



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ESCUELA DE INGENIERÍA EN SISTEMAS

**“ESTUDIO COMPARATIVO DE LOS PATRONES PARA
INTERFACES DE USUARIO MVVM Y MVC. APLICADO AL
DESARROLLO DEL SITIO DE GESTIÓN DE VENTAS PARA
VIDRIALUM”**

TESIS DE GRADO

Previo obtención del título de:

INGENIERO EN SISTEMAS INFORMÁTICOS

Presentado Por:

JAVIER ALEJANDRO VALLEJO BERMEO

RIOBAMBA-ECUADOR

2014

AGRADECIMIENTO

Al culminar una de las metas propuestas en mi vida quiero agradecer a mi familia, por el apoyo brindado en mi formación académica siendo guías, consejeros y motivadores, creyendo en mí a toda costa.

A mis tutores la Ing. Gloria Arcos y el Ing. Jorge Menéndez, gracias a su paciencia, dedicación y guía para culminar este trabajo de investigación necesario para terminar la carrera.

Javier Vallejo

DEDICATORIA

A mi madre que fue un apoyo incondicional en cada aspecto de mi vida siendo consejera amiga y un apoyo en toda mi carrera.

A mi esposa que juntos pasamos buenos y malos momentos en nuestra vida y ella ha soportado todo el esfuerzo de mis estudios apoyándome sin condiciones.

A mi hija que es el motor de mi vida, la fuerza para empezar un nuevo día con armonía es la persona por la que me levanto alegre cada día.

A mi padre amigo incondicional, inseparable fuerte y estricto que gracias a sus enseñanzas me llevo a alcanzar metas que me parecían imposibles.

A estas cuatro personas les dedico este trabajo de investigación

Javier Vallejo

FIRMAS DE RESPONSABILIDAD Y NOTA DEL TRIBUNAL

NOMBRES	FIRMAS	FECHA
ING. GONZALO SAMANIEGO DECANO DE LA FACULTAD INFORMÁTICA Y ELECTRÓNICA	_____	_____
DR. JULIO SANTILLAN DIRECTOR DE LA ESCUELA DE INGENIERÍA EN SISTEMAS	_____	_____
ING. GLORIA ARCOS DIRECTORA DE TESIS	_____	_____
ING. JORGE MENÉNDEZ MIEMBRO DE TESIS	_____	_____
DIRECTOR DEL CENTRO DE DOCUMENTACIÓN	_____	_____

NOTA: _____

RESPONSABILIDADES DEL AUTOR

Yo, Javier Alejandro Vallejo Bermeo, soy el responsable de las ideas, doctrinas y resultados expuestos en esta tesis y el patrimonio intelectual de la misma que pertenece a la Escuela Superior Politécnica de Chimborazo.

Javier Alejandro Vallejo Bermeo

ÍNDICE DE ABREVIATURAS

API.- Application Programming Interface

CPU.- Central Process Unit

DTO: Data Transfer Object

GoF: Gans of Four.

HMVC: Hierarchical Model View Controler

MVA: Modelo Vista Adaptador.

MVC.- Model View Controler

MVP: Modelo Vista Presentador.

MVP: Most Value Professional.

MVVM.- Model View ViewModel

POCO: Plain Old CLR Objects.

RAM.- Random Access Memory

RIA.- Rich Internet Application.

WPF: Windows Presentation Fundation.

XAML.- Extensible Application Markup Language

XML.- Extensible Markup Language

CONTENIDO

AGRADECIMIENTO

DEDICATORIA

FIRMAS DE RESPONSABILIDAD Y NOTA DEL TRIBUNAL

RESPONSABILIDADES DEL AUTOR

ÍNDICE DE ABREVIATURAS

ÍNDICE DE FIGURAS

ÍNDICE TABLAS

CAPÍTULO I

1	MARCO REFERENCIAL.....	- 16 -
1.1	ANTECEDENTES.....	- 16 -
1.2	JUSTIFICACIÓN DEL PROYECTO DE TESIS.....	- 19 -
1.2.1	JUSTIFICACIÓN TEÓRICA.....	- 19 -
1.2.2	JUSTIFICACIÓN PRÁCTICA	- 21 -
1.3	OBJETIVOS DEL PROYECTO DE TESIS.....	- 22 -
1.3.1	OBJETIVO GENERAL.....	- 22 -
1.3.2	OBJETIVO ESPECÍFICOS.....	- 22 -
1.4	HIPÓTESIS DEL PROYECTO DE TESIS	- 23 -
1.5	MÉTODOS Y TÉCNICAS	- 23 -
1.5.1	MÉTODOS	- 23 -

1.5.2	TÉCNICAS	- 24 -
CAPÍTULO II.....		- 25 -
2	ANÁLISIS DE LOS PATRONES DE DISEÑO MVC Y MVVM.....	- 25 -
2.1	PATRONES DE DISEÑO	- 26 -
2.1.1	HISTORIA.....	- 27 -
2.1.2	OBJETIVOS DE LOS PATRONES DE DISEÑO.....	- 28 -
2.1.3	ELEMENTOS DE LOS PATRONES DE DISEÑO	- 28 -
2.1.4	CATEGORÍA DE PATRONES	- 29 -
2.1.5	TIPO DE PATRONES DE DISEÑO.....	- 30 -
2.2	MODELO VISTA CONTROLADOR.....	- 33 -
2.2.1	HISTORIA.....	- 34 -
2.2.2	EL MODELO	- 35 -
2.2.3	LA VISTA	- 36 -
2.2.4	EL CONTROLADOR	- 36 -
2.2.5	¿CÓMO FUNCIONA EL PATRÓN MVC?	- 37 -
2.2.6	VENTAJAS Y DESVENTAJAS DEL PATRÓN MVC.....	- 39 -
2.3	MODELO VISTA MODELO DE VISTA.....	- 40 -
2.3.1	HISTORIA.....	- 42 -
2.3.2	EI MODELO.....	- 43 -
2.3.3	LA VISTA	- 44 -
2.3.4	EL VIEWMODEL.....	- 44 -

2.3.5	¿CÓMO FUNCIONA EL PATRÓN MVVM?	- 45 -
2.3.6	VENTAJAS Y DESVENTAJAS DEL PATRÓN MVVM.....	- 46 -
2.4	DIFERENCIA ENTRE EL PATRÓN MVC Y MVVM	- 47 -
 CAPÍTULO III		
3	ESTUDIO COMPARATIVO ENTRE LOS PATRONES DE DISEÑO MVC Y MVVM.....	- 49 -
3.1	PROTOTIPOS.....	- 50 -
3.2	DETERMINACIÓN DE LOS PARÁMETROS DE COMPARACIÓN....	- 51 -
3.2.1	VARIABLE 1: RENDIMIENTO	- 51 -
3.2.2	VARIABLE 2: PRODUCTIVIDAD	- 53 -
3.3	HERRAMIENTAS PARA PRUEBAS DE RENDIMIENTO Y PRODUCTIVIDAD.....	- 54 -
3.4	PRUEBA DE HIPÓTESIS.....	- 55 -
3.5	ANÁLISIS DE LOS RESULTADOS.....	- 57 -
3.5.1	METODOLOGÍA PARA LAS PRUEBAS DE RENDIMIENTO.....	- 57 -
3.5.2	ANÁLISIS DEL PARÁMETRO 1: RENDIMIENTO.....	- 58 -
3.5.3	ANÁLISIS DEL PARÁMETRO 2: PRODUCTIVIDAD.....	- 77 -
3.6	DEMOSTRACIÓN DE LA HIPÓTESIS	- 84 -
 CAPÍTULO IV		
4	SISTEMA AUTOMATIZADO DE INVENTARIO Y FACTURACIÓN.....	88
4.1	CONCEPTO.....	- 89 -

4.1.1	INFORMACIÓN GENERAL.....	- 90 -
4.1.2	DESCRIPCIÓN DEL PRODUCTO.....	- 91 -
4.1.3	FUNCIONES DEL PRODUCTO.....	- 91 -
4.1.4	CARACTERÍSTICAS DE USUARIO.....	- 92 -
4.1.5	SUPUESTOS Y DEPENDENCIAS.....	- 92 -
4.2	ESPECULACIÓN.....	- 92 -
4.2.1	REQUERIMIENTOS FUNCIONALES.....	- 93 -
4.2.2	REQUERIMIENTO DE RENDIMIENTO	- 94 -
4.2.3	LIMITACIONES DE DISEÑO	- 95 -
4.3	EXPLORACIÓN.....	- 95 -
4.3.1	PRODUCT BACKLOG	- 95 -
4.3.2	SPRINT 1.....	- 97 -
4.3.3	SPRINT 2.....	- 97 -
4.3.4	SPRINT 3.....	- 98 -
4.3.5	SPRINT 4.....	- 98 -
4.4	REVISIÓN	- 98 -
4.4.1	ESPECIFICACIONES DE PERSONAS Y EQUIPAMIENTO PARA EL PROYECTO	- 99 -
4.4.2	DESARROLLO DE LA BASE DE DATOS DEL SISTEMA SADIF.....	- 99 -
4.4.3	PROGRAMACIÓN DEL SISTEMA SADIF.....	- 101 -
4.4.4	CAPACITACIÓN SOBRE EL USO DEL SISTEMA.....	- 111 -

CONCLUSIONES

RECOMENDACIONES

RESUMEN

SUMMARY

GLOSARIO TÉRMINOS

ANEXOS PRUEBAS RENDIMIENTO PRODUCTIVIDAD

ANEXO SPRINT

ANEXOS CASO DE USOS

ANEXO DIAGRAMAS

BIBLIOGRAFÍA

ÍNDICE DE FIGURAS

FIGURA II-1 DIAGRAMA MODELO VISTA CONTROLADOR	- 33 -
FIGURA II-2 FUNCIONAMIENTO DEL PATRÓN DE DISEÑO MVC	- 38 -
FIGURA II-3 Patrón MVVM (MODEL VIEW VIEWMODEL)	- 42 -
FIGURA II-4 DISEÑO DE UNA APLICACIÓN USANDO MVVM	- 45 -
FIGURA II-5 PATRONES MVC Y MVVM.....	- 47 -
FIGURA III-6 RESULTADO PARÁMETRO RENDIMIENTO	- 76 -
FIGURA III-7 RESULTADO EN PORCENTAJE DEL PARÁMETRO DE RENDIMIENTO.....	- 76 -
FIGURA III-8 RESULTADO ÍNDICES DE PRODUCTIVIDAD	- 83 -
FIGURA III-9 PORCENTAJE DE CUMPLIMIENTO PARÁMETRO PRODUCTIVIDAD.....	- 83 -
FIGURA III-10 NIVEL DE CUMPLIMIENTO DE LA HIPÓTESIS.....	- 87 -
FIGURA IV-11 PROCESO SCRUM.....	- 89 -
FIGURA IV-12 ORGANIGRAMA VIDRIALUM.....	- 90 -
FIGURA IV-13 PERSONAL Y EQUIPO PROYECTO SADIF	- 99 -

ÍNDICE TABLAS

TABLA II-I DIFERENCIAS ENTRE LOS PATRONES MVC Y MVVM.....	- 48 -
TABLA III-I PARÁMETROS DE RENDIMIENTO	- 52 -
TABLA III-II INDICADORES DE LOS PARÁMETROS DE RENDIMIENTO ...	- 52 -
TABLA III-III PARÁMETROS DE PRODUCTIVIDAD.....	- 53 -
TABLA III-IV ÍNDICES DE LOS PARÁMETROS DE PRODUCTIVIDAD.....	- 53 -
TABLA III-V HERRAMIENTAS PARA REALIZAR LAS PRUEBAS DE RENDIMIENTO.....	- 54 -
TABLA III-VI DISEÑO DEL EXPERIMENTO.....	- 56 -
TABLA III-VII NIVEL DE CUMPLIMIENTO DEL RENDIMIENTO.....	- 60 -
TABLA III-VIII ESTADÍSTICA DESCRIPTIVA PROMEDIO DE TIEMPO DE DESCARGA DE LA PÁGINA	- 63 -
TABLA III-IX ESTADÍSTICOS DE CONTRASTE PROMEDIO DE TIEMPO DE DESCARGA DE LA PÁGINA	- 63 -
TABLA III-X CALIFICACIÓN POR PATRÓN PARA EL PROMEDIO DE TIEMPO DE LA PAGINA.....	- 64 -
TABLA III-XI ESTADÍSTICA DESCRIPTIVA PARA PAGINAS POR SEGUNDO ...	- 65 -
TABLA III-XII ESTADÍSTICOS DE CONTRASTE PAGINAS POR SEGUNDO	- 65 -
TABLA III-XIII CALIFICACIÓN POR PATRÓN DEL INDICADOR PÁGINA POR SEGUNDO	- 66 -
TABLA III-XIV RESUMEN RESULTADOS DE LA MEDICIÓN DEL INDICADOR SOLICITUD/SEGUNDO	- 67 -
TABLA III-XV ESTADÍSTICOS DE CONTRASTE SOLICITUD/SEGUNDO....	- 67 -

TABLA III-XVI CALIFICACIÓN POR PATRÓN DEL INDICADOR SOLICITUD/SEGUNDO	- 68 -
TABLA III-XVII ESTADÍSTICA DESCRIPTIVA DEL INDICADOR PROMEDIO TIEMPO DE RESPUESTA.....	- 69 -
TABLA III-XVIII ESTADÍSTICOS DE CONTRASTE PROMEDIO TIEMPO DE RESPUESTA.....	- 69 -
TABLA III-XIX CALIFICACIÓN POR PATRÓN DEL INDICADOR PÁGINA POR SEGUNDO	- 70 -
TABLA III-XX ESTADÍSTICA DESCRIPTIVA DEL INDICADOR PORCENTAJE DEL TIEMPO DEL PROCESADOR.....	- 71 -
TABLA III-XXI ESTADÍSTICOS DE CONTRASTE PORCENTAJE DEL TIEMPO DEL PROCESADOR	- 72 -
TABLA III-XXII CALIFICACIÓN POR PATRÓN DEL INDICADOR PÁGINA POR SEGUNDO	- 73 -
TABLA III-XXIII RESUMEN RESULTADOS DE LA MEDICIÓN DEL INDICADOR MBYTE DISPONIBLES	- 73 -
TABLA III-XXIV ESTADÍSTICOS DE CONTRASTE MBYTE DISPONIBLES	- 74 -
TABLA III-XXV CALIFICACIÓN POR PATRÓN DEL INDICADOR PÁGINA POR SEGUNDO	- 75 -
TABLA III-XXVI NIVEL DE CUMPLIMIENTO DE PRODUCTIVIDAD.....	- 78 -
TABLA III-XXVII RESULTADO DE LA MEDICIÓN DE LÍNEAS DE CÓDIGO-	78
-	
TABLA III-XXVIII CALIFICACIÓN POR PATRÓN DEL INDICADOR LÍNEAS DE CÓDIGO.....	- 79 -

TABLA III-XXIX RESULTADO DE LA MEDICIÓN DE COMPLEJIDAD	- 79 -
TABLA III-XXX CALIFICACIÓN POR PATRÓN DEL INDICADOR LÍNEAS DE CÓDIGO.....	- 80 -
TABLA III-XXXI RESULTADO DE LA MEDICIÓN DE DISPONIBILIDAD DE INFORMACIÓN	- 81 -
TABLA III-XXXII CALIFICACIÓN POR PATRÓN DEL INDICADOR LÍNEAS DE CÓDIGO.....	- 81 -
TABLA III-XXXIII VALORES DE LOS NIVELES DE CUMPLIMIENTO DE CADA PARÁMETRO E INDICADOR PARA MVC Y MVVM.....	- 86 -
TABLA IV-I PRODUCT BACKLOG PROYECTO SADIF.....	- 96 -
TABLA IV-II SPRINT 1	134
TABLA IV-III SPRINT 2.....	140
TABLA IV-IV SPRINT 3.....	143
TABLA III-V SPRINT 4	146

CAPÍTULO I

1 MARCO REFERENCIAL

1.1 ANTECEDENTES

En la empresa VIDRIALUM se han presentado ciertos problemas en tiempo, recurso y eficiencia de los procesos, debidos a que algunos trámites se los realiza de manera remota consumiendo muchos recursos de hardware y software que no son adecuados dificultando así las actividades que se realizan en la empresa.

