiento existentes hay en el sistema. Los datos que contiene son:
<table><tr><th>Id</th><th>Nombre</th></tr><tr><td>1</td><td>INGRESO VENTA</td></tr><tr><td>2</td><td>INGRESO ABONO CREDITO</td></tr><tr><td>3</td><td>INGRESO INVERSIÓN</td></tr><tr><td>4</td><td>EGRESO PAGOS SERVICIOS-ARRIENDOS</td></tr><tr><td>5</td><td>EGRESO CONSIGNACIÓN</td></tr><tr><td>6</td><td>EGRESO URGENCIA</td></tr><tr><td>7</td><td>EGRESO PAGO EMPLEADOS</td></tr><tr><td>8</td><td>EGRESO PAGO PEDIDOS</td></tr></table>
Imagen 50a. Tipos de movimiento de caja del sistema</td></tr>
</table>

CLIENTES	ClienteProyectoUsta	Tabla donde se persistirán cada uno de los clientes del sistema. Se tienen campos como el "email" donde se guardará el correo electrónico del cliente para notificaciones de créditos y ventas, haciendo énfasis en el objetivo de conectividad del sistema. Así mismo habrá campos como el límite de crédito a que el cliente tiene capacidad, explicado en la historia de usuario BUS-004, así como el valor adeudado por el cliente.
PEDIDOS	ProductosAPedirProyectoUsta	Tabla que hace referencia a todos los productos faltantes que se programan para ser pedidos a los proveedores. Es una tabla de información volátil, lo que quiere decir, que una vez ha finalizado el proceso de programación, pago, recibo y cargue de esos elementos al inventario, los datos allí contenidos se eliminaran.

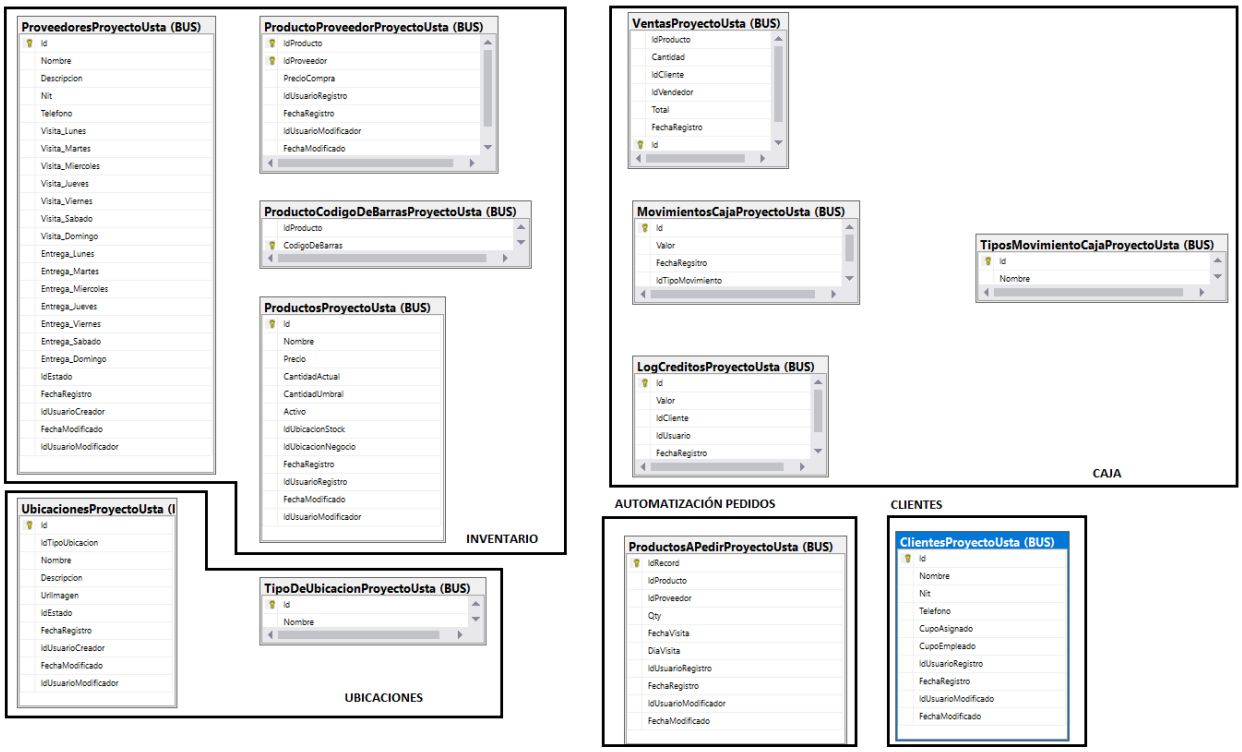


Imagen 51. Modelo entidad relación, negocio del sistema.

3.4 Modelo de clases

El modelo de clases hace referencia las entidades del negocio, en gran parte correspondientes a tablas del modelo entidad relación, como también a entidades propias de la aplicación, correspondientes a un lenguaje de programación que utiliza el paradigma de programación orientada a objetos.

Son la estructura base de operación en cuanto a codificación se refiere. Con este modelo se pretenden solucionar varias problemáticas técnicas como la relación estructurada entre el lenguaje de programación de código servidor, en este caso C# específicamente corriendo en un servidor web utilizando el framework de trabajo ASP.NET, con la persistencia de base de datos, para este caso SQL Server. De igual manera un modelo de estos pretende ser ubicuo en cuanto a las estructuras usadas dentro del lenguaje con el ánimo generar estructuras altamente reutilizables a nivel de código y un orden arquitectónico a lo que se refiere con la instanciación de objetos y demás temas de ejecución en memoria.

Se utilizará una herramienta denominada ORM con el objeto de generar automáticamente las clases de dominio para el código de servidor, basado en la estructura relacional propuesta en el modelo entidad relación de base de datos.

De igual forma se tendrá un apartado para cada segmento de la aplicación, refiriéndonos a la parte administrativa y a la parte de negocio, denominadas orm dominio de seguridad y orm de negocio correspondientemente.

En cuanto a los niveles de seguridad para el modelo de dominio de seguridad, se tendrá una definición contemplando usuarios y roles apoyados en el framework de trabajo unificado de

ASP.NET MemberShip de Microsoft.

3.4.1 ORM dominio seguridad

Es una fiel representación de los objetos de bases de datos del esquema ADM de administración. Al utilizar un ORM, nos entenderemos en el ámbito del código de servidor, con un modelo de objetos de la persistencia en base de datos. Esto agiliza el ciclo de vida de desarrollo del software potencialmente. En cuando a la seguridad, al utilizar el API de Microsoft, de libre uso, llamado MemberShip y que se integra totalmente con el marco de trabajo ASP.NET MVC utilizado en este proyecto, junto con el motor de base de datos SQL Server Express, se tendrá todo el control de la seguridad en esta capa de modelo.

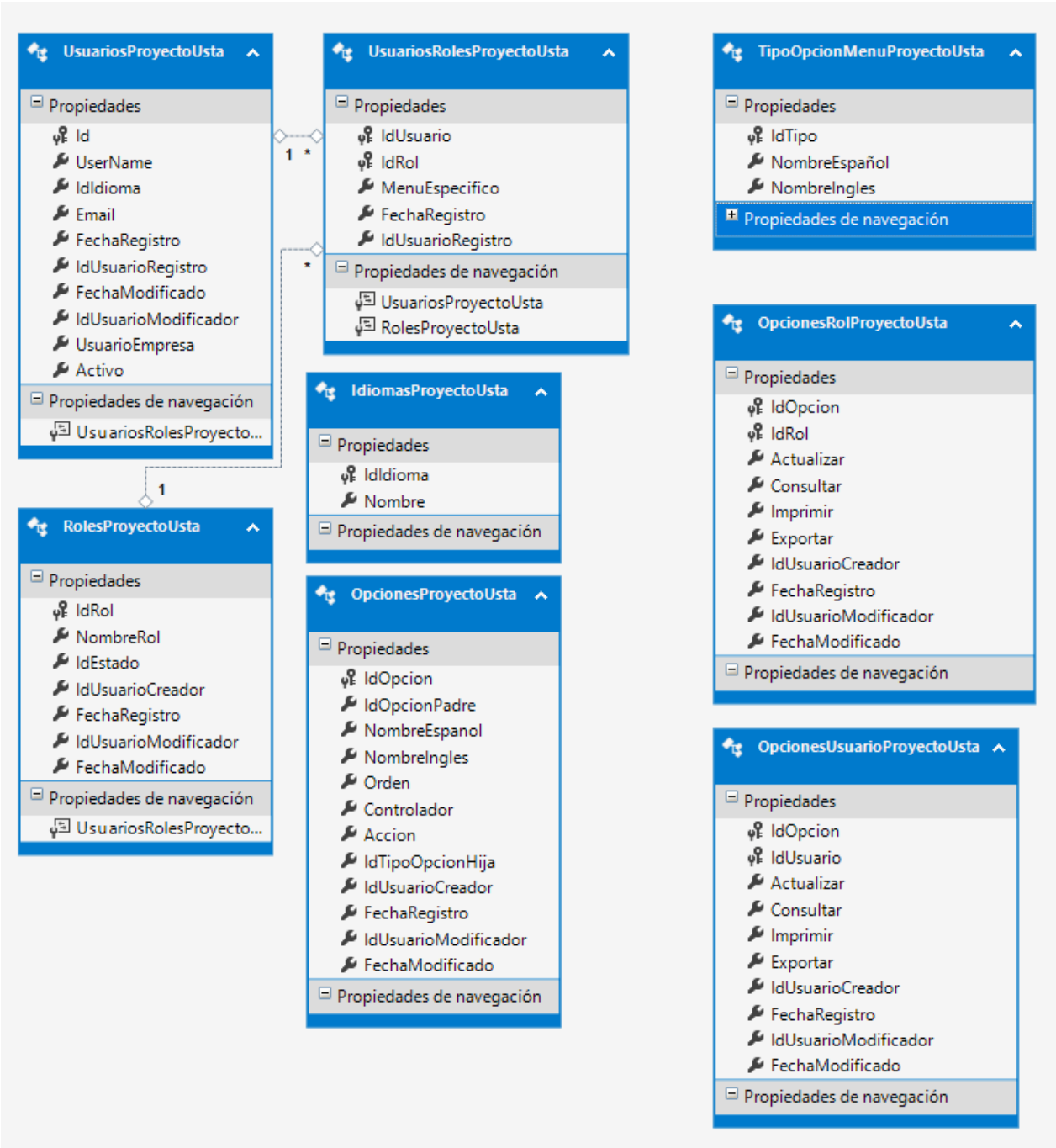


Imagen 52. Modelo de clases Seguridad Sistema ORM

3.4.2 ORM dominio negocio

Al igual que el ORM dominio seguridad, el ORM dominio negocio, es una fiel representación de las tablas de bases de datos del esquema BUS (Business) donde se tiene cada uno de los elementos del negoció, representado en una capa de objetos para ser manipuladas con código de servidor. Pretende agilizar potencialmente las transacciones de lectura, escritura, actualización y eliminación hacia el motor de base de datos, cuya funcionalidad se detallará en el apartado de modelamiento arquitectónico.