El problema se debe a que poseen un sistema de ventas desarrollado para entornos de escritorio, dicho sistema solo se lo utiliza en la red y para poder utilizarlo debe ser instalado en cada máquina lo que le representa a la empresa altas sumas de dinero que debe ser cancelado al técnico encargado de la instalación del sistema. Por otra parte dicho sistema no se puede acceder directamente desde la sucursal que se encuentra en Guaranda para esto utilizan la conexión remota por lo que poseen una máquina solo para el uso de la conexión remota lo que representa un gasto de recursos innecesarios.

Se pretende desarrollar un sistema web para la gestión de ventas de la empresa VIDRIALUM, este sistema pretende resolver el gasto excesivo por el pago de servicio técnico así como la facilidad de la conexión al sistema vía internet de la sucursal de la empresa. Pero para resolver el problema se deberá tener en cuenta que el diseño de las interfaces de usuario es muy importante, debido a que el usuario interactúa directamente con el sistema, es por esta razón que para brindar un mejor rendimiento y productividad al sistema de gestión de ventas se deberá utilizar un patrón para el desarrollo de interfaces.

La complejidad de un sistema, y los problemas recurrentes que contiene, fortalece para que los desarrolladores organicen su código de tal manera que sea fácil de comprender y solucionar problemas. Se disminuye el caos de un sistema complejo mediante la aplicación de patrones para el diseño de interfaces de usuarios, el uso de estos patrones para la interfaz de usuario nos brindará una aplicación más estructurada, organizada y escalable, es inevitable realizar un estudio de los patrones para interfaces de usuario y tomar la decisión más acertada sobre que patrón es el más adecuado para nuestra aplicación; en el desarrollo de este tema de investigación nos centraremos en dos patrones arquitectónicos: Modelo Vista Controlador y Model View ViewModel.

El sistema de gestión de Ventas utilizará uno de los patrones más utilizados en la actualidad para el desarrollo de interfaces de usuario enriquecidas, como son el Modelo Vista Controlador (MVC) y el Modelo Vista Modelo de Vista (MVVM).

Para saber que patrón es el que mejor se adapta al Sistema de Gestión de Ventas se realizará un estudio comparativo entre el rendimiento y la productividad de los patrones antes mencionados. Es importante tomar en cuenta estos parámetros, y de esta manera

centrarnos en el estudio de los patrones de diseño de interfaces de usuario dando a conocer sus primacías y perjuicios.

La creación de interfaces de usuario de cualquier tipo de tecnología se vuelve más poderosa y es una práctica común para la capa de Presentación hacer más de lo debido. Sin una clara separación de responsabilidades, la capa de interfaz de usuario, mejor conocido como la capa de Presentación, puede llegar a ser muy problemático para la lógica que pertenece a las otras capas de la aplicación.

Es por esta razón que la capa de presentación debe poseer una sub arquitectura que encaje adecuadamente con las otras capas de la aplicación, es por este motivo que se pretende realizar un estudio de los patrones de la capa de presentación más utilizados como son: Modelo Vista Controlador (MVC) y Modelo Vista Modelo de Vista (MVVM) que es la traducción más fiable de “Model View ViewModel”; para poder comprender cuál será el mejor patrón para crear aplicaciones tanto web como de escritorio profesionales.

El Modelo Vista Controlador (MVC) nos presenta un objetivo muy definido que es la separación del código responsable de la representación de los datos en pantalla, del código encargado de la ejecución de la lógica de negocio, para ello el patrón MVC divide en tres objetos básicos el Modelo, la vista y el controlador.

El Modelo Vista Modelo de Vista de sus siglas en inglés (Model View ViewModel), surgió como una implementación específica de los patrones Modelo Vista Presentación (MVP) y Modelo Vista Controlador (MVC) al hacer uso de ciertas capacidades muy potentes de nuevas tecnologías de interfaz de usuario disponible. El objetivo primordial de este patrón es trasladar los datos a la “Vista” para poder presentar y gestionar de la manera más sencilla.

La dificultad más notoria es identificar que patrón de la capa de presentación es la más óptima, segura y fácil de implementar en aplicaciones profesionales, es por esta razón la importancia del estudio comparativo entre los dos patrones. Aun los conceptos y características más básicas de ambos patrones permitirán entender las ventajas y desventajas que posee cada patrón con respecto al otro.

Se aplicará el tema de investigación en la empresa VIDRIALUM al crear un sistema web para la Gestión de Ventas.

El tema de tesis se enmarcará en la comparación de los patrones MVC y MVVM para entornos web en base al rendimiento y productividad lo cual permitirá desarrollar una aplicación web para la gestión de ventas en VIDRIALUM.

Por la parte aplicativa nos limitaremos en los siguientes módulos:

- Gestión de Empleados.
- Gestión de Clientes.
- Gestión de Productos.
- Gestión de Ventas.
- Gestión de Compras.
- Gestión de Proveedores.

1.2 JUSTIFICACIÓN DEL PROYECTO DE TESIS

1.2.1 JUSTIFICACIÓN TEÓRICA

El diseño de la arquitectura de un sistema es el proceso por el cual se define una solución para los requisitos técnicos y operacionales del mismo. Este proceso define qué componentes forman el sistema, cómo se relacionan entre ellos y cómo mediante su interacción lleva acabo la funcionalidad específica, cumpliendo con los criterios de calidad indicados como seguridad, disponibilidad, eficiencia o usabilidad.

El patrón MVVM tiene por objetivo simplificar las tareas de desarrollo y mantenimiento del software escrito a través de la división de ocupaciones.

- **Modelo:** El modelo, dentro de MVVM es el encargado de representar el modelo del negocio, suministrando de esta manera la base necesaria para la manipulación de los datos de la aplicación, además parte del modelo se lo puede usar como clases POCO (Plain Old CLR Objects) para poder usarlas con Entity Framework.
- **Vista:** La vista es la encargada de la parte visual de nuestra aplicación, no se tiene que ocupar en ningún momento en el manejo de datos. En MVVM la vista tiene un rol activo, esto significa que en algún momento la vista recibirá o manejará algún evento (Clic en un botón, alguna tecla presionada, etc.) y tendrá que comunicarse con el modelo, para poder cumplir el requerimiento.
- **ViewModel:** El ViewModel (modelo de vista en español) es el encargado de ser la capa intermedia entre el modelo y la vista, procesando todas las peticiones que tenga la vista hacia el modelo, además de tener que ocuparse de manejar las reglas del negocio, la comunicación con aplicaciones externas o consumir datos desde alguna fuente.

VENTAJAS

- MVVM facilita el desarrollo en paralelo de las interfaces de usuario y la programación.
- Abstrae las vistas y reduce la cantidad de la lógica de negocio requerida en el code behind.
- MVVM es más fácil realizar pruebas unitarias.

El MVC es el patrón más usado en la ingeniería del software. Este patrón nos separa el modelos de datos, el modelo de la capa de presentación (vista) y de la parte de control.

- **Modelo:** es el encargado de representar los datos de la aplicación y la lógica para recuperar y persistir datos
- **Vista:** es la parte encargada de la parte visual de nuestra aplicación.
- **Controlador:** es el responsable de controlar la interacción del usuario y luego actualizar el modelo o la vista

Es por esta razón que es necesario el estudio comparativo entre los dos patrones más importantes hoy en día, para poder comprender cuál será la arquitectura más apta para el desarrollo de aplicaciones de software profesional.

1.2.2 JUSTIFICACIÓN PRÁCTICA

Para la ejecución del estudio de los patrones de la capa de presentación se realizará ambientes de prueba que servirán como prototipo del sistema de gestión de ventas para la empresa VIDRIALUM.

Dichos ambientes de pruebas se codificará en ambos patrones tanto para Modelo Vista Control, como para el Modelo Vista Modelo de Vista, las mismas que se implementarán de la siguiente manera:

- La arquitectura Modelo Vista Controlador utilizará asp.net con MVC 4.
- La arquitectura Modelo Vista Modelo de Vista utilizará .Net con LigthSwitch.

Dichos ambientes de prueba contarán con los módulos de gestión de empleados, clientes, proveedores, productos facturas de compras y ventas. Con su respectivo módulo de autenticación de usuarios.

Los ambientes nos facilitarán la comparación entre las arquitecturas MVVM y MVC comparando la productividad, y el rendimiento.

En la empresa VIDRIALUM se han presentado ciertos problemas en tiempo, recurso y eficiencia de los procesos, debidos a que algunos trámites se los realiza de manera remota consumiendo muchos recursos de hardware y software que no son adecuados dificultando así las actividades que se realizan en la empresa.

El problema se deben a que poseen un sistema de ventas desarrollado para entornos de escritorio, dicho sistema solo se lo utiliza en la red y para poder utilizarlo debe estar instalado en cada máquina lo que representa un costo de pago por servicio técnico.

Para resolver los problemas antes mencionados se realizará el sistema web para la gestión de ventas que se pueda utilizar mediante Intranet e Internet, el mismo que permitirá disminuir el consumo de hardware y el uso de software inadecuado.

1.3 OBJETIVOS DEL PROYECTO DE TESIS

1.3.1 OBJETIVO GENERAL

Realizar un estudio comparativo entre los patrones para interfaces de usuario Modelo Vista Controlador y Modelo Vista Modelo de Vista y obtener el patrón con mejor rendimiento y productividad para el desarrollo del sistema de Gestión de Ventas para VIDRIALUM.

1.3.2 OBJETIVO ESPECÍFICOS

- Analizar los patrones MVC y MVVM para conocer las semejanzas y diferencias entre estos patrones.
- Definir variables e indicadores para los parámetros: rendimiento y productividad.

- Diseñar ambientes de prueba en cada una de los patrones para definir la evolución de los parámetros definidos.
- Escoger el patrón más adecuado para el desarrollo de la aplicación web.
- Desarrollar el sistema de gestión de ventas para la empresa VIDRIALUM.

1.4 HIPÓTESIS DEL PROYECTO DE TESIS

La arquitectura Modelo Vista Modelo de Vista presenta mejores condiciones en rendimiento y productividad para el desarrollo de aplicaciones de software profesionales.

1.5 MÉTODOS Y TÉCNICAS

Los métodos y técnicas describen la manera, los procesos, los pasos que se utilizará en esta investigación.

1.5.1 MÉTODOS

Para el estudio de los patrones MVC y MVVM como el estudio se aplicará el método científico el cual consta de las siguientes actividades.

- Planteamiento del problema.
- Formulación de la Hipótesis.
- Levantamiento de la información.
- Análisis e interpretación de Resultados.
- Comprobación de hipótesis.
- Difusión de resultados

1.5.2 TÉCNICAS

Las técnicas que se utilizará para el desarrollo de esta investigación son:

La observación participativa esta técnica permite participar en la construcción del tema en estudio las relaciones que se presentan entre la práctica integrada y la construcción de significados, participando del proceso de construcción descubriendo el sentido, la dinámica y los procesos de los acontecimientos que viven los protagonistas en el medio en que se desarrolla. Juego de rol esta técnica permitirá colocarse como actor para el desarrollo de las pruebas del sistema a realizarse.

CAPÍTULO II

2 ANÁLISIS DE LOS PATRONES DE DISEÑO MVC Y MVVM

La Capa de Presentación incluye diferentes elementos como Vistas, Controladores, Modelos, etc. Sin embargo antes de incorporarse en aspectos de Arquitectura, vamos a comenzar con aspectos importantes de la propia naturaleza de esta capa.

El objetivo principal de la capa de presentación es que mediante interfaces de usuario con sus respectivos procesos y validaciones presente los conceptos del negocio. Al usuario final, al mismo que llamaremos cliente, lo más importante es la aplicación en sí, de nada vale diseñar la mejor arquitectura si no logramos que el cliente utilice de la manera más sencilla la aplicación.

Pensar que la Capa de Presentación necesita una “arquitectura” durante todas las partes del ciclo de desarrollo reduce considerablemente el tiempo de desarrollo, para ello se necesita conocer qué “arquitecturas” para la Capa de Presentación, son las más eficientes. A éstas “arquitecturas” se la conoce como Patrones de diseño.

Existen algunos patrones de diseño conocidos que se pueden utilizar para el diseño de la Capa de Presentación como pueden ser Modelo Vista Controlador (MVC), Modelo

Vista Presentación (MVP), Modelo Vista Modelo de Vista (MVVM) y otras variantes. Al día de hoy, estos patrones se usan para separar los conceptos entre la interfaz y la lógica de presentación, pero en su momento fueron concebidos para separar la capa de presentación de la capa de lógica de negocio, ya que las tecnologías de presentación no solían abstraerse de la lógica de negocio.

2.1 PATRONES DE DISEÑO

Uno de los conceptos que describen de manera más efectiva ¿qué es un patrón de diseño? es el siguiente:

“Los patrones de diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interfaces de usuario”. (1)

Un patrón de diseño resulta ser una solución a un problema que se presenta en la Capa de Presentación, es decir, brinda una solución ya probada y documentada a problemas de desarrollo de software referentes a las interfaces de usuario.

Un desarrollador se ve obligado a utilizar código ya escrito y modificarlo las veces que sea necesario, empieza a pensar la manera de realizar código de manera que al trasladar a otro sitio sea más fácil utilizarlo. El programador empieza a inventar diferentes maneras de programar de forma que dicho código sea lo más versátil y configurable como para no volver a modificarlo.

Los patrones de diseño son soluciones a problemas con el diseño de la Capa de Presentación y hacen que éste sea más flexible, elegante y reutilizable.

2.1.1 HISTORIA

Según las definiciones de Wikipedia se encontró que la historia, sus objetivos, elementos y categorías son: (2)

En 1979 el arquitecto Christopher Alexander aportó al mundo de la arquitectura el libro “The Timeless Way of Building”; en él proponía el aprendizaje y uso de una serie de patrones para la construcción de edificios de una mayor calidad.

En palabras de este autor, "Cada patrón describe un problema que ocurre infinidad de veces en nuestro entorno, así como la solución al mismo, de tal modo que podemos utilizar esta solución un millón de veces más adelante sin tener que volver a pensarla otra vez."

Los patrones que Christopher Alexander y sus colegas definieron, publicados en un volumen denominado “A Pattern Language”, son un intento de formalizar y plasmar de una forma práctica generaciones de conocimiento arquitectónico. Los patrones no son principios abstractos que requieran su redescubrimiento para obtener una aplicación satisfactoria, ni son específicos a una situación particular o cultural; son algo intermedio. Un patrón define una posible solución correcta para un problema de diseño dentro de un contexto dado, describiendo las cualidades invariantes de todas las soluciones.

Más tarde, en 1987, Ward Cunningham y Kent Beck usaron varias ideas de Alexander para desarrollar cinco patrones de interacción hombre-ordenador (HCI) y publicaron un artículo en OOPSLA-87 titulado “Using Pattern Languages for OO Programs”.

No obstante, no fue hasta principios de la década de 1990 cuando los patrones de diseño tuvieron un gran éxito en el mundo de la informática a partir de la publicación del libro “Design Patterns” escrito por el grupo “Gang of Four (**GoF**)” Banda de los cuatro,

compuesto por Erich Gamma, Richard Helm, Ralph Johnson y John Vlissides, en el que se recogían 23 patrones de diseño comunes.

2.1.2 OBJETIVOS DE LOS PATRONES DE DISEÑO

Los patrones de diseño tienen por objetivo:

- Proporcionar catálogos de elementos reusables en el diseño de sistemas de software.
- Evitar la reiteración en la búsqueda de soluciones de problemas ya conocidos y solucionados anteriormente.
- Describir un vocabulario común entre diseñadores.
- Estandarizar el modo en que se realiza el diseño.
- Facilitar el aprendizaje de las nuevas generaciones de diseñadores condensando conocimiento ya existente.

2.1.3 ELEMENTOS DE LOS PATRONES DE DISEÑO

Para describir un patrón se utiliza plantillas, de forma que se expresen de manera uniforme y pueden constituir efectivamente un medio de comunicación uniforme entre diseñadores.

En general un patrón tiene cuatro elementos esenciales:

- **Nombre de Patrón:** permite describir en una o dos palabras, el problema de diseño junto con sus soluciones y consecuencias.
- **Clasificación del Patrón:** permite determinar a qué categoría pertenece el patrón siendo estos: creacional, estructural o de comportamiento.
- **Intención:** Problema que se pretende resolver con el patrón.
- **Motivación:** Escenario de ejemplo para la aplicación del patrón.
- **Aplicabilidad:** usos comunes y criterios de aplicabilidad del patrón.

- **Estructura:** Diagramas de clases oportunos para describir las clases que intervienen en el patrón.
- **Participantes:** Enumeración y descripción de las entidades abstractas (y sus roles) que participan en el patrón.
- **Colaboraciones:** Explicación de las interrelaciones que se dan entre los participantes.
- **Consecuencias:** Consecuencias positivas y negativas en el diseño derivadas de la aplicación del patrón.
- **Implementación:** Técnicas o comentarios oportunos de cara a la implementación del patrón.
- **Código de ejemplo:** Código fuente ejemplo de implementación del patrón.
- **Usos conocidos:** Ejemplos de sistemas reales que usan el patrón.
- **Patrones relacionados:** Referencias cruzadas con otros patrones.

2.1.4 CATEGORÍA DE PATRONES

Según la escala y nivel de abstracción los patrones se clasifican:

- **Patrones de arquitectura:** Aquellos que expresan un esquema organizativo estructural fundamental para sistemas de software.
- **Patrones de diseño:** Aquellos que expresan esquemas para definir estructuras de diseño con las que construir sistemas de software.
- **Dialectos:** patrones de bajo nivel específicos para un lenguaje de programación o entornos en concreto.
- **Interacción:** Son patrones que nos permiten el diseño de interfaces web.

2.1.5 TIPO DE PATRONES DE DISEÑO

La clasificación más adecuada es la presentada por GoF (Banda de Cuatro) en su libro “Design Patterns” y determina que los patrones se clasifican: (1)

- **Patrones Creacionales:** Abstraen el proceso de creación de instancias de objeto. Ayudan hacer un sistema independiente de cómo se crean, se componen y se representa sus objetos. Dentro de los patrones creacionales encontramos a.
 - **Abstract Factory:** Aborda el problema de la creación de familias de objetos, ofrece una interfaz para la creación de familias de productos relacionados o dependientes sin especificar las clases concretas a las que pertenecen. Este patrón se conoce también como Kit o toolkit.
 - **Factory Method:** Se le conoce como Constructor Virtual, su nombre en español es Método de Fábrica, este patrón consiste en utilizar una clase constructora abstracta con unos cuantos métodos definidos y otro(s) abstracto(s): es dedicado a la construcción de objetos de un subtipo de un tipo determinado.
 - **Prototype:** sirve para crear un duplicado de un objeto, clonando, para ello, una instancia de ese objeto que ya haya sido creada. Para ello, el patrón tiene que especificar el tipo de objeto que quiere clonar, creando así un 'prototipo' de esa instancia. Este tipo o clase de objetos deberá contener en su interfaz el procedimiento que permita solicitar esa copia, siendo desarrollado luego por las clases concretas del patrón que deseen crear ese clon.
 - **Singleton:** busca restringir la creación de objetos pertenecientes a una clase o el valor de un tipo a un único objeto. Su intención es garantizar

que una clase sólo sea instanciada una vez y, además, proporcionar un único punto de acceso global a la misma. Esto lo consigue gracias a que es la propia clase la responsable de crear esa única instancia, (declarando el constructor de la clase como privado) y a que se permite el acceso global a dicha instancia mediante un método de clase.

- **Patrones Estructurales:** Tratan con la composición de clases u objetos. Se ocupan de como las clases y objetos son utilizados para componer estructuras de mayor tamaño. Dentro de los patrones estructurales encontramos a:
 - **Adapter:** El patrón Adapter convierte la interfaz de una clase en la que otra necesita, permitiendo que clases con interfaces incompatibles trabajen juntas.
 - **Bridge:** también conocido como Handle/Body, es una técnica usada en programación para desacoplar una abstracción de su implementación, de manera que ambas puedan ser modificadas independientemente sin necesidad de alterar por ello la otra. Esto es, se desacopla una abstracción de su implementación para que puedan variar independientemente.
 - **Composite:** Permite tratar objetos compuestos como si de un simple se tratase.
 - **Decorator:** Añade funcionalidad a una clase dinámicamente.
 - **Facade:** Provee de una interfaz unificada simple para acceder a una interfaz o grupo de interfaces de un subsistema.
 - **Flyweight:** Reduce la redundancia cuando gran cantidad de objetos poseen idéntica información.
 - **Proxy:** Mantiene un representante de un objeto.

- **Módulo:** Agrupa varios elementos relacionados, como clases, singletons, y métodos, utilizados globalmente, en una entidad única.
- **Patrones de Comportamiento:** Caracterizan el modo en que las clases y objetos interactúan y se reparten la responsabilidad. Atañen a los algoritmos y a la asignación de responsabilidad entre objetos. Entre los que se tienen:
 - **Chain of Responsibility:** Permite establecer la línea que deben llevar los mensajes para que los objetos realicen la tarea indicada.
 - **Command:** Encapsula una operación en un objeto, permitiendo ejecutar dicha operación sin necesidad de conocer el contenido de la misma.
 - **Interpreter:** Dado un lenguaje, define una gramática para dicho lenguaje, así como las herramientas necesarias para interpretarlo.
 - **Iterator:** Permite realizar recorridos sobre objetos compuestos independientemente de la implementación de estos.
 - **Mediator:** Define un objeto que coordine la comunicación entre objetos de distintas clases, pero que funcionan como un conjunto.
 - **Memento:** Permite volver a estados anteriores del sistema.
 - **Observer:** Define una dependencia de uno a muchos entre objetos, de forma que cuando un objeto cambie de estado se notifique y actualicen automáticamente todos los objetos que dependen de él.
 - **State:** Permite que un objeto modifique su comportamiento cada vez que cambie su estado interno.
 - **Strategy:** Permite disponer de varios métodos para resolver un problema y elegir cuál utilizar en tiempo de ejecución.

- **Template Method:** Define en una operación el esqueleto de un algoritmo, delegando en las subclasses algunos de sus pasos, esto permite que las subclasses redefinan ciertos pasos de un algoritmo sin cambiar su estructura.
- **Visitor:** Permite definir nuevas operaciones sobre una jerarquía de clases sin modificar las clases sobre las que opera.

2.2 MODELO VISTA CONTROLADOR

El Modelo Vista Controlador (MVC) es un patrón de arquitectura de software que separa los datos y la lógica de presentación de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones. Para ello MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado define componentes para la representación de la información, y por otro lado para la interacción del usuario (3). Este patrón de diseño se basa en las ideas de reutilización de código y la separación de conceptos, características que buscan facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento. (4)

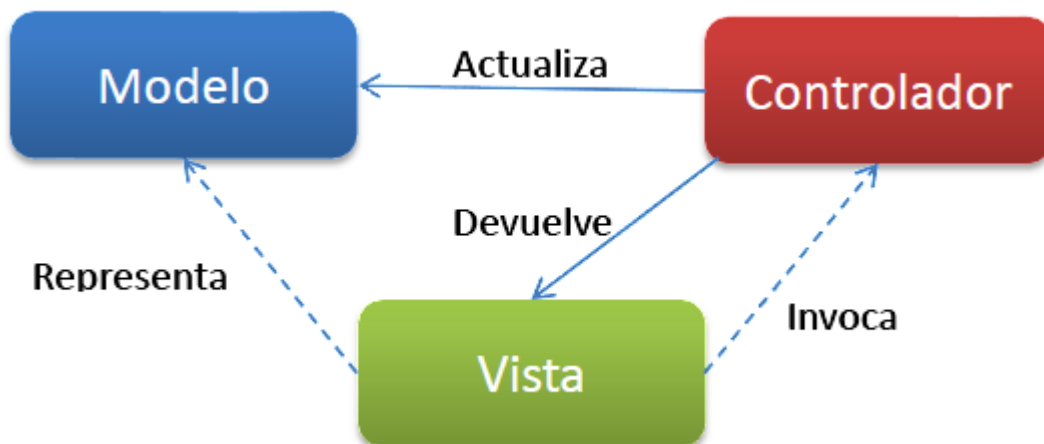


FIGURA II-1 DIAGRAMA MODELO VISTA CONTROLADOR

2.2.1 HISTORIA

Wikipedia indica (5):

El patrón MVC fue una de las primeras ideas en el campo de las interfaces gráficas de usuario y uno de los primeros trabajos en describir e implementar aplicaciones software en términos de sus diferentes funciones.

MVC fue introducido por Trygve Reenskaug en Smalltalk-76 durante su visita a Xerox Parc en los años 70 y, seguidamente, en los años 80, Jim Althoff y otros implementaron una versión de MVC para la biblioteca de clases de Smalltalk-80. Sólo más tarde, en 1988, MVC se expresó como un concepto general en un artículo.

En esta primera definición de MVC el **controlador** se definía como "*el módulo que se ocupa de la entrada*" (de forma similar a como la **vista** "*se ocupa de la salida*"). Esta definición no tiene cabida en las aplicaciones modernas en las que esta funcionalidad es asumida por una combinación de la 'vista' y algún framework moderno para desarrollo. El 'controlador', en las aplicaciones modernas de la década de 2000, es un módulo o una sección intermedia de código, que hace de intermediario de la comunicación entre el 'modelo' y la 'vista', y unifica la validación (utilizando llamadas directas o el "*observer*" para desacoplar el 'modelo' de la 'vista' en el 'modelo' activo).