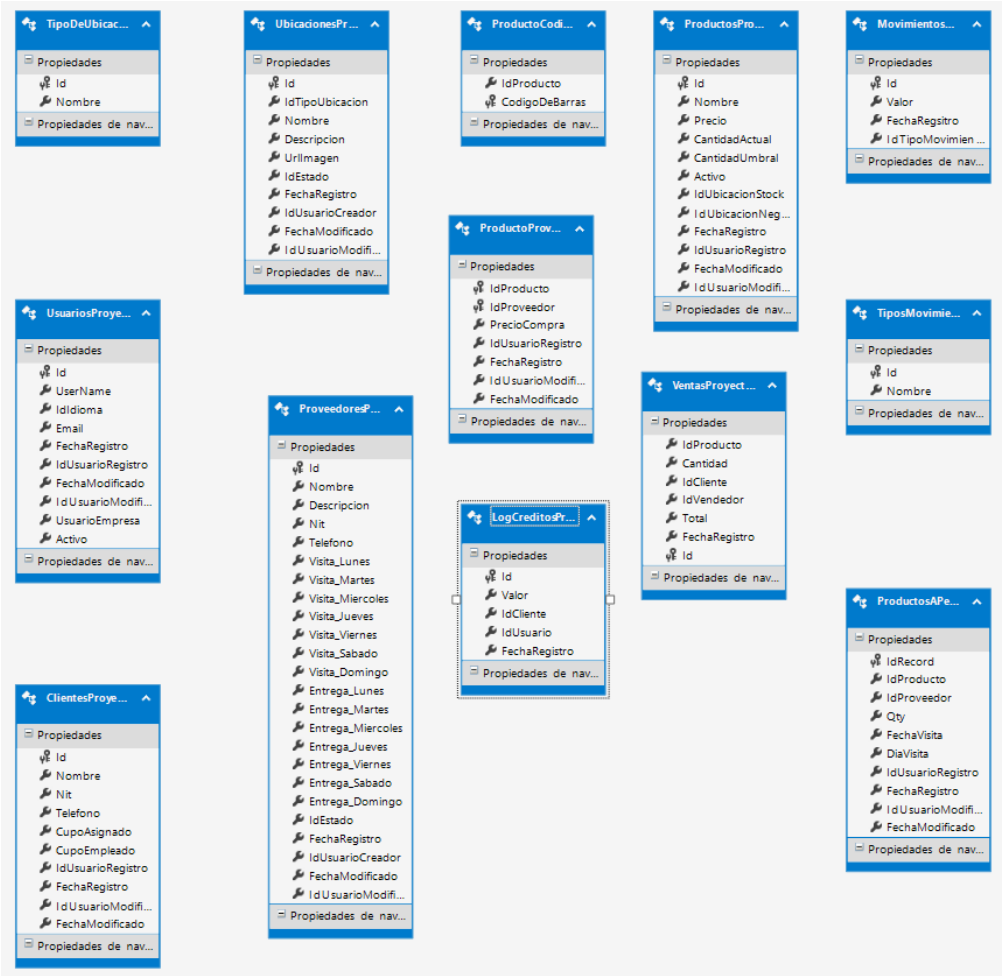


Imagen 53. Modelo de clases Negocio Sistema ORM

3.5 Modelo arquitectonico

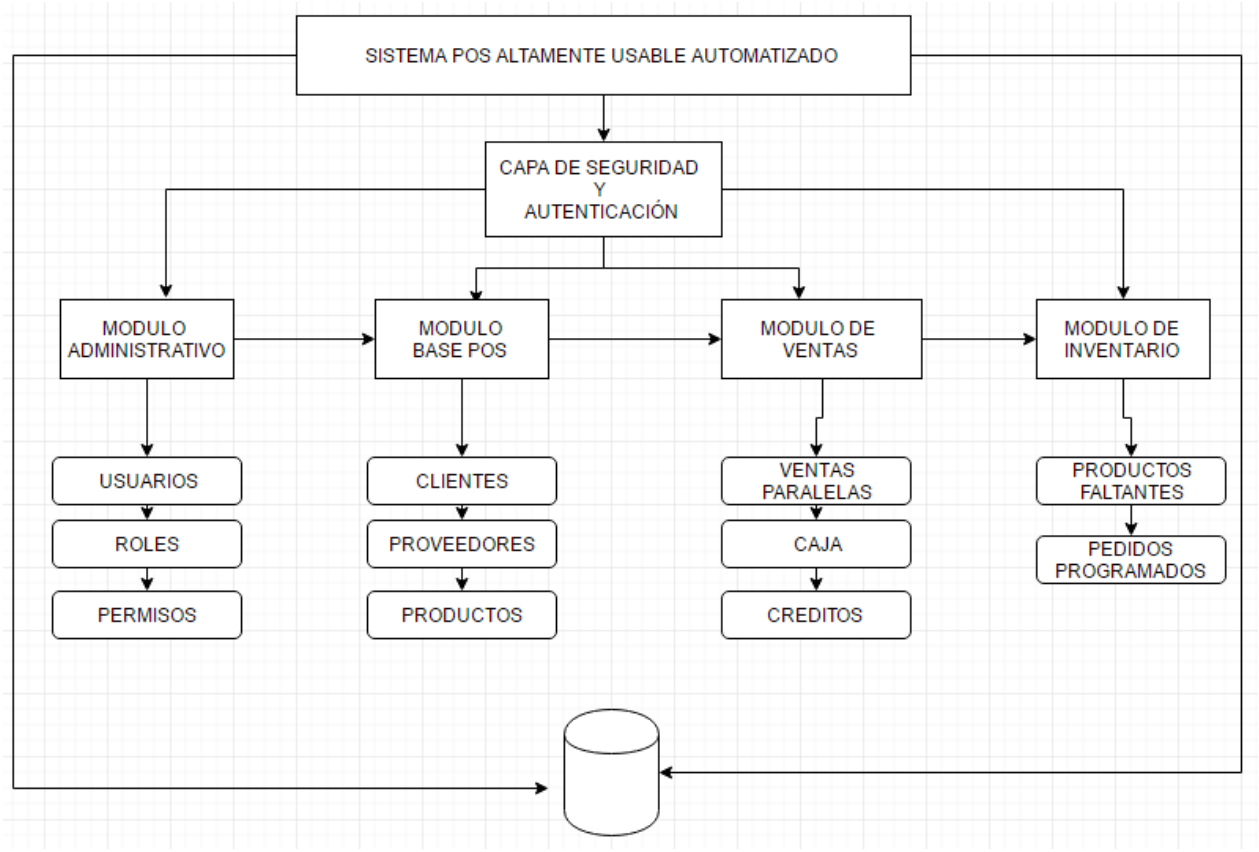


Imagen 54. Modelo arquitectónico de la aplicación

El sistema contempla el modelo arquitectónico basado en la segmentación e interoperabilidad de los siguientes componentes lógicos con sus respectivas funcionalidades.

El modelamiento arquitectónico se interesa por entender cómo debe organizarse un sistema y como tiene que diseñarse la estructura global de ese sistema (Sommerville, 2011, pág. 148).

El desarrollo arquitectónico basado en módulos es una abstracción muy arraigada a los conceptos orientados a objetos. Un módulo actúa como un conjunto de una o más declaraciones de programa contiguas que tienen un nombre por el cual otras unidades del sistema pueden invocarlo o referenciarlo. (Schach, 2006, pág. 167).

3.5.1 Módulo de seguridad y autenticación

Corresponde a la funcionalidad transversal de seguridad del sistema, implementada en el API de libre uso de Microsoft MemberShip. Este se compone de un conjunto de librerías de código de servidor .dll, específicamente WebMatrix.Data y Webmatrix.WebData, que en conjunto con un repositorio de datos SQL Server, maneja toda la funcionalidad de gestión de contraseñas de usuarios, log de ingreso de usuarios de la aplicación, verificación de sesión y acceso a los diferentes módulos y vínculos del sistema, basado en una configuración de roles. Membership tiene varios modos de trabajo, el elegido para la implementación de este proyecto es “Simple membership” ya que permite controlar la definición de la estructura de las tablas de base de datos, ampliarla, adecuarla de acuerdo con las necesidades propias del sistema y aun así utilizar todas sus funcionalidades automáticas de gestión de una manera organizada y sencilla.

En conjunto con el acceso y registro de usuarios al sistema, este componente provee funcionalidades de autorización basada en roles, lo que permite segmentar los accesos de la aplicación, a un grupo de usuarios pertenecientes a un rol, así como el acceso a los vínculos o módulos a los que el rol tiene acceso.

Este módulo lógico arquitectónico, es el punto de entrada, verificación y seguimiento a las demás funcionalidades o módulos lógicos arquitectónicos del sistema.

3.5.2 Módulo lógico de administración del sistema

Se compone de las funcionalidades de administración de usuarios, administración de roles y administración de permisos. Es el componente principal para la administración del sistema. Está relacionado directamente con el módulo lógico de seguridad y autenticación, en lo referente a el

acceso por roles. Los casos de uso relacionadas con este componente arquitectónico son todas las del ámbito “administración del sistema”. Este módulo esta soportado en toda la persistencia comentada en la sección de modelo de clases y modelo entidad relación (imagen 52, imagen 50, tabla 5). Los servicios expuestos por este módulo son el de acceso a las páginas, registro de ingreso a la aplicación, registro de salida de la aplicación, actualización de contraseñas y actualización de información de los usuarios del sistema.

3.5.3 Módulo lógico base POS

Se trata de la base del negocio del sistema POS, en ella se comprenden todas las entidades a utilizar en procesos de ventas, inventario y programación automatizada de pedidos del sistema. Entre las funcionalidades se encuentran, el repositorio y funcionalidad de administración de los clientes del sistema, como también el repositorio de datos y de administración de los productos del sistema, de igual manera el repositorio de datos es y administración de los proveedores del sistema. Todas estas funcionalidades están interrelacionadas entre sí, de manera física y lógica, tanto en la capa de datos, como en la capa lógica de programación. Es un módulo desacoplado del módulo de administración del sistema tanto como del módulo de seguridad y autenticación, pero está fuertemente relacionado con el módulo de ventas del sistema, como el módulo de inventario. En resumen, este módulo proporciona todos los elementos de insumo para el módulo de ventas y el módulo de inventario.

3.5.4 Módulo lógico ventas

Hace referencia a todos los elementos, de repositorio de datos, páginas con funcionalidad de

interfaz gráfica proporcionando alta usabilidad como las ventas paralelas, así como todos los procesos de movimientos económicos del sistema. Ofrece servicios de registro de ventas en bloque, individuales, contemplando afectar movimientos de caja y de créditos. Es un módulo muy relacionado con el módulo de inventario tanto con el módulo “base pos”, ya el este último le ofrece los insumos de trabajo, pero el módulo “inventario” utiliza los registros generados por el módulo de ventas para ofrecer una funcionalidad consistente en cuanto al inventario, haciendo énfasis en la automatización de programación de pedidos, objetivo de trabajo de este proyecto informático.

3.5.5 Módulo lógico de inventario

Proporciona funcionalidades específicas de automatización de pedidos basado en algoritmos de análisis de ventas junto con configuraciones predefinidas de usuario para cada producto, de igual manera proporciona funcionalidades específicas de mantenimiento al inventario, como la realización parcial del mismo, así como la retroalimentación de ingreso de productos, lo que completa el ciclo de entrada, salida y reabastecimiento de un producto en su proceso de venta.

Los insumos que utiliza este módulo son en su mayoría del módulo de ventas, y en una parte esencial, pero de menor participación, del módulo “base pos”.

3.6 Arquitectura de software

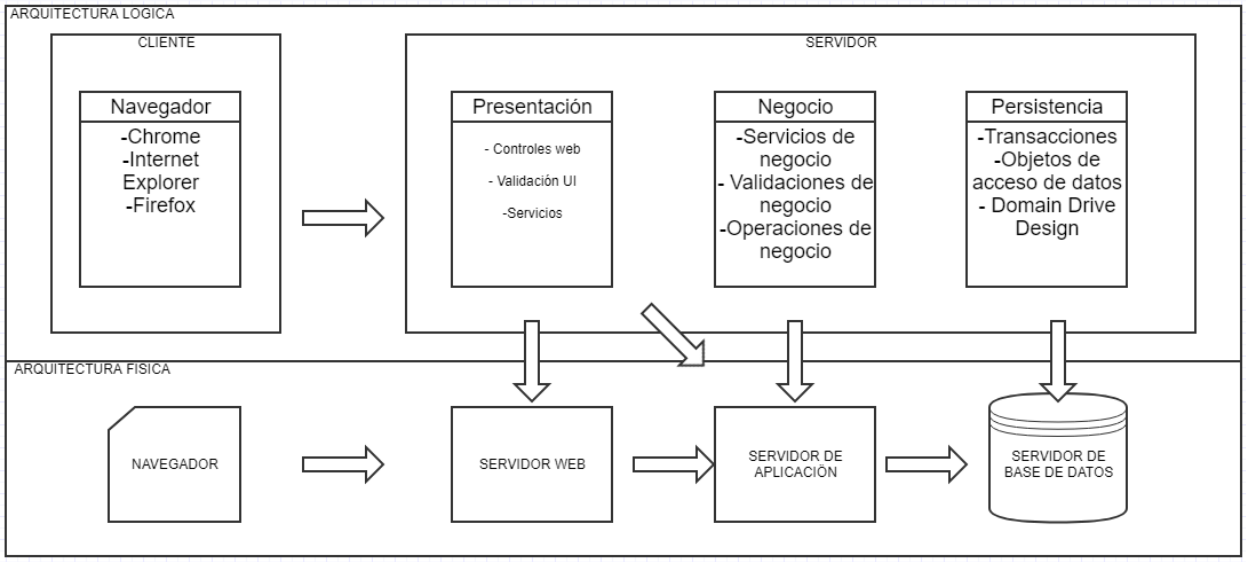


Imagen 55. Arquitectura general

Apoyado en la arquitectura cliente servidor, para el caso de los aplicativos webs, se cuenta con un navegador que efectúa las peticiones, resultado de la operación de cada una de las funcionalidades del sistema, realizando un llamado, o envío de mensaje al servidor, quien a su vez soportado en un programa denominado servidor web, procesa estas peticiones, encaminándolas a su respectivo manejador.

La primera capa que recibe estas peticiones, se denomina capa de presentación, y es a su vez el módulo que se encarga de responder al cliente el resultado de todas las operaciones, bien sea en la entrega de una página HTML, o mensajes, resultado de operaciones tipo Ajax.

Siguiendo el camino de la petición, la capa anterior, haciendo uso del servidor web, procesa este mensaje, serializando toda la información enviada, y lo comunica al nivel físico de servidor de aplicación, donde se encuentra toda la lógica del negocio, que, soportándose en una capa final o de persistencia, realiza todos los llamados, inserciones y actualizaciones al motor de base de datos donde se almacena la información.

El retorno de la aplicación se realiza recíprocamente, donde cada capa que proceso el mensaje, en su siguiente instrucción hará las operaciones enfocadas en devolver los datos a la capa anterior que lo invoco. En este proceso, los datos viajan de forma inversa entre las capas y nuevamente se serializan para que sean procesados por el cliente.

En la imagen 57, se puede apreciar el modelo transversal y relacionado de los módulos de administración del sistema.

Imagen 57. Modelamiento arquitectónico de la administración del sistema

3.6.1 El lenguaje ubicuo

Como se referenció en el marco teórico del proyecto, esta parte de la arquitectura de software anima a que se hable en un lenguaje estandarizado de las entidades o dominio del negocio de la aplicación con el ánimo de entender de qué se está discutiendo en cada una de las fases de definición del sistema. La principal meta de este apartado es evitar errores o malentendidos por todos los involucrados en el desarrollo de un sistema. Comprende sustantivos, verbos, adjetivos que son compartidos como un manifiesto para cada uno de los participantes en la producción de un sistema informático.

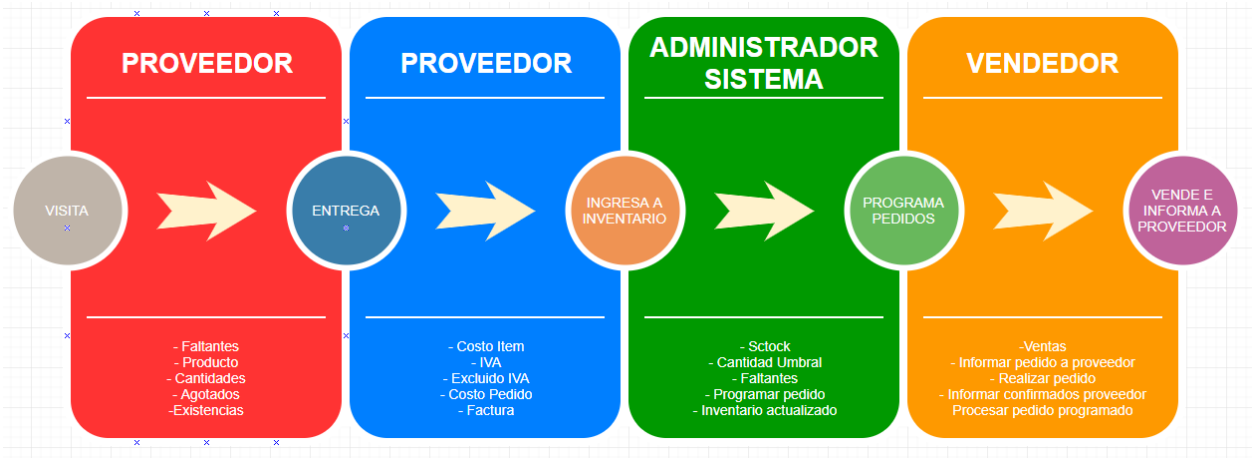


Imagen 58. Proceso repetitivo del mini mercado, con varios términos que han de manejarse de forma ubicua.

Como se aprecia en la imagen 58, se evidencia el proceso repetitivo de entrada y salida de productos en el sistema, involucrando a los actores principales. El proceso involucra el manejo de términos en cada una de las etapas, y deberá promoverse a que se entiendan estos términos y verbos entre cada etapa con el ánimo de entablar la misma terminología y significado para cada entidad o evento del proceso.

3.6.2 Capas arquitectónicas

La implementación de las capas arquitectónicas con el ánimo de separar responsabilidades y promover la reutilización de código, se agrupa con el contenedor de proyectos tipo solución. Este elemento es propio de las tecnologías .NET con el entorno de desarrollo integrado IDE, visual studio.