Algunos aspectos del patrón MVC han evolucionado dando lugar a ciertas variantes del concepto original, ya que "*las partes del MVC clásico realmente no tienen sentido para los clientes ricos actuales*":

- HMVC (MVC Jerárquico)
- MVA (Modelo-Vista-Adaptador)
- MVP (Modelo-Vista-Presentador)
- MVVM (Modelo-Vista Vista-Modelo)

2.2.2 EL MODELO

Según Juan José Estruch indica que (6) :

El modelo es el conjunto de clases encargado de representar la información con que trabaja el usuario. Es muy importante entender que estas clases pueden ser clases de dominio, DTO. (Data Transfer Object) o ViewModel (Entendiendo como ViewModel una clase que almacena los datos que representa una determinada vista, no un ViewModel como lo entenderemos en el modelo MVVM). La opción a escoger depende de la situación:

- Clases de dominio: El modelo de la vista se corresponde fuertemente con una clase de dominio y no estamos empleando DTO. en la aplicación. Por ejemplo en una acción de editar una entidad.
- DTO.: El modelo de la vista se corresponde fuertemente con los datos de un DTO. que estamos utilizando en nuestra aplicación.
- ViewModel: Los datos que necesita la vista no se corresponden directamente ni con un DTO. ni con una clase de dominio de nuestra aplicación. Este ViewModel almacena esa pequeña información adicional y probablemente incluye pequeños métodos de consulta (Propiedades) que simplifican la descripción de la vista. Por ejemplo: Un listado paginado de un conjunto de datos, donde el ViewModel se compone de los datos de la página y de propiedades como el número de página, su tamaño, el número total de páginas, etc.

2.2.3 LA VISTA

Las vistas son las delegadas de representar gráficamente el modelo y de brindar las operaciones de los controladores para que el usuario pueda interactuar con el sistema diseñado. En el caso de una aplicación web la vista es una página HTML con contenido dinámico sobre el cual puede realizar operaciones

Hay que tener muy claro que una vista no debe invocar ningún método que provoque un cambio de estado en el modelo o llamar a algún método que requiera parámetros, es decir, debe limitarse a acceder a propiedades simples y métodos de consulta del objeto que no tengan parámetros.

Las características principales de las vistas son (7):

- Administra la visualización y la presentación de la información.
- Observa al modelo para actualizar los cambios.
- Al definirse en el modelo una interfaz clara y estable, es fácil implementar múltiples vistas para un mismo modelo.
- Muy dependiente del modelo.

2.2.4 EL CONTROLADOR

El controlador organiza la interacción entre las vistas y el modelo. Recibe las peticiones del usuario, interactúa con el modelo realizando consultas y modificaciones a este, decide que vista se muestra en respuesta y le proporciona los datos requeridos para su renderizado, o delega la respuesta a otra acción de otro controlador. También puede enviar comandos a su vista asociada si solicita un cambio en la forma en la que se presenta el modelo, por lo tanto, requiere de dicho modelo la información que debe presentarse como salida.

Las principales características del controlador son (7):

- Responsable de definir el comportamiento de la aplicación.
- Recibe los eventos del usuario y decide qué es lo que se debe hacer, mapeándolos en comandos hacia el Modelo.
- Altamente dependiente de los dispositivos y mecanismos de interacción del usuario.

2.2.5 ¿CÓMO FUNCIONA EL PATRÓN MVC?

Aunque se pueden encontrar diferentes implementaciones de MVC, el flujo de control que se sigue generalmente es el siguiente: (5)

- El usuario interactúa con la interfaz de usuario de alguna forma (por ejemplo, el usuario pulsa un botón, enlace, etc.)
- El controlador recibe (por parte de los objetos de la interfaz-vista) la notificación de la acción solicitada por el usuario. El controlador gestiona el evento que llega, frecuentemente a través de un gestor de eventos (handler) o callback.
- El controlador accede al modelo, actualizándolo, posiblemente modificándolo de forma adecuada a la acción solicitada por el usuario. Los controladores complejos están a menudo estructurados usando un patrón de comando que encapsula las acciones y simplifica su extensión.
- El controlador delega a los objetos de la vista la tarea de desplegar la interfaz de usuario. La vista obtiene sus datos del modelo para generar la interfaz apropiada para el usuario donde se reflejan los cambios en el modelo (por ejemplo, produce un listado del contenido del carro de la compra). El modelo

no debe tener conocimiento directo sobre la vista. Sin embargo, se podría utilizar el patrón Observador para proveer cierta dirección entre el modelo y la vista, permitiendo al modelo notificar a los interesados de cualquier cambio. Un objeto vista puede registrarse con el modelo y esperar a los cambios, pero aun así el modelo en sí mismo sigue sin saber nada de la vista. Este uso del patrón Observador no es posible en las aplicaciones Web puesto que las clases de la vista están desconectadas del modelo y del controlador. En general el controlador no pasa objetos de dominio (el modelo) a la vista aunque puede dar la orden a la vista para que se actualice.

- La interfaz de usuario espera nuevas interacciones del usuario, comenzando el ciclo nuevamente.

A continuación se presenta la ilustración que muestra la manera cómo funciona el patrón MVC:



FIGURA 2II-2 FUNCIONAMIENTO DEL PATRÓN DE DISEÑO MVC

2.2.6 VENTAJAS Y DESVENTAJAS DEL PATRÓN MVC

Las ventajas y desventajas del patrón MVC son muchas (8) (9) pero las más importantes son:

VENTAJAS

- **Claridad en el diseño:** facilita en entendimientos del modelo, facilita la implementación y mantenimiento de la aplicación.
- **Modularidad:** Esta modularidad en el diseño permite a los componentes de la aplicaciones sean flexibles, permitiendo agregar o quitar estos sin afectar a gran medida la aplicación. El desarrollo de los distintos componentes se puede realizar en paralelo, una vez que la interfaz entre los componentes este claramente definida.
- **Vistas Múltiples:** permite mostrar el estado en que se encuentra el modelo de diferentes maneras como por ejemplo los juegos. Permitiendo tener una vista para mostrar el estado del modelo y otra vista que recoge los datos.
- **Facilidad de crecimientos:** esto se da cuando existe versionamientos de la aplicación en donde cada componente se va desarrollando, así mismo se puede presentar cuando se necesitan dos tipos de usuarios para la aplicación: usuario o administrador, en donde existe una variación para cada uno, ya que utilizan el mismo modelo, sin embargo cambia la vista y el controlador.
- **Distribuible:** se puede distribuir cualquier aplicación MVC, en aplicaciones cliente y servidor. En donde físicamente la parte de la vista se ubica en un cliente, mientras que la parte del controlador y el modelo se encuentra ubicada en otra locación, accediendo este por medio de una red.

DESVENTAJAS

- El tiempo de desarrollo de una aplicación que implementa el patrón de diseño MVC es mayor, al menos en la primera etapa, que el tiempo de desarrollo de una aplicación que no lo implementa, ya que MVC requiere que el programador implemente una mayor cantidad de clases que en un entorno de desarrollo común no son necesarias. Sin embargo, esta desventaja es muy relativa ya que posteriormente, en la etapa de mantenimiento de la aplicación, una aplicación MVC es muchísimo más fácil de realizar mantenimiento, extensible y modificable que una aplicación que no lo implementa.
- MVC requiere la existencia de una arquitectura inicial sobre la que se deben construir clases e interfaces para modificar y comunicar los módulos de una aplicación. Esta arquitectura inicial debe incluir, por lo menos: un mecanismo de eventos para poder proporcionar las notificaciones que genera el modelo de aplicación; una clase Modelo, otra clase Vista y una clase Controlador genéricas que realicen todas las tareas de comunicación, notificación y actualización que serán luego transparentes para el desarrollo de la aplicación.

2.3 MODELO VISTA MODELO DE VISTA

La creación de interfaces de usuario para una aplicación de software profesional no es fácil, ya que puede ser una mezcla de objetos visuales, conectividad multi-hilo, la seguridad, validación y pruebas unitarias. Hay patrones de diseño que resultan útiles para resolver dicha dificultad; pero separar adecuadamente y hacer frente a la multitud de problemas puede ser aún más difícil. Lo más complicado de los patrones de diseño

son los atajos que se utilizarán puede socavar todos los esfuerzos por hacer las cosas correctas.

Modelo Vista Modelo de Vista es la traducción más fiable de Model View ViewModel, este patrón es una variación del Modelo Vista Controlador que está diseñado para plataformas de desarrollo de interfaz de usuario moderna donde la vista es la responsabilidad de un diseñador en lugar de un desarrollador clásico, además MVVM también se basa en un mecanismo general de enlace de datos que facilita el desarrollo de la capa de separación de vista desde el resto del patrón mediante la eliminación del todo el código subyacente de la capa de la vista.

Los desarrolladores no requieren escribir código de interfaz de usuario puede utilizar el lenguaje de marcado y crear enlaces con el modelo de vista (ViewModel), lo que está escrito y mantenido por los desarrolladores de aplicaciones. Esta separación de funciones permite a los diseñadores interactivos centrarse en las interfaces de usuario sin necesidad de conocer la lógica de presentación.

El concepto fundamental de MVVM es separar el Modelo (Model) y la Vista (View) introduciendo una capa abstracta entre ellos, un “Modelo de la Vista” o “ViewModel”. La vista y el modelo de la Vista son instanciados normalmente por la aplicación contenedora. La vista guarda una referencia al ViewModel. El ViewModel expone comandos y entidades observables o enlazables a los que la Vista puede enlazarse. Las interacciones del usuario con la Vista dispararán comandos contra el ViewModel y de forma análoga, las actualizaciones en el ViewModel se propagarán a la Vista de forma automática mediante enlace de datos (10).

MVVM tiene como objetivo que los datos trasladados a la vista se puedan presentar y gestionar de la manera más sencilla. Es el ViewModel quien expone los datos desde el

modelo y, en este sentido, el ViewModel es más un modelo que una vista. Pero además gestiona la lógica de visualización de la vista por lo que esta, desde este punto de vista, es más una vista que un modelo. A día de hoy, la mezcla de responsabilidades sigue siendo un tema de discusión y exploración continua en el sector. (11)

El siguiente diagrama lo muestra a un alto nivel, sin entrar en detalles de implementación tecnológica:

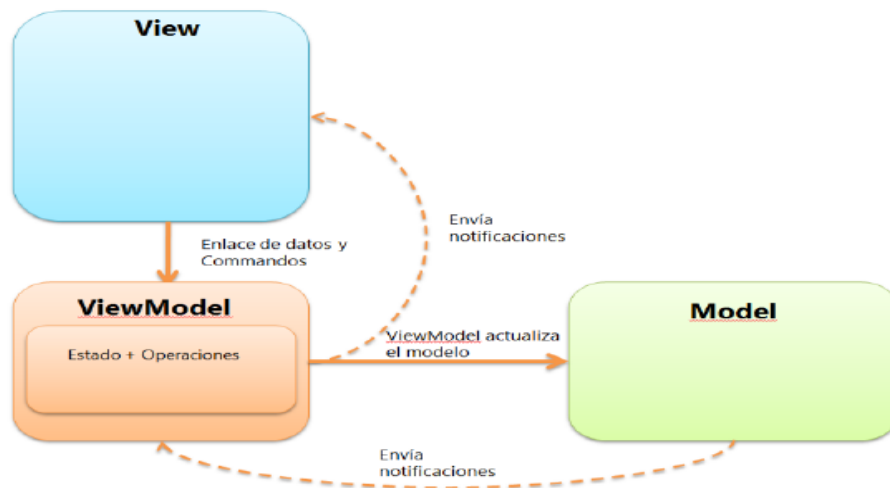


FIGURA II-3 Patrón MVVM (MODELO VIEW VIEWMODEL)

Fuente: Guía arquitectura N capas orientada al Dominio Figura 84

2.3.1 HISTORIA

Microsoft MVP Josh Smith informó que (10):

"En 2005, John Gossman, actualmente uno de los creadores WPF y Arquitectos de Silverlight de Microsoft, dio a conocer el patrón Modelo-Vista-ViewModel (MVVM) en su blog. MVVM es idéntica al modelo de presentación de Fowler, en la que ambos modelos cuentan con una abstracción de un View, que contiene el estado de una vista y la conducta. Fowler presentó el modelo de presentación como un medio para crear una

abstracción independiente de la plataforma de interfaz de usuario, mientras que Gossman introdujo MVVM como una forma estandarizada de las características principales de apalancamiento de WPF para simplificar la creación de interfaces de usuario. En ese sentido, considero MVVM como una especialización del patrón más general de PM, hecho a medida para el WPF y Silverlight”.

El patrón MVVM fue concebido para apoyar WPF y Silverlight, las dos piezas que debutó con el .NET Framework 3.0, que fue lanzado el 21 de noviembre de 2006. Este modelo se está aplicando de manera más amplia en otros campos tecnológicos, como ocurrió con los anteriores patrones Modelo Vista Controlador (MVC) o Modelo Vista Presentador (MVP).

Varios arquitectos que trabajan en Microsoft WPF ha escrito extensamente acerca de MVVM en los medios de comunicación en línea, incluyendo el creador John Gossman, Microsoft MVP Josh Smith y Microsoft Gerente del Programa Karl Shifflett.

Más recientemente, el patrón ha sido descrito también como modelo-vista-aglutinante (MVB). Esta es la primera aplicación informado del patrón fuera de la pila de Microsoft, a es ZK / Java. Una aplicación notable JavaScript de este patrón es Knockout.js.

2.3.2 EI MODELO

Encapsula la lógica de negocio y datos. La lógica empresarial se define como cualquier lógica de aplicación que se ocupa de la recuperación y gestión de datos de la aplicación y de asegurar que las reglas de negocio que garanticen la coherencia de los datos y la validez se imponen. Para maximizar las oportunidades de reutilización, los modelos no deben contener ningún caso de uso o conducta específica del usuario o lógica de la aplicación.

2.3.3 LA VISTA

Encapsula la interfaz de usuario y la lógica de la interfaz de usuario. Las vistas definen la interfaz de usuario específica para una parte de la aplicación; las vistas son normalmente sólo los controles que se han diseñado para funcionar bien cuando se une a un ViewModel. Su función específica es presentar al usuario una interfaz que representa los datos subyacentes de la aplicación y la lógica de la forma más adecuada, y para interpretar las acciones del usuario y las tramitará al ViewModel que define la forma en que afectan a los datos subyacentes y el estado de aplicación en el modelo. En algunos casos, la vista puede contener la lógica de la interfaz de usuario en la vista, el código subyacente. Sin embargo, tales códigos específicos de interfaz de usuario es difícil de probar y se debe evitar tanto como sea posible. Por supuesto, habrá momentos en que la vista va a tener un comportamiento específico que no puede ser expresado mediante declaración en XAML, lo que puede haber algo de código en la vista de código subyacente, pero este código debe centrarse específicamente en la gestión de los controles de interfaz de usuario en la vista y no debe implementar la lógica de presentación.