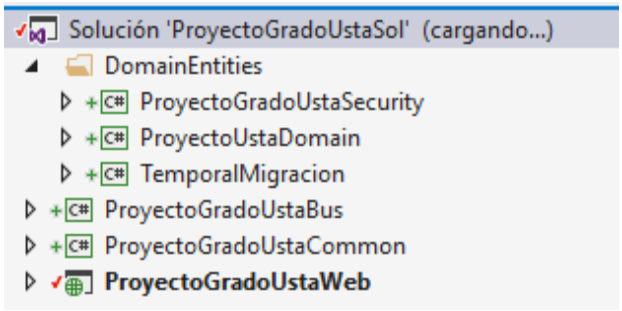


Imagen 61. Capas de la solución en IDE Visual Studio

Cada una de las librerías de clases C# representa a un conjunto de código, segmentado en cada capa para la realización de un trabajo en particular explicado a continuación.

3.6.3 Capa de presentación

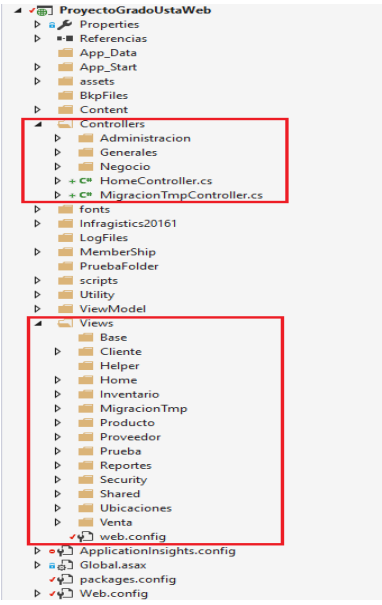


Imagen 62. Patrón de diseño MVC en capa de presentación.

Implementando el patrón de arquitectura de software ampliamente conocido como MVC, realizamos la codificación de cada una de las vistas o páginas web mencionadas en este proyecto con una lógica mínima dentro de ellas salvo las validaciones de cliente e incluso un código de cliente que garantice la experiencia y usabilidad, objetivo de este proyecto informático.

La función de las vistas es entregar elementos HTML con un estado que es controlado por otra capa de la aplicación que no está en este contexto que es la capa de negocio.

Adicional en esta capa podemos ver a los controladores que no son más que segmentos de código que realizan el proceso de recibir los datos de entrada de las peticiones http generalmente, llamar a la capa de negocio y haciendo uso de una capa adicional de modelo que tampoco hace parte de esta capa, gestionar el llamado de las vistas con su respectiva información dependiendo el estado de la aplicación.

Esta capa provee endpoints o puntos de entrada de servicios que son ofrecidos generalmente a diferentes clientes o paginas servidas por esta capa de presentación o de software tercero con el ánimo de garantizar la conectividad del sistema.

3.6.4 Capa de negocio

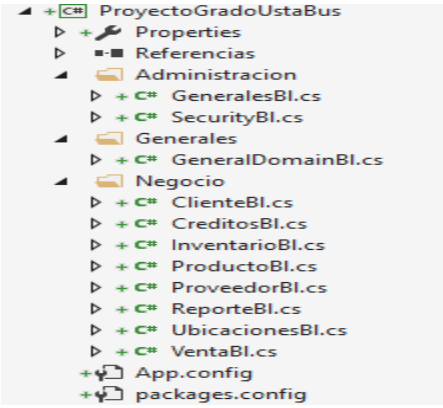


Imagen 63. Capa de negocio en IDE Visual Studio.

Es la capa que realiza las necesidades funcionales del sistema y la que contiene el código más amplio de la aplicación en cuanto a procesos propios del modelo de negocio de la solución informática. Aunque es claro que un diseño o aplicación con alta usabilidad tendrá bastante codificación y funcionalidad con respecto a código cliente, al final del camino todo tendrá que

pasar y salir por esta capa de aplicación o negocio. Para esta capa se han segmentado las funcionalidades en diferentes folders con el ánimo de estructurar aún más su aplicabilidad. Esta capa es generalmente llamada por los controladores MVC de la capa de presentación y esta orquesta el llamado a la capa de infraestructura o persistencia y proyecta los diferentes modelos basados en la capa de dominio de la aplicación basada en clases POCO (Plain old clr object / Objeto de clase estandar). Se puede apreciar cada una de las clases encargadas en la carpeta "Negocio", donde existe una clase para cada funcionalidad del dominio como lo son: ClienteBl, CreditosBl, InventarioBl. Adoptando un estándar, todas las clases de esta capa, deben finalizar con las siglas "Bl" (Businness Logic / Logica de negocio), haciendo relación a que es un artefacto de codificación perteneciente a esta capa.

3.6.5 Capa de dominio

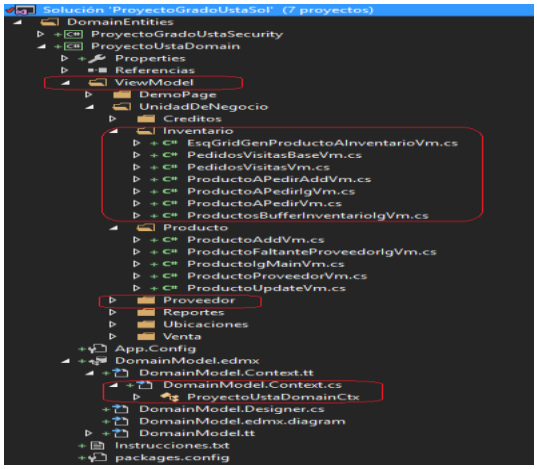


Imagen 64. Capa de dominio en IDE Visual Studio.

Es la capa que provee todo el dominio principal de la aplicación en cuanto a clases que representan entidades propias del negocio, en este caso en particular representan directamente a una tabla de nuestro sistema SGBD. Cada entidad representa una entidad del negocio lo cual hace más entendible su manipulación. En un modelo DDD como se mencionaba inicialmente en

este apartado de arquitectura de software, se tendría una segmentación más estructurada de esta capa para poder hacer uso de interfaces, herencia y polimorfismo, así como de inyectores de dependencia o mappers que faciliten la proyección de las entidades en view models específicos para cada utilidad UI que solicite la capa de presentación. Se agrupan en el folder principal de los view models, elementos específicos del dominio, relacionados con la interfaz de usuario o con el proceso de persistencia. Estas se agrupan por unidad de negocio o entidad del dominio, y allí se ubicarán todos las clases o artefactos de la entidad. En la imagen 64, se puede apreciar todos los view models relacionados con la entidad o unidad de negocio "inventario".

3.6.6 Capas de infraestructura

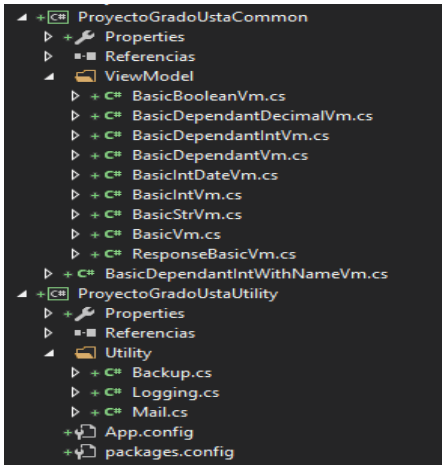


Imagen 65. Capas de infraestructura en IDE Visual Studio.

Se realiza dos capas que serán transversales a toda la aplicación, denominadas ProyectoGradoUstaCommon y ProyectoGradoUstaUtility, las cuales tendrán entidades view model específicas para toda la aplicación, utilizadas en mensajes cliente servidor e incluso en formatos de mensajes entre capas, como también la capa utilidades, que proporcionarán elementos para el backup del sistema, logging de la aplicación y utilidades de envíos de correo y notificaciones.

3.7 Metodología de desarrollo

3.7.1 Creación proyecto scrum

Inicialmente creamos el proyecto y cada una de sus tareas en la herramienta visual studio team foundation online de uso gratuito para el registro de nuestro proyecto con esta metodología.

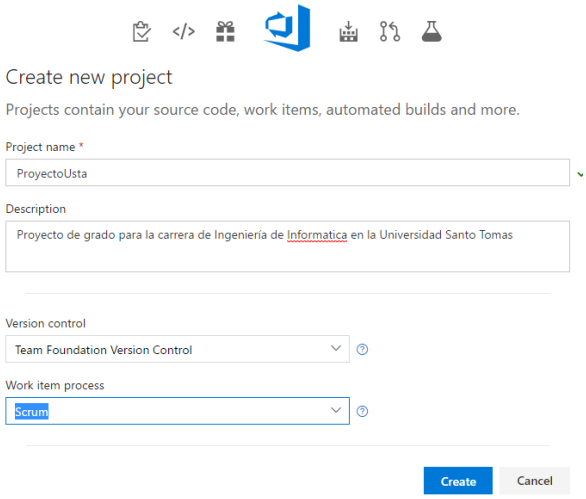


Imagen 66. Creación de tareas bajo metodología SCRUM en Teamfoundation free Microsoft.

En consenso con el interesado o product owner del mini mercado El DORADO, se han ingresado cada una de las actividades detalladas en el cronograma de actividades, priorizando que tareas han de realizarse primero y segmentándolas en diferentes sprint de trabajo obtenemos el product backlog del proyecto.

3.7.2 Dashboard general

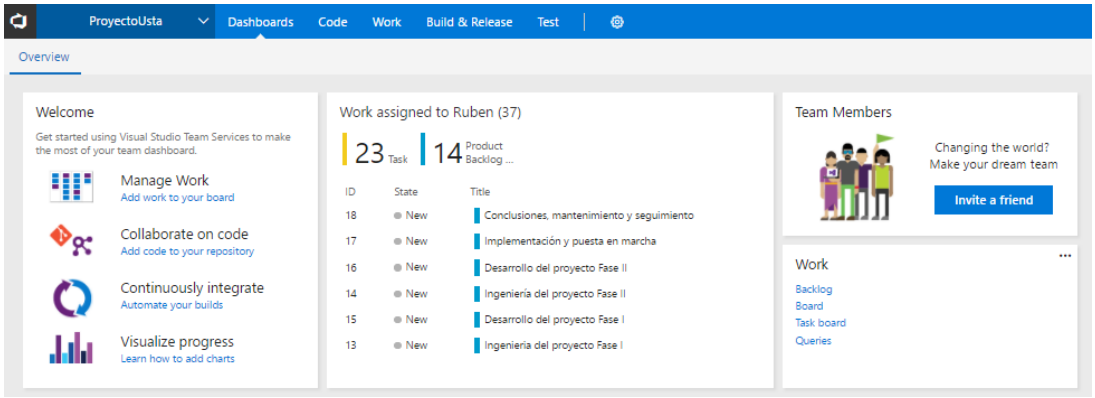


Imagen 67. Resumen general principal de PBI y tareas del proyecto.