2.3.4 EL VIEWMODEL

Encapsula la lógica de presentación y estado. La lógica de presentación se define aquí como la lógica de la aplicación que se ocupa de casos de uso de la aplicación (o casos de usuario, tareas de usuario, flujo de trabajo, etc.) y define el comportamiento lógico y la estructura de la aplicación. El modelo de vista no es el código de un control de atrás. La lógica de presentación no debe tener idea de lo que la vista será y debe ser independiente de cualquier implementación específica de la interfaz de usuario. Para maximizar las oportunidades de reutilización, el ViewModel no debe tener ninguna

referencia a las clases específicas de interfaz de usuario, elementos, controles o comportamiento (como animaciones). No se debe derivar de cualquier clase base de interfaz de usuario (lo ideal es que sea un POCO). Se debe probar con pruebas unitarias y ser capaz de funcionar sin ningún tipo de interfaz de usuario.

2.3.5 ¿CÓMO FUNCIONA EL PATRÓN MVVM?

Un ViewModel nos sirve de la tercera capa intermedia que necesitamos, nos proporciona una abstracción que actúa como una meta-vista (un modelo de una vista), guardando estados y políticas que son compartidas por una o un conjunto de Vistas.

Al introducir el concepto de ViewModel, nuestro diseño quedaría como el siguiente:

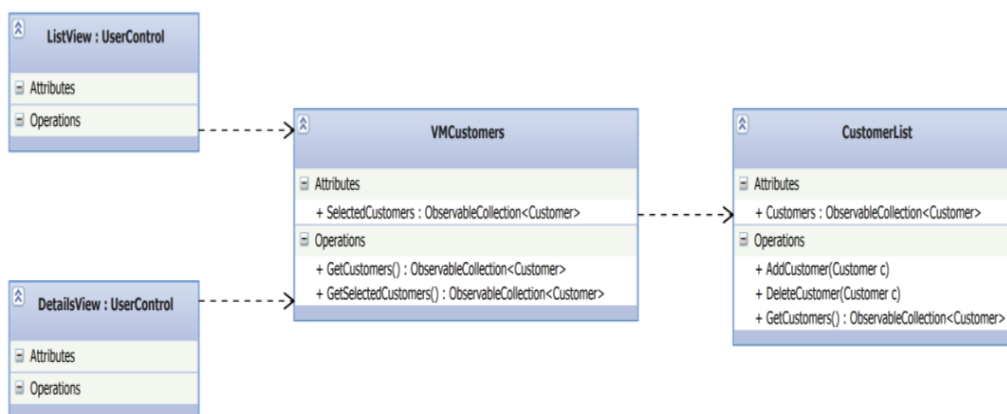


FIGURA II-4 DISEÑO DE UNA APLICACIÓN USANDO MVVM

Fuente: Guía arquitectura N capas orientada al Dominio Figura 96

En este diseño, las vistas conocen el ViewModel y se enlazan a sus datos, para poder reflejar cualquier cambio que tenga. El ViewModel no tiene ninguna referencia a las Vistas, solo tiene una referencia al Modelo, en nuestro caso CustomerList.

Para las Vistas, el ViewModel actúa como frente del modelo pero también como una forma de compartir estados entre vistas.

Adicionalmente, el ViewModel expone normalmente “Comandos” a los que las Vistas pueden enlazarse (11).

Esto pretende que cuando el usuario manipule las vistas, las acciones realizadas por el usuario modifiquen al ViewModel sin realizar ninguna acción al modelo.

2.3.6 VENTAJAS Y DESVENTAJAS DEL PATRÓN MVVM

El uso del patrón MVVM proporciona varias ventajas entre los más importantes tenemos: (11)

- Un ViewModel proporciona un único depósito de estados y políticas de presentación, lo que mejora la reutilización del Modelo (desacoplándolo de las Vistas) y facilita el reemplazamiento de las Vistas (al eliminar políticas específicas de presentación, de las Vistas)
- Un diseño MVVM mejora la facilidad de realizar Testing (Pruebas Unitarias especialmente) de la aplicación. Al separar la lógica de las vistas y controles visuales, podemos crear fácilmente pruebas unitarias que se encarguen exclusivamente del Modelo y del ViewModel (Puesto que las Vistas serán normalmente solo XAML, sin “code-behind”). Adicionalmente, MVVM también facilita la implementación de MOCKs en la Capa de Presentación, porque habiendo desacoplado las Vistas del Modelo y situando la lógica cliente en los ViewModels, esa lógica es fácilmente sustituible por MOCKs (simulación de ejecución), que es fundamental para las pruebas de aplicaciones complejas.
- El patrón MVVM ofrece un diseño desacoplado. Las Vistas solo tienen referencia al ViewModel y el ViewModel referencia solo al Modelo. El resto lo realiza el databinding y Commands de la infraestructura.

Las desventajas provocadas por el uso del patrón MVVM son:

- La relación típica entre un ViewModel y sus correspondientes Vistas son normalmente “una a muchas”, pero hay situaciones en las que eso no es cierto. En general, cualquier lógica de negocio cliente y lógica de negocio de validación de entrada de datos (seguimiento de selección de elementos, etc.) debe implementarse en el ViewModel.
- Hay situaciones donde un ViewModel es “consciente” de otro ViewModel dentro de la misma aplicación. Esas situaciones aparecen cuando hay una relación maestro-detalle entre dos ViewModels o cuando un ViewModel representa a un elemento suelto (por ejemplo, la representación visual de un único Cliente). Cuando esto sucede, un ViewModel puede representar a una colección de ViewModels.

2.4 DIFERENCIA ENTRE EL PATRÓN MVC Y MVVM

En la Figura II-5 se observa la manera que interactúan los elementos de cada patrón mientras que en la Tabla II-I. Se muestra algunas diferencias entre los patrones en estudio. El detalle a continuación:

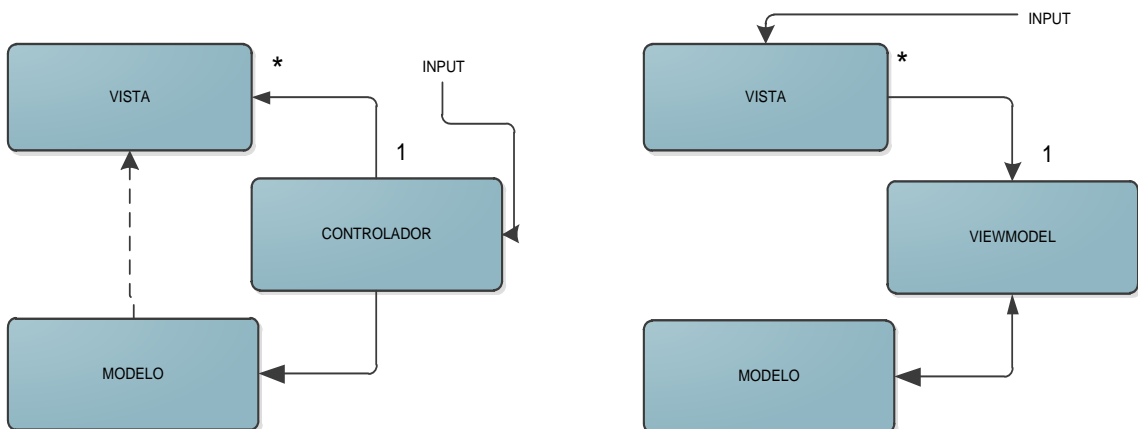


FIGURA II-5 PATRONES MVC Y MVVM

En la Tabla II-I se muestra las diferencias entre Model View ViewModel y Model View Controller.

TABLA II-I DIFERENCIAS ENTRE LOS PATRONES MVC Y MVVM

MVC	MVVM
La acción realizada por el usuario afecta directamente al controlador	La acción realizada por el usuario afecta directamente a la vista
La relación entre el controlador y la vista es de uno a muchos.	La relación entre la Vista y el ViewModel es de muchos a uno.
La vista no tiene conocimiento del controlador	La vista sostiene relación con el ViewModel mientras que el ViewModel no posee información de la vista. Diferentes tecnologías pueden compartir el mismo ViewModel
El Controlador el paso de acciones del modelo a la vista, la vista también posee conocimiento del modelo	La vista no es consciente del Modelo

CAPÍTULO III

3 ESTUDIO COMPARATIVO ENTRE LOS PATRONES DE DISEÑO MVC Y MVVM

Para el diseño de interfaces de usuario se debe tener en cuenta el contenido, la tecnología, los componentes visuales y los económicos. El contenido busca informar, la tecnología se orienta a la funcionalidad del sitio, los componentes visuales proporcionan la apariencia y presentan la información y los económicos se enfocan hacia los bajos costos.

En la actualidad existen varios patrones para el diseño de interfaces de usuario, cada uno con sus facilidades y dificultades. Para lograr identificar el patrón que nos presente mejores ventajas en el desarrollo de aplicaciones web profesionales se deberá realizar un estudio comparativo entre los patrones existentes, en el presente caso serán los patrones Modelo Vista Controlador (MVC) y Model Vista Modelo de Vista (MVVM).

El estudio comparativo de los patrones MVC y MVVM se realizará en dos frentes: rendimiento y productividad.

Normalmente los desarrolladores se hacen cargo de proyectos cuando ya tiene problemas y se debe tener en cuenta que lo que funciona bien con los desarrolladores no funciona bien con los usuarios.

Uno de los puntos más importantes en el desarrollo web es el rendimiento de la página web puede que se haya invertido mucho tiempo y dedicación al diseño de nuestro sitio web y, sin embargo, no estemos obteniendo los resultados que esperábamos. Una página web veloz y optimizada mejora considerablemente la experiencia web del usuario.

La productividad otra características importante para poder seleccionar el mejor patrón para el desarrollo de interfaces de usuario. Se debe tener en cuenta el servidor de desarrollo y el equipo de trabajo.

3.1 PROTOTIPOS

Con el propósito de determinar que patrón de interfaces de usuario es la más adecuada con respecto a productividad y rendimiento, se busca comparar dos tipos de aplicaciones web, las que usarán el patrón Model View Controller (MVC) y Model View ViewModel (MVVM).

- Prototipo 1: fue implementado con el patrón MVC con la tecnología ASP.net MVC 4.
- Prototipo 2: fue implementado con el patrón MVVM con la tecnología Light Switch y Silverlight

Cada prototipo constará de los siguientes módulos:

- Módulo de autenticación.
- Módulo de gestión de empleados.
- Módulo de gestión de clientes.
- Módulo de gestión de proveedores.

- Módulo de gestión de productos.
- Módulo de gestión de compras.
- Módulo de gestión de ventas.

3.2 DETERMINACIÓN DE LOS PARÁMETROS DE COMPARACIÓN

La productividad determina como los patrones de diseño pueden mejorar la comunicación desarrollador-diseñador y así mejorar el sitio web de manera fácil, simple y rápida.

Además los factores que intervienen para determinar el grado de rendimiento de los patrones de diseño estudiados como son la velocidad de carga, envío de peticiones y las transferencias de recursos del sitio web.

3.2.1 VARIABLE 1: RENDIMIENTO

Según Wikipedia el concepto de rendimiento es: “En la ingeniería del software, las pruebas de rendimiento son los ensayos que se realizan, desde una perspectiva, para determinar lo rápido que realiza una tarea la aplicación web en condiciones particulares de trabajo. Las pruebas de rendimiento son un subconjunto de la ingeniería de pruebas, una práctica informática que se esfuerza por mejorar el rendimiento, englobándose en el diseño y la arquitectura de un sistema, antes incluso del esfuerzo inicial de la codificación” (12)

Para realizar correctamente las pruebas de rendimiento se ha propuesto los siguientes parámetros de rendimiento: Infraestructura de Hardware, pruebas de carga transaccional alta y pruebas de estrés. Estos parámetros están especificados en la Tabla III-I.

TABLA III-I PARÁMETROS DE RENDIMIENTO

PARÁMETRO	DESCRIPCIÓN
1.1.Indicadores claves	Contadores que describen los aspectos básicos de la realización de la prueba.
1.2.Tiempo de Respuesta de la Pagina	El tiempo de medio de la respuesta de las páginas web que se accede durante las pruebas de carga.
1.3.Controladores y Agentes	Información sobre los equipos en los que se ejecutan las pruebas de carga. Esto incluye datos sobre el uso de la memoria, el procesador, el disco físico, los procesos.

Los parámetros antes mencionados poseen indicadores que servirán de referencia cuando se realice las pruebas de rendimiento. La Tabla III-II detalla los indicadores de cada parámetro de rendimiento que se utilizará.

TABLA III-II INDICADORES DE LOS PARÁMETROS DE RENDIMIENTO

PARÁMETRO	INDICADOR	DESCRIPCIÓN
1.1. Indicadores claves	1.1.1. Promedio de tiempo de descarga de la página	Tiempo medio para descargar las páginas y todas sus solicitudes.
	1.1.2. Páginas por segundo	Número de veces que se está accediendo a la memoria virtual.
1.2. Tiempo de Respuesta de la pagina	1.2.1. Solicitudes/segundo	El número de solicitudes ejecutado por segundo. Esto representa el rendimiento actual de la aplicación.
	1.2.2. Promedio tiempo de respuesta	Muestran una relación entre el tiempo total transcurrido del intervalo de muestra con el número de procesos u operaciones realizadas durante ese tiempo.
1.3. Controladores y Agentes	1.3.1. Porcentaje del tiempo del procesador	Porcentaje del CPU utilizado.
	1.3.2. Mbyte disponibles	Memoria disponible para nuevos procesos

3.2.2 VARIABLE 2: PRODUCTIVIDAD

El probar un sitio web durante la etapa de diseño y desarrollo permite obtener vistas previas del trabajo y por ende tomar decisiones en base a una referencia completa de los datos de la página.

Para realizar una correcta medición de la productividad de las aplicaciones web se ha propuesto los siguientes parámetros de productividad: generación de código, modularidad, presentación que se detallan a continuación en la Tabla III-III

TABLA III-III PARÁMETROS DE PRODUCTIVIDAD

PARÁMETRO	DESCRIPCIÓN
2.1.Productividad	Mediante este parámetro se demuestra el nivel de sencillez y aprendizaje que posee los patrones MVC y MVVM a la hora de instruirse e implementarlos en el desarrollo de aplicaciones web.