3.7.3 Features proyecto

De igual manera en conjunto con el producto owner del mini mercado EL DORADO, se han definido los 4 features que componen el proyecto, como se aprecia en la imagen 67a.

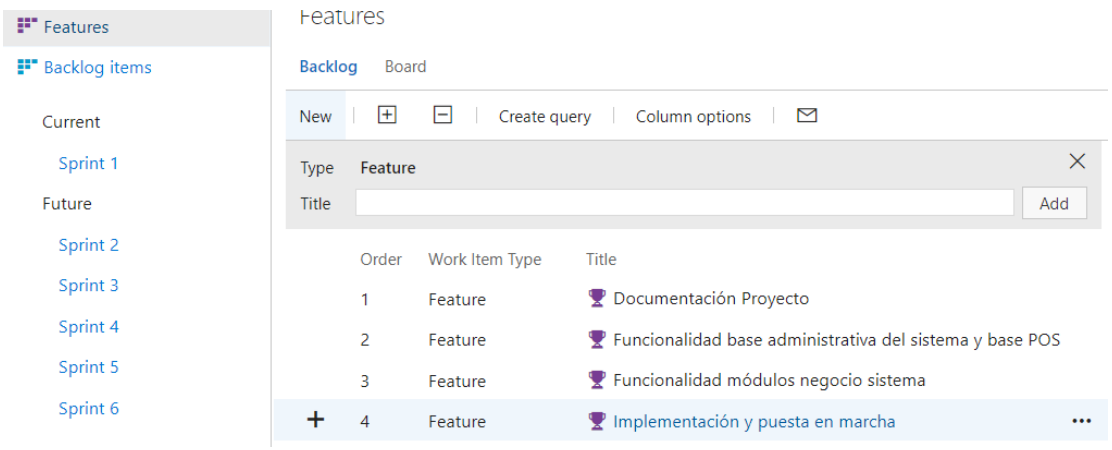


Imagen 67a. Scrum features del proyecto.

Donde de manera general podemos ver los 14 product backlog que la metodología exige y su división en cada uno de los sprint del desarrollo.

3.7.4 PBI proyecto

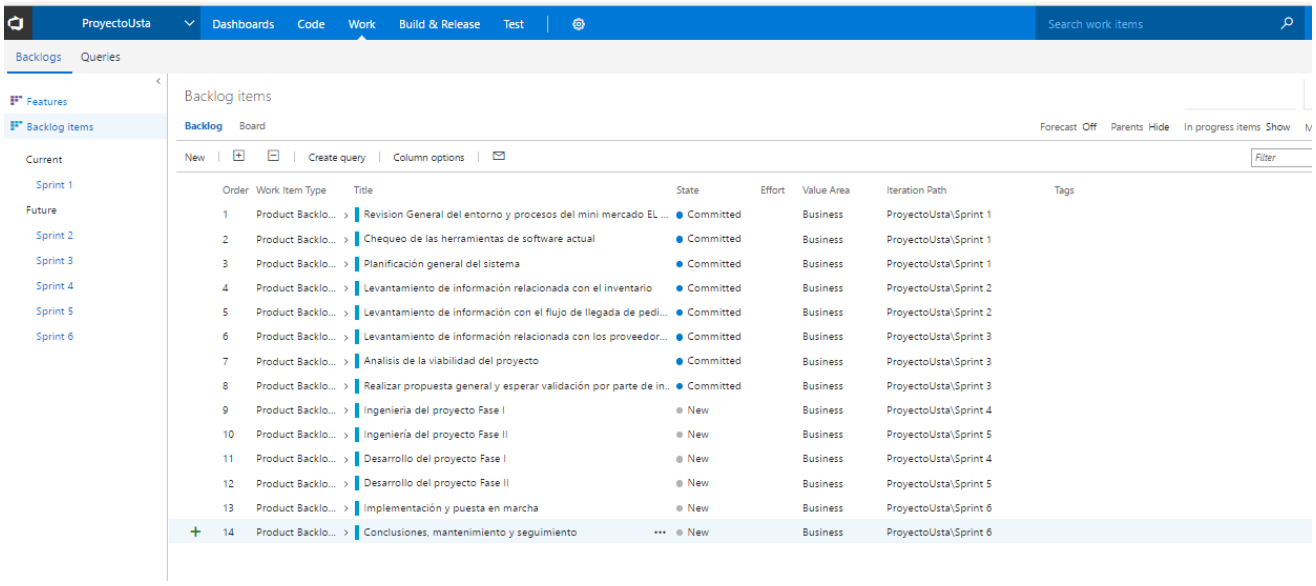


Imagen 68. Backlog ítems generales

En detalle, cada sprint tiene sus respectivos PBI (Prducto backog ítem) y cada PBI a su vez tiene sus respectivas tareas. En la imagen 68a se pueden apreciar los PBI del sprint 1 como las tareas del PBI “Revisión general del entorno en procesos del mini mercado” (imagen 68b).

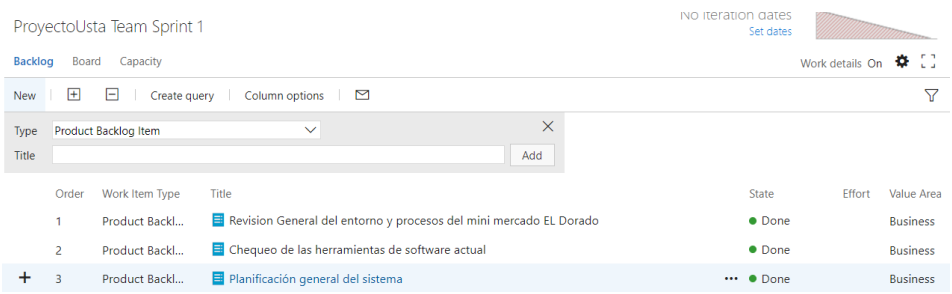


Imagen 68a. PBI sprint 1.

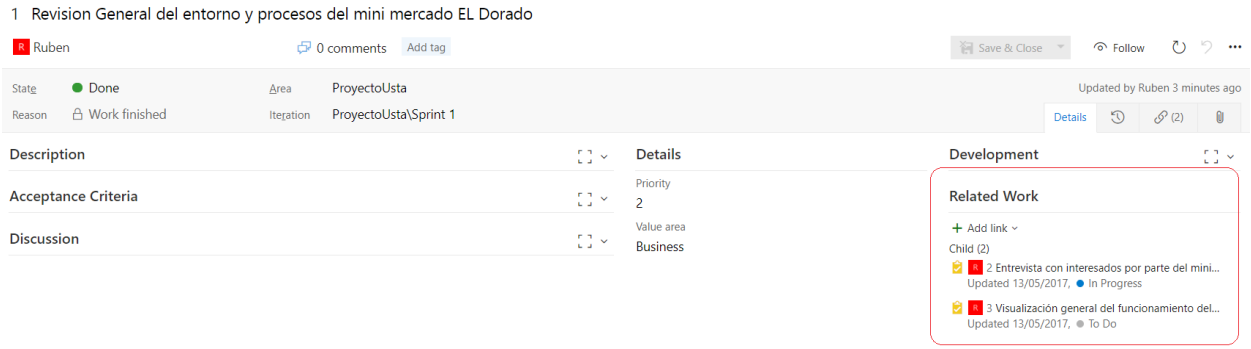


Imagen 68b. Tareas PBI "Revisión general del entorno y procesos del mini mercado EL DORADO".

En el dashboard o tablero de avance, se pueden apreciar unas tareas en estado de aprobación por parte del product owner, así como unas tareas nuevas. Este es el tablero indicativo del sprint 1 en sus comienzos donde no se había realizado aún ninguna tarea.

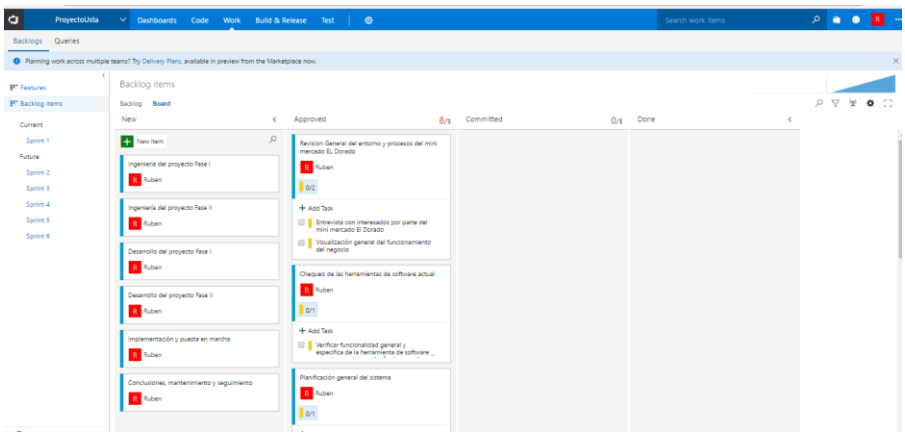


Imagen 69. Estado de tareas en sprint.

Para cada sprint podemos ver los PBI correspondientes a realizar.