De los parámetros establecidos para la comparación se ha definido indicadores que se los utilizará como referencia en la medición de la productividad de las aplicaciones web.

Los indicadores se los detalla en la Tabla III-IV.

TABLA III-IV ÍNDICES DE LOS PARÁMETROS DE PRODUCTIVIDAD

PARÁMETRO	INDICADORES	DESCRIPCIÓN
2.1.Productividad	2.1.1. Líneas de Código	Índice que permite medir la cantidad de líneas de código para la realización de una consulta en específico.
	2.1.2. Complejidad	Índice que permite medir la facilidad o complejidad de entendimiento del código.
	2.1.3. Disponibilidad de información	La información disponible de cada una de las herramientas es importante para aquellas personas que necesitan realizar diseños profesionales o tener conocimiento más avanzado.

3.3 HERRAMIENTAS PARA PRUEBAS DE RENDIMIENTO Y PRODUCTIVIDAD

Para las pruebas de rendimiento se utilizan herramientas que puedan monitorear, que midan qué partes del software contribuyen más al deficiente rendimiento o para establecer niveles del mismo que mantenga un tiempo de respuesta aceptable.

Para alcanzar un nivel de rendimiento aceptable para un nuevo sistema, los esfuerzos de estas pruebas deben comenzar en el inicio del proyecto de desarrollo y se continúen durante su construcción.

Las herramientas que se utilizará para las medir los indicadores de las pruebas de rendimiento se detallan en la Tabla III-V.

TABLA III-V HERRAMIENTAS PARA REALIZAR LAS PRUEBAS DE RENDIMIENTO

HERRAMIENTA	DESCRIPCIÓN	INDICADORES A MEDIR
Herramientas de Análisis de Código de Visual Studio 2012	“Las métricas de código son un conjunto de medidas de software que proporcionan a los programadores una mejor visión del código que están desarrollando. Al aprovechar las métricas de código, los programadores pueden entender qué tipos y métodos se deben rehacer o probar más a fondo. Los equipos de desarrollo pueden identificar los riesgos potenciales, entender el estado actual de un proyecto y seguir el progreso durante el desarrollo del software”. (13)	<ul style="list-style-type: none">• Líneas de Código.• Complejidad.• Disponibilidad de información.
Herramientas de Prueba y Rendimiento Web de Visual Studio	Para determinar lo bien que responde su programa a diferentes niveles de uso, se realizan pruebas de carga. Una	<ul style="list-style-type: none">• Promedio de tiempo de descarga de la

2012	prueba de carga modela el uso previsto de un programa de software simulando varios usuarios que obtienen acceso al programa al mismo tiempo. Visual Studio Ultimate le permite utilizar un número ilimitado de usuarios virtuales en una ejecución de prueba de carga local y remota. Los usuarios virtuales que se utilizan en las pruebas de carga utilizan arquitectura de multiprocesador (14).	página. <ul style="list-style-type: none">• Páginas por segundo.• Solicitudes/segundo.• Promedio tiempo de respuesta.• Porcentaje del tiempo del procesador.• Mbyte disponibles
------	---	---

3.4 PRUEBA DE HIPÓTESIS

Según la definición: La prueba de hipótesis es un procedimiento basado en la evidencia muestral y la teoría de probabilidad que se emplea para determinar si la hipótesis es una afirmación razonable (15).

Para realizar la prueba de Hipótesis siguiendo los pasos a continuación descritos (15):

- 1. Redacción de la hipótesis:** Para la redacción se tomara en cuenta la hipótesis alternativa y nula y las describirá para cada variable en estudio.
- 2. Determinar el valor alfa:** Para el valor al alfa tomamos el 5% ya que dicho valor es valor por defecto para las pruebas de hipótesis el valor de alfa será de 0.05 este será el valor de significancia o grado de error.
- 3. Elección de la prueba estadística:** El tema de tesis en exposición muestra que es un estudio transversal de dos grupos (MVC y MVVM) y la variable que se mide es numérica determina que la prueba para determinar las diferencias significativas de las hipótesis alterna y nula es la de U de Mann-Whitney prueba no paramétrica para muestras independientes.

4. Lectura de P-Valor: Para calcular el P-valor se usará el software estadístico SPSS 17 en español, dicho programa permitirá calcular la estadística descriptiva de las mediciones así como el P-valor

5. Decisión estadística: Consiste en el rechazo o aceptación de la hipótesis nula

El criterio para aceptar o rechazar la hipótesis nula es:

- **P-valor > α :** se acepta la hipótesis nula.
- **P-valor $\leq \alpha$:** se rechaza la hipótesis nula.

Los indicadores en estudio se los calificará de acuerdo al siguiente análisis detallado en la Tabla III-VI en la cual propone el diseño del experimento.

TABLA III-VI DISEÑO DEL EXPERIMENTO

Indicador	Decisión Optima	Peso Ponderado Máximo
Promedio de tiempo de descarga de la página	A menor valor mayor rendimiento	15
Páginas por segundo	A menor valor mayor rendimiento	10
Solicitudes/segundo	A menor valor mayor rendimiento	10
Promedio tiempo de respuesta	A menor valor mayor rendimiento	15
Porcentaje del tiempo del procesador	A menor valor mayor rendimiento	10
Mbyte disponibles	A mayor valor mayor rendimiento	10
Líneas de Código	A menor valor mayor productividad	10
Complejidad	A menor valor mayor productividad	15
Disponibilidad de información	A mayor valor mayor productividad	5
Total		100

3.5 ANÁLISIS DE LOS RESULTADOS

Para realizar el análisis del rendimiento y de la productividad de los patrones MVC y MVVM se lo realiza por cada indicador de rendimiento y productividad.

3.5.1 METODOLOGÍA PARA LAS PRUEBAS DE RENDIMIENTO

Según Microsoft en su guía “Performance Testing Guidance for Web Applications” la metodología para las pruebas de rendimiento consisten en las siguientes actividades:

(16)

- 1. Identificar el entorno de las pruebas:** Identificar el entorno físico de pruebas y el entorno de producción, así como las herramientas y recursos de que dispone el equipo de prueba. El entorno físico incluye hardware, software y configuraciones de red. Tener un profundo conocimiento de todo el entorno de prueba desde el principio permite diseños más eficientes de pruebas y la planificación y ayuda a identificar problemas en las pruebas en fases tempranas del proyecto. En algunas situaciones, este proceso debe ser revisado periódicamente durante todo el ciclo de vida del proyecto.
- 2. Identificar los criterios de aceptación del rendimiento:** Determinar el tiempo de respuesta, el rendimiento, la utilización de los recursos y los objetivos y limitaciones. En general, el tiempo de respuesta concierne al usuario, el rendimiento al negocio, y la utilización de los recursos al sistema. Además, identificar criterios de éxito del proyecto que no hayan sido recogidos por los objetivos y limitaciones.
- 3. Planificar y diseñar las pruebas:** Identificar los principales escenarios, determinar la variabilidad de los usuarios y la forma de simular esa variabilidad, definir los datos de las pruebas, y establecer las métricas a recoger. Consolidar

esta información en uno o más modelos de uso del sistema a implantar, ejecutarlo y analizarlo.

4. **Configurar el entorno de prueba:** Preparar el entorno de prueba, herramientas y recursos necesarios para ejecutar cada una de las estrategias, así como las características y componentes disponibles para la prueba. Asegurarse de que el entorno de prueba se ha preparado para la monitorización de los recursos según sea necesario.
5. **Aplicar el diseño de la prueba:** Desarrollar las pruebas de rendimiento de acuerdo con el diseño del plan.
6. **Ejecutar la prueba:** Ejecutar y monitorizar las pruebas. Validar las pruebas, los datos de las pruebas, y recoger los resultados. Ejecutar pruebas válidas para analizar, mientras se monitoriza la prueba y su entorno.
7. **Analizar los resultados, realizar un informe y repetirlo:** Consolidar y compartir los resultados de la prueba. Analizar los datos, tanto individualmente, como con un equipo multidisciplinario. Volver a priorizar el resto de las pruebas y volver a ejecutarlas de ser necesario. Cuando todas las métricas estén dentro de los límites aceptados, ninguno de los umbrales establecidos han sido rebasados, y toda la información deseada se ha reunido, las pruebas han acabado para el escenario definido por la configuración.

3.5.2 ANÁLISIS DEL PARÁMETRO 1: RENDIMIENTO

Para realizar el análisis del rendimiento se empleará la metodología propuesta por Microsoft y que se detalló en el apartado anterior.

Identificar el entorno de las pruebas:

Para realizar las pruebas de rendimiento se utilizará el siguiente hardware y software:

Hardware:

- Sistema operativo: Windows 7 Ultimate de 64 bits.
- Procesador: Intel® Core™ i5-2430 CPU@ 2.40 GHz.
- Memoria: 8 Gigabytes
- Maquina: Sony VAIO VPCEG25FL.

Software:

- Herramientas de medición de rendimiento Visual Studio 2012 Ultimate.

Configurar el entorno de prueba

Para desarrollar las pruebas de rendimiento se necesita un entorno de pruebas que permita obtener las mediciones de los indicadores anotados en el apartado anterior.






Para el entorno de pruebas se seleccionó el módulo de Gestión de Compras. Se tomó la decisión de optar por estos módulos para la medición del rendimiento debido a que son los módulos más utilizados por los usuarios, es lo más obvio ya que el prototipo se lo desarrollado para la gestión de compra de productos.

Se implementará el proyecto de Test denominado “Rendimiento” en donde se escogerá “Web Performance and Load Test Project” en donde se registrará las acciones que se deseen realizar en los prototipos. Además se creará una prueba de carga en donde se configurará los pasos necesarios para la ejecución de la prueba.

Identificar los criterios de aceptación del rendimiento

Los criterios que se utilizarán son los expuestos en la Tabla III-II Indicadores de los parámetros de rendimiento, con lo cual se obtendrá resultados cuantitativos y cualitativos que permitirán realizar una selección sustentada de uno de las dos arquitecturas analizadas, estos serán calificados o evaluados según el nivel de cumplimiento expuesto en la Tabla III-VII

TABLA 3III-VII NIVEL DE CUMPLIMIENTO DEL RENDIMIENTO

VALORACIÓN	CALIFICACIÓN	FORMA GRAFICA	VALOR	DESCRIPCIÓN
Excelente	>65% y =70%		5	Cumple los requerimientos en su totalidad
Muy Bueno	>60% y <=65%		4	Cumple los requerimientos en su mayor parte
Buena	>55% y <=60%		3	Cumple los requerimientos aceptablemente
Regular	>45% y <=55%		2	Cumple los requerimientos en mínima parte
Malo	>40% y <=45%		1	No cumple los requerimientos

Aplicar el diseño de la prueba y Ejecutar la Prueba

Se diseñará la prueba donde los prototipos deberán seguir los siguientes pasos:

- Gestión de Ventas:
 - a) Seleccionar Cliente.
 - b) Seleccionar empleado.
 - c) Seleccionar fecha.
 - d) Añadir número de factura.
 - e) Seleccionar el estado de la factura.
 - f) Seleccionar 5 productos.
 - g) Seleccionar los productos con las diferentes unidades de venta que existen como son: unidad, decena, ciento y millar.
 - h) Guardar la factura.

El diseño propuesto se aplicará en el proyecto de “Rendimiento” ejecutando el web test record que grabará las acciones que se realiza en el diseño de la prueba que se concibió anteriormente. Para realizar las mediciones de los parámetros de rendimiento se deberá

utilizar la prueba de carga previamente configurado. Ejecutamos la prueba y se analiza los datos que se obtienen de las pruebas.

Analizar los resultados

El número de usuarios concurrentes para la aplicación es calculado mediante la fórmula estadística que es aplicada cuando no se conoce la población universal.

$$n = \frac{Z_{\alpha}^2 * p * q}{d^2}$$

De donde:

n: Tamaño de la muestra.

Z_{α}^2 : El nivel de confianza escogido es del 95%. Este valor es 1,96 según la tabla de valores Z más usados.

P: Proporción esperada, para el caso de estudio se considera una proporción del 50% para reducir el factor de error (0.5).

q: Proporción **1-p**

d: Margen de error considerado para la prueba, considerado el 5%.

Al aplicar la formula anterior obtenemos una muestra de 384.16 que aproximamos al inmediato superior la cantidad de usuarios y obtenemos 385.

La calificación de cada parámetro se realiza mediante la suma de las calificaciones obtenidas en los análisis de cada indicador, haciendo uso de las siguientes formulas:

$$C_{MVVM} = \sum_{i=0}^n V_i \quad 1$$

$$C_{MVC} = \sum_{i=0}^n V_i \quad 2$$

$$C_{MAX} = \sum_{i=0}^n VM_i \quad 3$$

De donde:

C_{MVVM} : Calificación para el patrón Modelo Vista Modelo de Vista.

C_{MVC} : Calificación para el patrón Modelo Vista Controlador

C_{MAX} : Calificación máxima sobre el que se mide el parámetro.

n: número de indicadores del parámetro.

V_i : Valor de calificación de cada indicador.

VM_i : Valor máximo de cada calificador.

Luego de ejecutar las pruebas se obtiene los resultados individuales con los cuales se calcula la media, el promedio, los errores relativos, porcentuales, desviación estándar e intervalos de confianza, con el objetivo de establecer datos significativos y confiables que servirán a su vez en la evaluación de los indicadores.

Las pruebas ejecutadas para los patrones MVC y MVVM se las puede apreciar en la sección de anexos y los resultados de las mediciones para los patrones en estudio e indicadores seleccionados se presenta a continuación:

1.1. Indicadores claves

Para este parámetro se realizó las pruebas en los indicadores detallados a continuación, cabe mencionar que los resultados de las dichas pruebas se encuentran en la sección de Anexos I y Anexo III para los patrones MVC y MVVM respectivamente

1.1.1. Promedio de Tiempo de descarga de la Página:

Para el análisis estadístico del indicador Promedio de Tiempo de descarga de la Página tenemos que:

H₀: No Existe una diferencia significativa entre la media del promedio de tiempo de página para el patrón MVC con respecto al patrón MVVM.

H₁: Existe una diferencia significativa entre la media del promedio de tiempo de página para el patrón MVC con respecto al patrón MVVM.

En la Tabla III-VIII se muestra la estadística descriptiva obtenida del software SPSS.