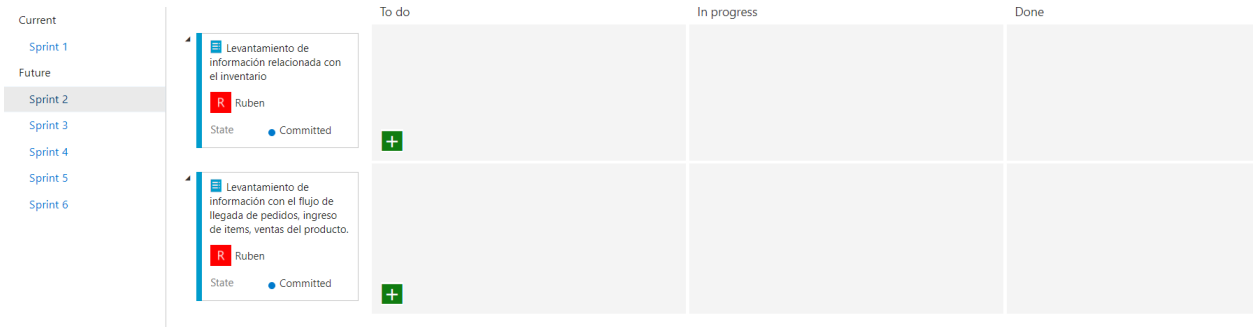


Imagen 69a. PBI Sprint 2

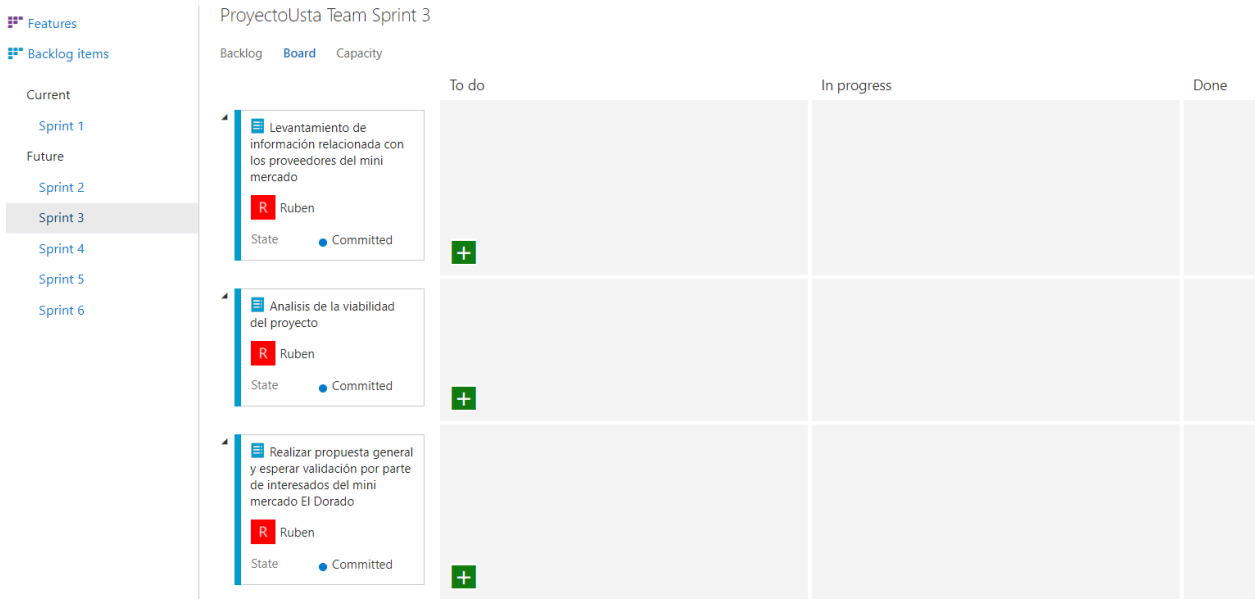


Imagen 69b. PBI sprint 3.

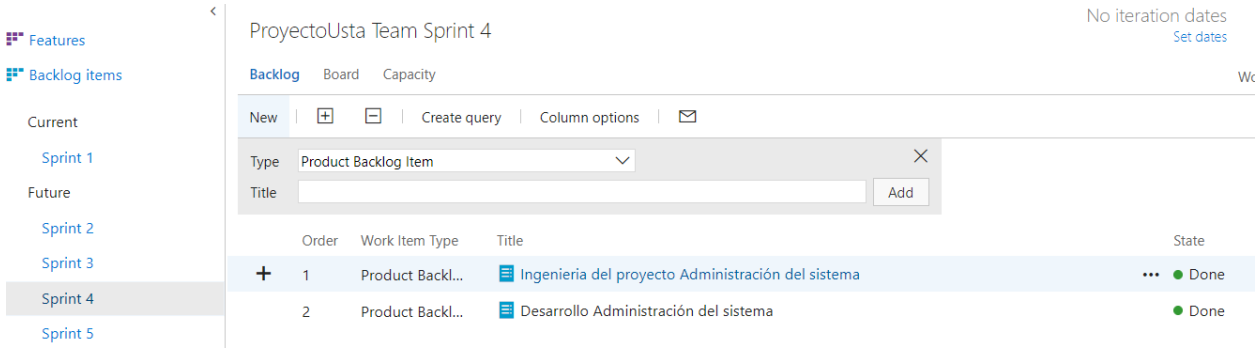


Imagen 69c. PBI sprint 4

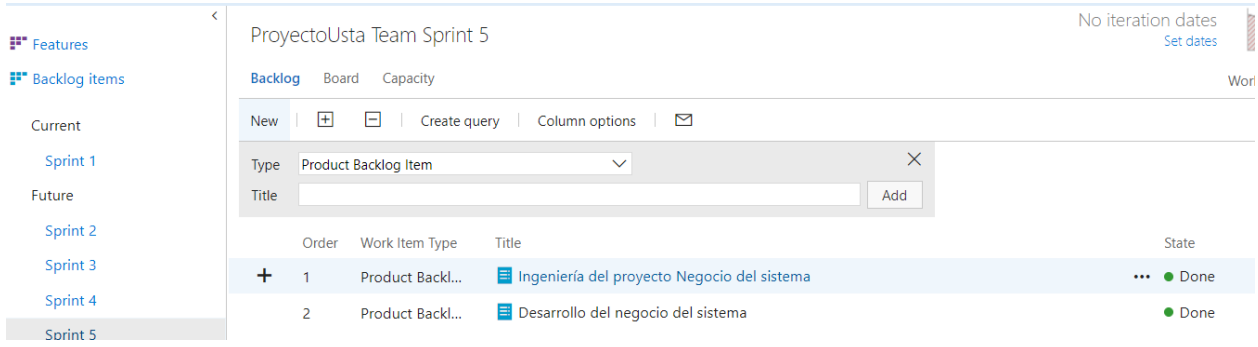


Imagen 69d. PBI Sprint 5.

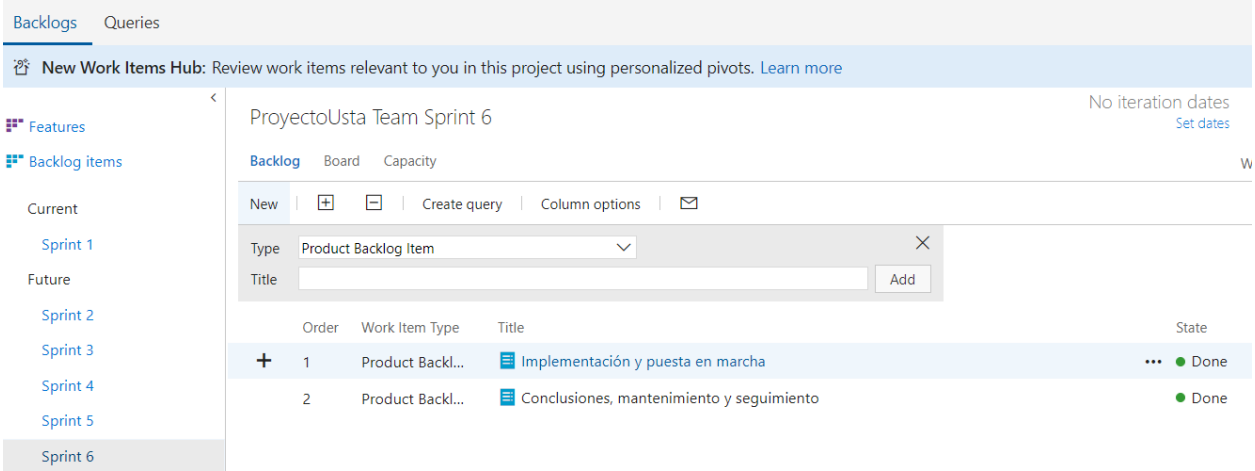


Imagen 69e. PBI Sprint 6.

3.7.5 Finalización proyecto

Todos estos sprint fueron llevados a cabo ordenadamente, y en cada paso se podía evidenciar el avance, en cuanto a tareas que estaban en proceso, por hacer y ya realizadas.

Al finalizar cada uno de los sprint, la herramienta informa que el trabajo se ha realizado por completo (imagen 69f).

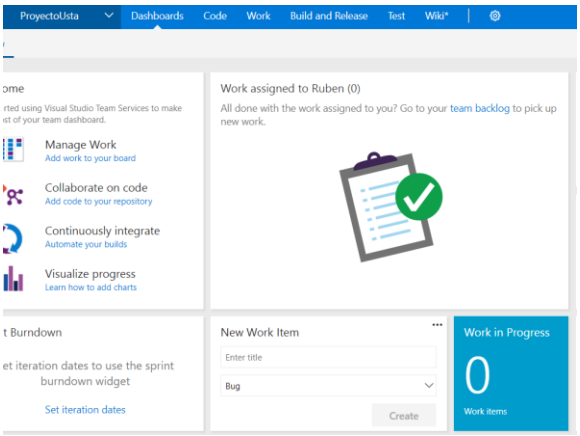


Imagen 69f. Tareas completadas.

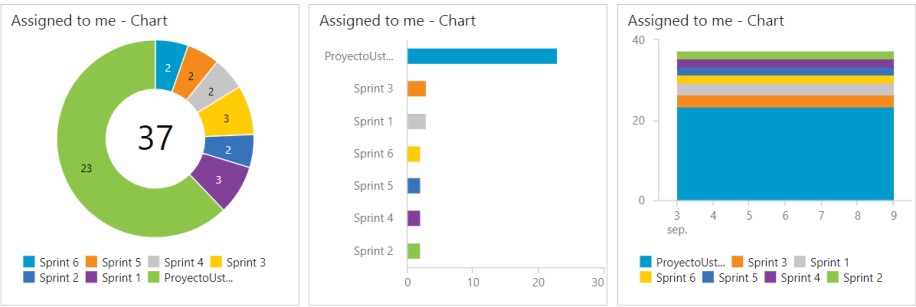


Imagen 69g. Iteraciones totales PBI y por sprint.