Los datos obtenidos son:

TABLA III-VIII ESTADÍSTICA DESCRIPTIVA PROMEDIO DE TIEMPO DE DESCARGA DE LA PÁGINA

	Frecuencia	Promedio	Error Relativo	Sx
MVVM	241	0.07052574	0.020537639	0.318829891
MVC	241	1.433100848	0.46068736	7.151791065

En la Tabla III-IX se muestra los datos obtenidos de la prueba U de Mann-Whitney:

TABLA III-IX ESTADÍSTICOS DE CONTRASTE PROMEDIO DE TIEMPO DE DESCARGA DE LA PÁGINA

	AvgPageTime
U de Mann-Whitney	13157.500
W de Wilcoxon	42318.500
Z	-10.388
Sig. asintót. (bilateral)	2.803E-25

El P-Valor para la prueba U de Mann-Whitney como se observa en la Tabla III-IX es de 2.803E-25, por lo tanto se deduce que el valor de significancia asintótica es menor al

valor alfa ($2.803E-25 < 0,05$) por lo tanto se concluye que se rechaza la hipótesis nula y se acepta la hipótesis alterna es decir:

“Existe una diferencia significativa entre la media del promedio de tiempo de descarga de la página para el patrón MVC con respecto al patrón MVVM”.

Debido a que el indicador PROMEDIO DE TIEMPO DE DESCARGA DE LA PÁGINA es un valor en promediado medido en segundos la relación entre ambos patrones es inversamente proporcional, es decir a menos tiempo mayor es su rendimiento por tal motivo el patrón MVVM presenta mejores condiciones, en el indicador en estudio, que el patrón MVC.

Calificación

Para la calificación del indicador Promedio de Tiempo de la página se utilizará los valores establecidos en la Tabla III-VI el valor del peso ponderado máximo, por lo tanto en la Tabla III-X se explica la calificación para cada patrón para el indicador en estudio. Para esto se analiza los promedios observados en la Tabla III-VIII

TABLA 3III-X CALIFICACIÓN POR PATRÓN PARA EL PROMEDIO DE TIEMPO DE DESCARGA DE LA PAGINA

Patrón	Valor cualitativo
MVVM	15
MVC	0.74

Como el patrón MVVM presenta mejores condiciones en el indicador en estudio que MVC; MVVM se lo califica con el valor más alto que es 15 y a MVC se lo calificar con un valor inversamente proporcional al de MVVM.

1.1.2. Páginas por segundo:

Para el análisis estadístico para el indicador Páginas por segundo se tiene:

H₀: No Existe una diferencia significativa entre la media de páginas por segundo para el patrón MVC con respecto al patrón MVVM.

H₁: Existe una diferencia significativa entre la media de páginas por segundo para el patrón MVC con respecto al patrón MVVM

En la Tabla III-XI muestra la estadística descriptiva de los patrones para el indicador en estudio.

TABLA 3-XI ESTADÍSTICA DESCRIPTIVA PARA PÁGINAS POR SEGUNDO

	Frecuencia	Promedio	Error Relativo	Sx
MVVM	241	314.1613732	59.39171856	0,04606
MVC	241	394.6475189	58.84198053	0,06815

En la tabla III-XII se muestra los datos obtenidos de la prueba U de Mann-Whitney:

TABLA 3III-XII ESTADÍSTICOS DE CONTRASTE PÁGINAS POR SEGUNDO

	PagesSec
U de Mann-Whitney	15327.000
W de Wilcoxon	44488.000
Z	-8.969
Sig. asintót. (bilateral)	2.989E-19

El P-Valor para la prueba U de Mann-Whitney como se observa en la Tabla 3-12 es de 2.989E-19, por lo tanto se deduce que el valor de significancia asintótica es menor al valor alfa ($2.989E-19 < 0,05$) por lo tanto se concluye que se rechaza la hipótesis nula y se acepta la hipótesis alterna es decir:

“Existe una diferencia significativa entre la media de páginas por segundo para el patrón MVC con respecto al patrón MVVM”.

Debido a que el indicador PÁGINAS POR SEGUNDO es un valor en promediado medido en segundos la relación entre ambos patrones es inversamente proporcional, es decir a menos tiempo mayor es su rendimiento por tal motivo el patrón MVVM presenta mejores condiciones, en el indicador en estudio, que el patrón MVC.

Calificación

Para la calificación del indicador Promedio de Tiempo de la página se utilizará los valores establecidos en la Tabla III-VI el valor del peso ponderado máximo, por lo tanto en la Tabla III-XIII se explica la calificación para cada patrón para el indicador en estudio.

Para esto se analiza los promedios observados en la Tabla III-XI

TABLA III-XIII CALIFICACIÓN POR PATRÓN DEL INDICADOR PÁGINA POR SEGUNDO

Patrona	Valor Cuantitativo
MVVM	10
MVC	7.96

Como el patrón MVVM presenta mejores condiciones en el indicador en estudio que MVC; MVVM se lo califica con el valor más alto que es 10 y a MVC se lo calificar con un valor inversamente proporcional al de MVVM.

1.2. Tiempo de Respuesta de la página

Para este parámetro se realizó las pruebas en los indicadores detallados a continuación, cabe mencionar que los resultados de las dichas pruebas se encuentran en la sección de Anexos I y Anexo III para los patrones MVC y MVVM respectivamente.

1.2.1. Solicitudes/segundo:

Para el análisis estadístico para el indicador Solicitudes/segundo se tiene:

H₀: No Existe una diferencia significativa entre la solicitud/segundo para el patrón MVC con respecto al patrón MVVM.

H₁: Existe una diferencia significativa entre la solicitud/segundo para el patrón MVC con respecto al patrón MVVM.

En la tabla III-XIV muestra el resumen de las mediciones de los patrones para el indicador en estudio.

TABLA III-XIV RESUMEN RESULTADOS DE LA MEDICIÓN DEL INDICADOR SOLICITUD/SEGUNDO

	Frecuencia	Promedio	Error Relativo	Sx
MVVM	241	11.70788382	0.326928659	5.075297619
MVC	241	678.7236515	13.87582041	215.4106601

En la tabla III-XV se muestra los datos obtenidos de la prueba U de Mann-Whitney:

TABLA III-XV ESTADÍSTICOS DE CONTRASTE SOLICITUD/SEGUNDO

	RequestSec
U de Mann-Whitney	2322.500
W de Wilcoxon	31483.500
Z	-17.475
Sig. asintót. (bilateral)	2.212E-68

El P-Valor para la prueba U de Mann-Whitney como se observa en la Tabla 3-15 es de $2.212E-68$, por lo tanto se deduce que el valor de significancia asintótica es menor al valor alfa ($2.212E-68 < 0,05$) por lo tanto se concluye que se rechaza la hipótesis nula y se acepta la hipótesis alterna es decir:

“Existe una diferencia significativa entre la media de solicitud/segundo para el patrón MVC con respecto al patrón MVVM”.

Debido a que el indicador SOLICITUD/SEGUNDO es un valor promediado medido en segundos la relación entre ambos patrones es inversamente proporcional, es decir a menos tiempo mayor es su rendimiento por tal motivo el patrón MVVM presenta mejores condiciones, en el indicador en estudio, que el patrón MVC.

Calificación

Para la calificación del indicador Promedio de Tiempo de la página se utilizará los valores establecidos en la Tabla III-VI el valor del peso ponderado máximo, por lo tanto en la Tabla III-XVI se explica la calificación para cada patrón para el indicador en estudio.

Para esto se analiza los promedios observados en la Tabla III-XIV

TABLA III-XVI CALIFICACIÓN POR PATRÓN DEL INDICADOR SOLICITUD/SEGUNDO

Patrón	Valor Cuantitativo
MVVM	10
MVC	0.17

Como el patrón MVVM presenta mejores condiciones en el indicador en estudio que MVC; MVVM se lo califica con el valor más alto que es 10 y a MVC se lo calificar con un valor inversamente proporcional al de MVVM.

1.2.2. Promedio tiempo de respuesta:

Para el análisis estadístico para el indicador Promedio tiempo de respuesta se tiene:

H₀: No Existe una diferencia significativa entre el promedio de tiempo de respuesta para el patrón MVC con respecto al patrón MVVM.

H₁: Existe una diferencia significativa entre promedio de tiempo de respuesta para el patrón MVC con respecto al patrón MVVM.

En la Tabla III-XVII muestra el resumen de las mediciones de los patrones para el indicador en estudio.

TABLA III-XVII ESTADÍSTICA DESCRIPTIVA DEL INDICADOR PROMEDIO TIEMPO DE RESPUESTA

	Frecuencia	Promedio	Error Relativo	Sx
MVVM	241	0.070525753	0.020537639	0.31882989
MVC	241	0.289595568	0.114490168	1.777365376

En la Tabla III-XVIII se muestra los datos obtenidos de la prueba U de Mann-Whitney:

TABLA 3III-XVIII ESTADÍSTICOS DE CONTRASTE PROMEDIO TIEMPO DE RESPUESTA

	RequestSec
U de Mann-Whitney	8931.000
W de Wilcoxon	38092.000
Z	-13.152
Sig. asintót. (bilateral)	1.647E-39

El P-Valor para la prueba U de Mann-Whitney como se observa en la Tabla III-XVIII es de 1.647E-39, por lo tanto se deduce que el valor de significancia asintótica es menor al valor alfa ($1.647E-39 < 0,05$) por lo tanto se concluye que se rechaza la hipótesis nula y se acepta la hipótesis alterna es decir:

“Existe una diferencia significativa entre la media del promedio de tiempo de respuesta para el patrón MVC con respecto al patrón MVVM”.

Debido a que el indicador PROMEDIO TIEMPO DE RESPUESTA es un valor promediado medido en segundos la relación entre ambos patrones en inversamente proporcional, es decir a menos tiempo mayor es su rendimiento por tal motivo el patrón MVVM presenta mejores condiciones, en el indicador en estudio, que el patrón MVC.

Calificación

Para la calificación del indicador Promedio de Tiempo de la página se utilizará los valores establecidos en la Tabla III-VI el valor del peso ponderado máximo, por lo tanto en la Tabla III-XIX se explica la calificación para cada patrón para el indicador en estudio.

Para esto se analiza los promedios observados en la Tabla III-XVII

TABLA III-XIX CALIFICACIÓN POR PATRÓN DEL INDICADOR PROMEDIO TIEMPO PÁGINA

Patrón	Valor Cuantitativo
MVVM	15
MVC	3.65

Como el patrón MVVM presenta mejores condiciones en el indicador en estudio que MVC; MVVM se lo califica con el valor más alto que es 15 y a MVC se lo calificar con un valor inversamente proporcional al de MVVM.

1.3 Controladores y Agentes

Para este parámetro se realizó las pruebas en los indicadores detallados a continuación, cabe mencionar que los resultados de las dichas pruebas se encuentran en la sección de Anexos I y Anexo III para los patrones MVC y MVVM respectivamente.

1.3.1. Porcentaje del tiempo del procesador:

Para el análisis estadístico para el indicador Porcentaje del tiempo del procesador se tiene:

H₀: No Existe una diferencia significativa entre el porcentaje del tiempo del procesador para el patrón MVC con respecto al patrón MVVM.

H₁: Existe una diferencia significativa entre porcentaje del tiempo del procesador para el patrón MVC con respecto al patrón MVVM.

En la Tabla III-XX muestra el resumen de las mediciones de los patrones para el indicador en estudio.

**TABLA 3III-XX ESTADÍSTICA DESCRIPTIVA DEL INDICADOR
PORCENTAJE DEL TIEMPO DEL PROCESADOR**

	Frecuencia	Promedio	Error Relativo	Sx
MVVM	241	33.70982817	0.488277434	7.564361484
MVC	241	62.1948585	0.657010877	10.17836873

En la Tabla III-XXI se muestra los datos obtenidos de la prueba U de Mann-Whitney:

TABLA III-XXI ESTADÍSTICOS DE CONTRASTE PORCENTAJE DEL TIEMPO DEL PROCESADOR

	ProcesorTime
U de Mann-Whitney	1494.000
W de Wilcoxon	30414.000
Z	-17.971
Sig. asintót. (bilateral)	3.305E-72

El P-Valor para la prueba U de Mann-Whitney como se observa en la Tabla III-XXI es de 3.305E-72, por lo tanto se deduce que el valor de significancia asintótica es menor al valor alfa ($3.305E-72 < 0,05$) por lo tanto se concluye que se rechaza la hipótesis nula y se acepta la hipótesis alterna es decir:

“Existe una diferencia significativa entre la media del porcentaje del tiempo del procesador para el patrón MVC con respecto al patrón MVVM”.

Debido a que el indicador PORCENTAJE DEL TIEMPO DEL PROCESADOR es un valor promediado medido en porcentaje la relación entre ambos patrones es inversamente proporcional, es decir a menos cantidad mayor es su rendimiento por tal motivo el patrón MVVM presenta mejores condiciones, en el indicador en estudio, que el patrón MVC.

Calificación

Para la calificación del indicador Promedio de Tiempo de la página se utilizará los valores establecidos en la Tabla III-VI el valor del peso ponderado máximo, por lo tanto en la Tabla III-XXII se explica la calificación para cada patrón para el indicador en estudio.

Para esto se analiza los promedios observados en la Tabla III-XX

TABLA III-XXII CALIFICACIÓN POR PATRÓN DEL INDICADOR PORCENTAJE DE TIEMPO DEL PROCESADOR

Patrona	Valor Cuantitativo
MVVM	10
MVC	5.42

Como el patrón MVVM presenta mejores condiciones en el indicador en estudio que MVC; MVVM se lo califica con el valor más alto que es 10 y a MVC se lo calificar con un valor inversamente proporcional al de MVVM.

1.3.2. Mbyte disponible:

Para el análisis estadístico para el indicador Mbyte disponible se tiene:

H₀: No Existe una diferencia significativa entre el porcentaje de Mbyte disponible para el patrón MVC con respecto al patrón MVVM.

H₁: Existe una diferencia significativa entre porcentaje Mbyte disponible para el patrón MVC con respecto al patrón MVVM.

En la Tabla III-XXIII muestra el resumen de las mediciones de los patrones para el indicador en estudio.