3.8 Implementación

Para el proceso de implementación de la solución bien sea de manera incremental o como producto de una nueva instalación, es preciso referirnos a los apartados 1.9 viabilidad técnica y 1.11 viabilidad de gestión como prerrequisitos de esta implementación. De igual manera como requisitos físicos se debe contar con los siguientes componentes con especificaciones técnicas mínimas recomendadas, los cuales el mini mercado EL DORADO tiene en su momento.

- Equipo de cómputo con procesador igual o mayor 1 Ghz de velocidad, memoria RAM de 8 gigabytes, espacio libre de disco duro mínimo de 20gb, pantalla mínima de 14 pulgadas, sistema operativo Windows en cualquiera de las versiones detalladas (xp, vista, 7, 8 o 10)
- Lector de código de barras de 2d conectable por medio de USB y compatible con los sistemas operativos compatibles con el sistema (imagen 73).

Imagen 73. Lector de código de barras 2d

Una vez se cumpla con los requisitos técnicos, se deberán instalar los siguientes programas:

- Sql Server versión Express en cualquiera de sus ediciones especificadas (2008, 2012, 2014 o 2016) y asegurar de tener las credenciales de acceso del usuario sa (super administrador).

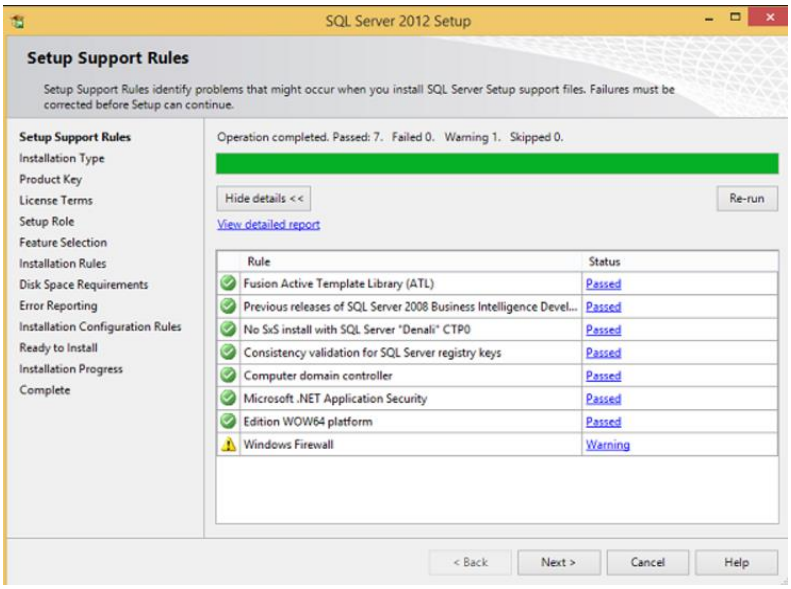


Imagen 74.

- Habilitar el programa IIS (internet information services) disponible en cada uno de los sistemas operativos mencionados por medio de la configuración de programas incluidos del sistema de Windows.

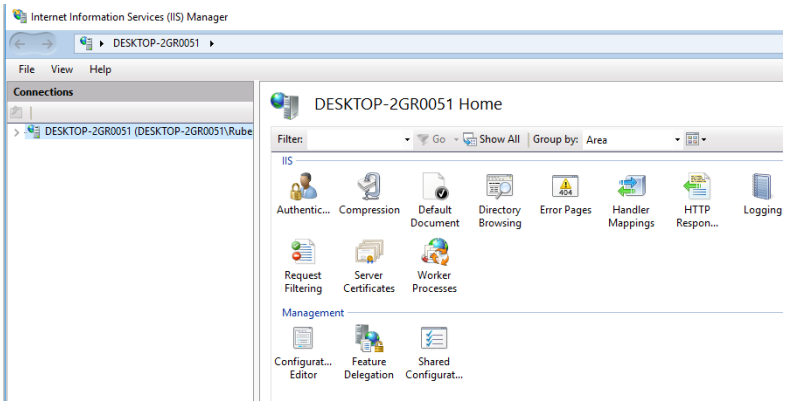


Imagen 75. Internet Information Services

- Crear carpetas de destino de publicación del sistema dentro de la carpeta inetpub de Windows, asegurando que la cuenta con la que se acceda a el disco sea una cuenta con privilegios ASP.NET

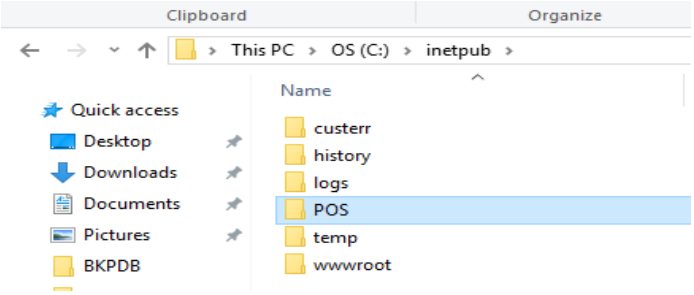


Imagen 76. Estructura de folder, archivos publicación web

- Crear pool de aplicación ASP.NET en programa IIS (internet information services) de tipo integrado, haciendo uso de la versión del .NET framework 4.0

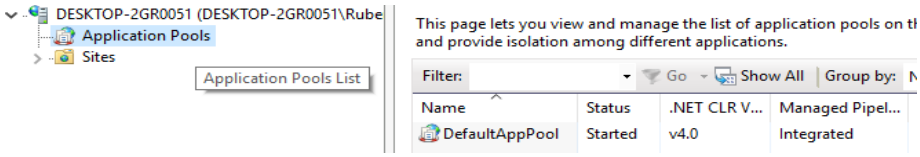


Imagen 77. Application pool Internet Information Services

- Crear el sitio web, utilizando el gestor de servicios web IIS (internet information services), para poder alojar la aplicación.

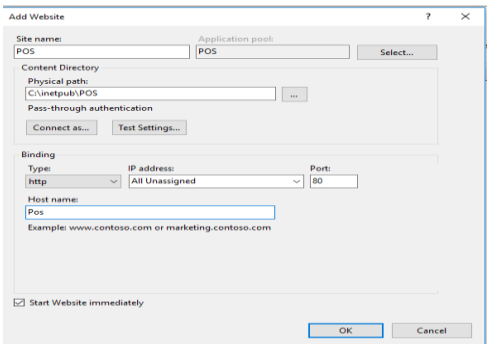


Imagen 78. Creación sitio web Internet information Services

- Desplegar versión compilada del sistema en carpeta destinada para tal fin.

- Configurar parámetros de conexión, específicamente las cadenas de conexión relacionadas con la instancia del motor de base de datos, previamente instalado en la máquina de producción (imagen 79). Adicionalmente se parametriza la cuenta de correo electrónico a la que el sistema notificara eventos administrativos vía correo electrónico (imagen 80).

```
<!--  For more information on how to configure your ASP.NET application, please visit  http://go.microsoft.com/fwlink/?LinkId=301880
<configuration>
  <configSections>
    <!-- For more information on Entity Framework configuration, visit http://go.microsoft.com/fwlink/?LinkID=237468 -->
    <section name="entityFramework" type="System.Data.Entity.Internal.ConfigFile.EntityFrameworkSection, EntityFramework, Version=6.
  <!-- For more information on Entity Framework configuration, visit http://go.microsoft.com/fwlink/?LinkID=237468 --></configSectio
  <connectionStrings>
    <add name="ProyectoGradoUstaMemberShip" connectionString="Data Source=DESKTOP-2GR0051;Initial Catalog=ProyectoUsta;user id=sa;pa
    <add name="ProyectoUstaEntities" connectionString="metadata=res://*/SecurityModel.csdl|res://*/SecurityModel.ssdl|res://*/Securi
    <add name="ProyectoUstaDomainCtx" connectionString="metadata=res://*/DomainModel.csdl|res://*/DomainModel.ssdl|res://*/DomainMod
  </connectionStrings>
  <appSettings>
    <add key="webpages:Version" value="3.0.0.0" />
```

Imagen 79. Configuración de cadenas de conexión a bases de datos.

```
  <system.net>
    <mailSettings>
      <smtp from="supermercadoeldorado202@gmail.com">
        <network host="smtp.gmail.com" port="587" userName="supermercadoeldorado202@gmail.com" password="erpAX2014
      </smtp>
    </mailSettings>
  </system.net>
  <system.web>
    <authentication mode="Forms">
      <forms loginUrl="~/Security/LogIn" timeout="500" />
    </authentication>
    <sessionState timeout="500" />
    <compilation debug="true" targetFramework="4.5" />
    <httpRuntime targetFramework="4.5" />
    <membership defaultProvider="SimpleMembershipProvider">
    <providers>
```

Imagen 80. Configuración de correo de notificaciones mail administrador del sistema.

- Iniciar sitio web desde internet information services IIS o desde cualquier navegador web, apuntando al sitio previamente creado.

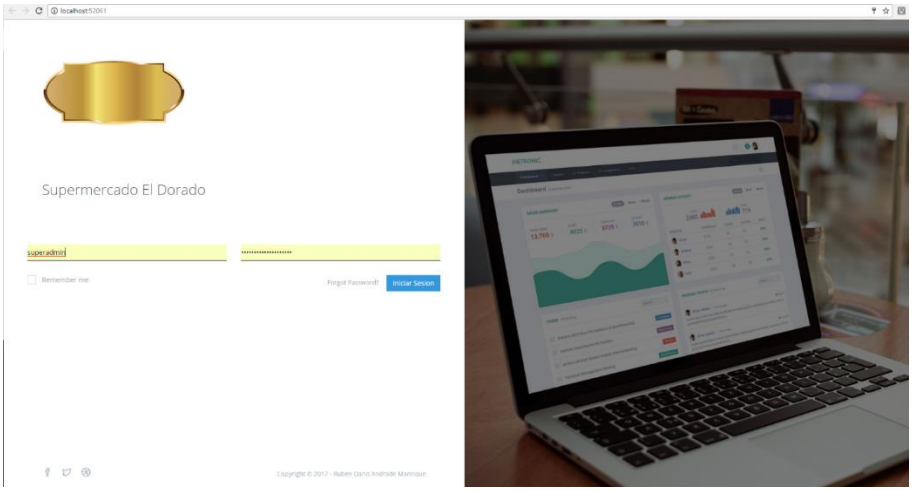


Imagen 81. Página de inicio login del sistema.

Adicional al proceso técnico debe contemplarse la capacitación y disponibilidad de uso por parte de los usuarios del mini mercado EL DORADO, para lo cual se destinaria un mínimo de 4 horas, para la capacitación en todos los módulos del sistema.

4- Análisis de resultados

4.1 Conclusiones

La realización de este proyecto fue un proceso bastante enriquecedor para mi carrera profesional y de igual manera para los procesos del mini mercado EL DORADO. Actualmente es la herramienta de uso principal y se ha tenido una aceptación muy buena de parte del personal del mini mercado, así como de los clientes.

Adicional a la instalación del sistema en el mini mercado EL DORADO, se han realizado un par de instalaciones más, de manera exitosa en negocios similares de venta al detal, con tipos de productos totalmente diferentes, pero que de igual manera le ofrecen al personal del negocio una funcionalidad general de un sistema tipo POS, con las ventajas de usabilidad, automatización y conectividad.

El desarrollo de cada uno de los capítulos y secciones de este proyecto ha sido con el ánimo de dar cumplimiento al objetivo general del proyecto, así como de cada uno de los objetivos específicos. Argumentando cada uno de los beneficios obtenidos, los agruparemos de acuerdo con los objetivos de este proyecto.

4.1.1 Acerca del objetivo general

El mini mercado EL DORADO cuenta ahora con una herramienta desarrollada a la medida de sus necesidades, en especial en los apartados de usabilidad, conectividad, automatización y explotación de sus datos. El personal del mini mercado EL DORADO asegura, que la herramienta le ha proporcionado un mejor desenvolvimiento en las ventas, en especial con la utilidad de ventas paralelas, enfocadas a ofrecer alta usabilidad en la herramienta. De igual manera la opción de poder buscar los pedidos por medio escrito ha sido una funcionalidad bastante aprovechada, ya que la

consulta funciona de manera ágil, proporcionando alta usabilidad.

Al mismo tiempo, el proceso de la comunicación entre proveedor y usuario vendedor, en cuanto al proceso de programación de pedidos, soportado en la funcionalidad automatizada de pedidos, ha servido para evitar reprocesos en esta funcionalidad del sistema, asegurando la programación correcta de la totalidad del pedido y en su defecto, realizando notificaciones vía correo al administrador del sistema, para que tenga la información en tiempo real, de lo que son los estados de sus pedidos y pueda tomar decisiones rápidamente. Esto último haciendo énfasis en la conectividad del sistema.

En cuanto a la automatización de pedidos, es una herramienta que ha requerido un minucioso inventario del sistema, antes de poder hacer uso efectivo de ella, pero que ha tenido una utilidad bastante poderos, como lo afirma la propietaria del mini mercado EL DORADO, ya que para una tarea que de manera visual tenía que realizar, muchas veces cometiendo errores de tipo humano, asegura que esta herramienta es de bastante utilidad y ha reducido el error en su proceso de realización de pedidos en un porcentaje bastante alto, así como se ha reducido el tiempo en que lo realiza; todo esto haciendo énfasis en el objetivo de automatización del sistema. De igual manera la propietaria manifiesta que debido a la arquitectura del sistema, y su ahora posible conectividad, puede realizar esta tarea de forma remota y sin necesidad de ocupar la estación de trabajo principal donde se realizan las ventas.

El personal del mini mercado EL DORADO, manifiesta que el servicio al cliente ha mejorado desde la implementación y puesta en marcha del sistema, así como se tiene un mejor control en los pedidos a proveedores, mejor control o conocimiento del inventario, como una mejor conectividad.

El personal del mini mercado EL DORADO, confía en que se ha cumplido con esto objetivo, y

desde el análisis de los resultados, detallamos cada una de las responsabilidades del objetivo principal.

4.1.2 Acerca de los objetivos específicos

El trabajo en conjunto con el personal del mini mercado, aplicando una metodología de trabajo y entregables incrementales, fue un proceso de trabajo en equipo bastante exitoso ya que en todo momento hubo una comunicación directa y un acuerdo en cada una de las necesidades y cambios que en el camino surgieron. De manera general se pudieron entender completamente las necesidades del cliente y mejor aún se pudieron sistematizar de forma que cumpliera con las necesidades manifestadas en las etapas iniciales de reconocimiento.

De igual forma, se pudo conocer de primera mano las herramientas que el mini mercado había utilizado con anterioridad y se pudo evidenciar las necesidades expuestas por el mini mercado EL DORADO.

Todos los artefactos, como documentos de ingeniería, marcos de referencia para este proyecto, como el enfoque y metodología que se le dio al proyecto, tanto en su programación como en su análisis e ingeniería de requerimientos, fueron de bastante utilidad para la realización del proyecto.

La elección de cada una de las tecnologías y metodologías fueron de manera acertada, ya que fomentaron al desarrollo exitoso de este proyecto.

4.2 Recomendaciones

La ampliación del “sistema automatizado altamente usable para mini mercados”, deberá hacerse bajo los parámetros de arquitectura de software acá descritos ya que promoverán la adecuada mantenibilidad del sistema, así como su escalabilidad.

El sistema ha sido implementado en tres mini mercados, un negocio de distribución al detal de productos de belleza, un negocio de venta de accesorios para celulares y una dulcería que maneja un rango de1000 productos diferentes. Las implementaciones han resultado exitosas, con lecciones aprendidas de cada tipo de negocio, que motiva a la codificación de nuevas funcionalidades para el sistema. Dado que cada negocio tiene su particularidad, las modificaciones o ampliaciones al sistema deberán contemplarse generalmente, para poder darle uso en diferentes ámbitos.

Los objetivos de este proyecto, como la automatización de pedidos, alta usabilidad y conectividad, han servido para motivar a posibles interesados en la herramienta como en parámetros diferenciadores con respecto a herramientas similares que se consiguen en el mercado.

Es importante en el largo plazo, corroborar su operatividad, ya que al ser un programa que funciona en un entorno web, es susceptible a actualizaciones por parte del navegador web lo que puede interferir en el funcionamiento de la herramienta.

Bibliografía y referencias

Alvarez, J. A. (2012). *Estructura de computadores.* Argentina: Alfaomega.

Anonimo. (9 de 8 de 2015). *Point of sale*. Obtenido de www.wikipedia.com: https://en.wikipedia.org/wiki/Point_of_sale

Ayers, C. (01 de 07 de 2016). *Conducting business analysis and developing requirements*. Obtenido de Pluralsight: https://app.pluralsight.com/library/courses/conducting-business-analysis-developing-requirements/table-of-contents

Ayers, C. (01 de 05 de 2016). *Monitoring requirements and evaluating business analysis*. Obtenido de Pluralsight: https://app.pluralsight.com/library/courses/monitoring-requirements-evaluating-business-analysis-solutions/table-of-contents

Cleland, D. (2016). *Gestion de proyectos.* Mexico: Mc Graw Hill.

contributors, W. (30 de 4 de 2017). *https://www.wikipedia.org/*. Obtenido de https://www.wikipedia.org/: https://en.wikipedia.org/wiki/Point_of_sale

Corporation, M. (01 de 01 de 2009). *msdn*. Obtenido de What is software architecture: https://msdn.microsoft.com/en-us/library/ee658098.aspx

Cuadra, D., Castro, E., Iglesias, A., Martinez, P., Calle, J., de Pablo, C., . . . Al-Jumaily, H. (2008). *Desarrollo de bases de datos: Casos practicos desde el analisis a la implementación.* Mexico: Alfaomega Ra-Ma.

Grunbacher, P. (01 de 07 de 2004). *Ercim*. Obtenido de Automated Software Engineering: https://www.ercim.eu/publication/Ercim_News/enw58/intro.html

Halabi, M. (01 de 06 de 2015). *Software Engineering Essential Course*. Obtenido de Pluralsight: https://app.pluralsight.com/library/courses/software-engineering-essentials/table-of-contents

Kenneth E, K., & Julie E., K. (2011). *Analisis y diseño de sistemas.* Mexico: Pearson.

Krasowski, M. (01 de 03 de 2014). *Best practices for project estimation course*. Obtenido de Pluralsight: https://app.pluralsight.com/library/courses/project-estimation-best-practices/table-of-contents

LLC, D. (01 de 01 de 2017). *dictionary*. Obtenido de connectivity: http://dictionary.com/browse/connectivity

Mancilla Herrera, A., Ebratt Gomez, R., & Capacho Portilla, J. (2015). *Diseño y construcción de algoritmos.* Barranquilla: Universidad del norte, editorial.

Pantaleo, G., & Rinaudo, L. (2015). *Ingeniería de Software.* Buenos Aires: Alfaomega.

Schach, S. R. (2006). *Ingenieria de software clasica y orientada a objetos.* Mexico: Mc Graw Hill.

Sommerville, I. (2011). *Ingenieria de Software.* Mexico: Pearson.

Starr, D. (01 de 01 de 2012). *Scrum fundamentals course*. Obtenido de Pluralsight: https://app.pluralsight.com/library/courses/scrum-fundamentals/table-of-contents

Thomas, P., Galdamez, N., Delia, L., Cristina, F., Dapoto, S., Tinetti, F., . . . De Giusti, A. (01 de 01 de 2013). *Ingenieria de software en el desarrollo de aplicaciones moviles*. Obtenido de http://sedici.unlp.edu.ar: http://sedici.unlp.edu.ar/bitstream/handle/10915/27247/Documento_completo.pdf?sequence=1