TABLA III-XXIII RESUMEN RESULTADOS DE LA MEDICIÓN DEL INDICADOR MBYTE DISPONIBLES

	Frecuencia	Promedio	Error Relativo	Sx
MVVM	241	2136.394191	15.53211857	241.1233221
MVC	241	2202.556017	13.57566871	210.7510527

En la tabla III-XXIV se muestra los datos obtenidos de la prueba U de Mann-Whitney:

TABLA 3III-XXIV ESTADÍSTICOS DE CONTRASTE MBYTE DISPONIBLES

	ProcesorTime
U de Mann-Whitney	1494.000
W de Wilcoxon	30414.000
Z	-17.971
Sig. asintót. (bilateral)	2.685E-16

El P-Valor para la prueba U de Mann-Whitney como se observa en la Tabla III-XXIV es de 2.685E-16, por lo tanto se deduce que el valor de significancia asintótica es menor al valor alfa ($2.685E-16 < 0,05$) por lo tanto se concluye que se rechaza la hipótesis nula y se acepta la hipótesis alterna es decir:

“Existe una diferencia significativa entre Mbyte Disponibles para el patrón MVC con respecto al patrón MVVM”.

Debido a que el indicador MBYTE DISPONIBLES es un valor promediado medido en Mbyte la relación entre ambos patrones es proporcional, es decir a mayor cantidad mayor es su rendimiento por tal motivo el patrón MVVM presenta mejores condiciones, en el indicador en estudio, que el patrón MVC.

Calificación

Para la calificación del indicador Promedio de Tiempo de la página se utilizará los valores establecidos en la Tabla III-VI el valor del peso ponderado máximo, por lo tanto en la Tabla III-XXV se explica la calificación para cada patrón para el indicador en estudio.

Para esto se analiza los promedios observados en la Tabla III-XXIII

TABLA III-XXV CALIFICACIÓN POR PATRÓN DEL INDICADOR MBYTE DISPONIBLE

Patrón	Valor Cuantitativo
MVVM	9.7
MVC	10

Como el patrón MVVM presenta mejores condiciones en el indicador en estudio que MVC; MVVM se lo califica con el valor más alto que es 10 y a MVC se lo calificar con un valor inversamente proporcional al de MVVM.

Evaluación de resultados para la variable Rendimiento

Para evaluar el parámetro de rendimiento se utiliza las formulas 1, 2, 3 previamente definidas en el capítulo de análisis de resultados.

La calificación máxima del parámetro de comparación se establece mediante la suma de los valores máximos de calificación de cada indicador del parámetro de rendimiento, es decir el parámetro de rendimiento posee 6 indicadores entonces:

$$C_{MAX} = \sum_{i=0}^n VM_i = 15 + 10 + 10 + 15 + 10 + 10 = 70.$$

Para obtener la calificación numérica para el patrón MVVM se realiza una suma de las calificaciones de cada indicador del parámetro de rendimiento:

$$C_{MVVM} = \sum_{i=0}^n V_i = 15 + 10 + 10 + 15 + 10 + 9.7 = \mathbf{69,7\%}.$$

Para obtener la calificación numérica para el patrón MVC se realiza una suma de las calificaciones de cada indicador del parámetro de rendimiento:

$$C_{MVC} = \sum_{i=0}^n V_i = 0.74 + 7.96 + 0.17 + 3.65 + 5.42 + 10 = \mathbf{27.94\%}.$$

Los resultados del cumplimiento del parámetro de rendimiento se muestran en la Figura III-6

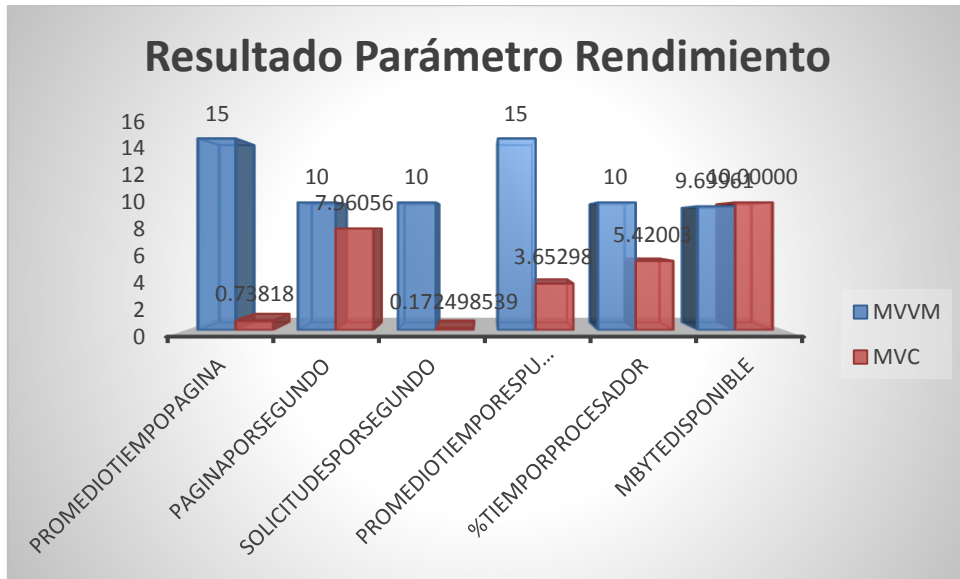


FIGURA III-6 RESULTADO PARÁMETRO RENDIMIENTO

En la Figura III-7 muestra el porcentaje de cumplimiento del parámetro de rendimiento

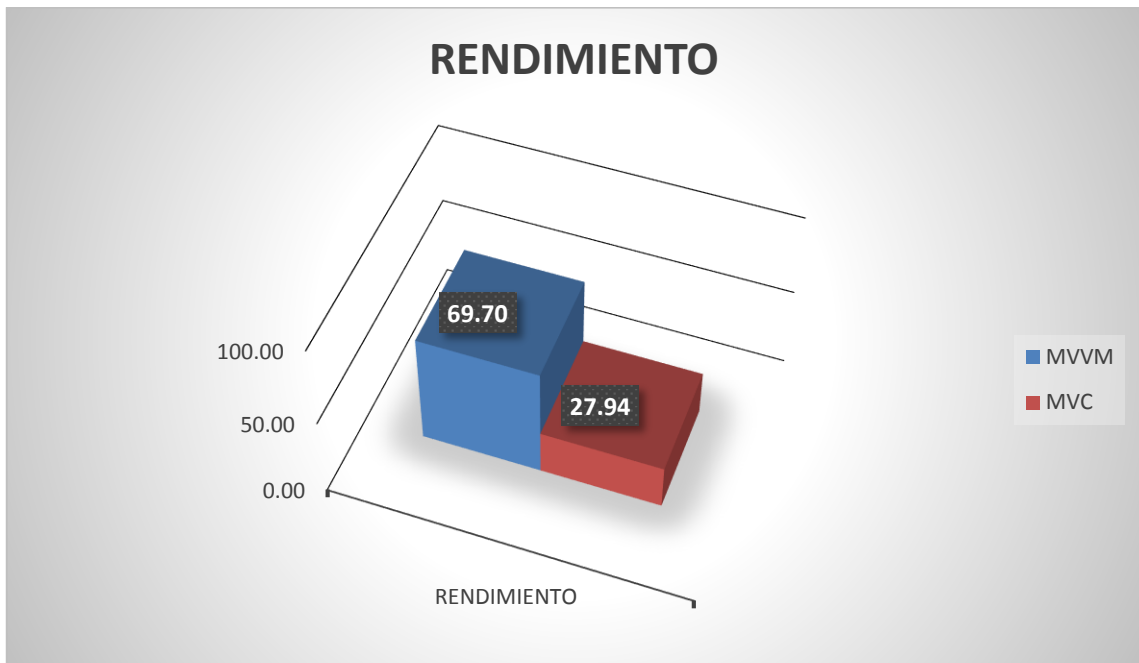


FIGURA III-7 RESULTADO EN PORCENTAJE DEL PARÁMETRO DE RENDIMIENTO

Interpretación de resultados

En la Tabla III-VII permite establecer la calificación cuantitativa y cualitativa mediante el resultado en porcentajes obtenidos para el parámetro de rendimiento para los parámetros MVVM y MVC.

Los resultados obtenidos para el parámetro MVVM es de 69.7% lo que es equivalente a excelente y 5 para la calificación cualitativo y cuantitativo respectivamente. Mientras que para el patrón MVC el resultado es de 27.94% lo que es equivalente a malo y 1 para la calificación cualitativo y cuantitativo respectivamente.

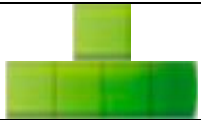




3.5.3 ANÁLISIS DEL PARÁMETRO 2: PRODUCTIVIDAD

Para realizar el análisis de productividad se utilizara las formulas 1, 2, 3, 4, 5. Los índices a utilizar se encuentran detallado en la Tabla III-IV Índices de los parámetros de productividad, además la herramienta a utilizar para las mediciones es el IDE Visual Studio 2012 Ultimate.

Los índices, líneas de código y complejidad del parámetro de productividad se medirán del módulo de gestión de compras debido a que este módulo es el más utilizado por los usuarios del sistema.

Los niveles de cumplimiento que serán utilizadas en el desarrollo de esta investigación, de esta forma se presentan mejor productividad de un patrón con respecto a la otra se presentan en la Tabla III-XXVI.

TABLA 3III-XXVI NIVEL DE CUMPLIMIENTO DE PRODUCTIVIDAD

VALORACIÓN	CALIFICACIÓN	FORMA GRAFICA	VALOR	DESCRIPCIÓN
Excelente	>25% y <=30%		5	Cumple los requerimientos en su totalidad
Muy Buena	>20% y <=25%		4	Cumple los requerimientos en su mayor parte
Buena	>15% y <=20%		3	Cumple los requerimientos aceptablemente
Regular	>10% y <=15%		2	Cumple los requerimientos en mínima parte
Malo	>5% y <=10%		1	No cumple los requerimientos

2.1. Productividad

Para este parámetro se realizó las pruebas en los siguientes indicadores, el resultado de las dichas pruebas se encuentra en la sección de Anexos II y IV para los patrones MVVM y MVC respectivamente.

Líneas de código:

Índice que permite medir la cantidad de líneas de código para la realización de una consulta en específico. En la Tabla III-XVII se muestra el número de Líneas de Código del patrón MVC y MVVM del módulo de gestión de compras, usando la función de cálculo de métricas de código de Visual Studio.

TABLA III-XXVII RESULTADO DE LA MEDICIÓN DE LÍNEAS DE CÓDIGO

Patrón	MVVM	MVC
Líneas de Código	68	223

Calificación

Para la calificación del indicador Líneas de Código se utilizará los valores establecidos en la Tabla III-VI el valor del peso ponderado máximo, por lo tanto en la Tabla III-VIII se explica la calificación para cada patrón para el indicador en estudio.

Para esto se analiza las cantidades observados en la Tabla III-XVII

TABLA 3III-XXVIII CALIFICACIÓN POR PATRÓN DEL INDICADOR LÍNEAS DE CÓDIGO

Patrón	Valor Cuantitativo
MVVM	10
MVC	3.04

Como el patrón MVVM presenta mejores condiciones en el indicador en estudio que MVC; MVVM se lo califica con el valor más alto que es 10 y a MVC se lo calificar con un valor inversamente proporcional al de MVVM.

2.1.1. Complejidad

Índice que permite medir la facilidad o complejidad de entendimiento del código. En la Tabla III-XIX se muestra el número de Líneas de Código del patrón MVC y MVVM del módulo de gestión de compras, usando la herramienta CodeMaid de Visual Studio.

TABLA III-XXIX RESULTADO DE LA MEDICIÓN DE COMPLEJIDAD

Patrón	MVVM	MVC
Complejidad	3	8

Calificación

Para la calificación del indicador de Complejidad se utilizará los valores establecidos en la Tabla III-VI el valor del peso ponderado máximo, por lo tanto en la Tabla III-XXX se explica la calificación para cada patrón para el indicador en estudio.

Para esto se analiza las cantidades observados en la Tabla III-XIX

**TABLA III-XXX CALIFICACIÓN POR PATRÓN DEL INDICADOR
LÍNEAS DE CÓDIGO**

Patrona	Valor Cuantitativo
MVVM	15
MVC	5.62

Como el patrón MVVM presenta mejores condiciones en el indicador en estudio que MVC; MVVM se lo califica con el valor más alto que es 15 y a MVC se lo calificar con un valor inversamente proporcional al de MVVM.

2.1.2. Disponibilidad de Información

La disponibilidad de información se ha valorado a través de consultas, con lo que se ha comprobado que existe mayor cantidad de información para el patrón MVC en comparación con el patrón MVVM.

La cantidad de fuentes de información se ha evaluado obteniendo un promedio de los resultados obtenidos mediante la búsqueda en español y la búsqueda en cualquier idioma a cuyo resultado se ha asignado la valoración máxima al resultado más alto.

La cantidad promedio de información para los patrones es el siguiente:

Model Vista Controlador (MVC): 8869000.

Modelo Vista Modelo de Vista (MVVM): 680600;

TABLA III-XXXI RESULTADO DE LA MEDICIÓN DE DISPONIBILIDAD DE INFORMACIÓN

Patrón	MVVM	MVC
Complejidad	680600	8869000

Calificación

Para la calificación del indicador Disponibilidad de Información se utilizará los valores establecidos en la Tabla III-VI el valor del peso ponderado máximo, por lo tanto en la Tabla III-XXXII se explica la calificación para cada patrón para el indicador en estudio.

Para esto se analiza las cantidades observados en la Tabla III-XXXI

TABLA III-XXXII CALIFICACIÓN POR PATRÓN DEL INDICADOR LÍNEAS DE CÓDIGO

Patrón	Valor Cuantitativo
MVVM	0.38
MVC	5

Como el patrón MVC presenta mejores condiciones en el indicador en estudio que MVVM; MVC se lo califica con el valor más alto que es 5 y a MVVM se lo calificar con un valor inversamente proporcional al de MVC.

3.5.3.1 Evaluación de resultados para el parámetro Productividad

Para evaluar el parámetro de productividad se utiliza las diferentes calificaciones obtenidas en las diferentes mediciones de cada indicador propuesto para el parámetro y las formulas 1,2 y 3 expuestas en el capítulo anterior.

Los resultados que se obtengan se presentaran en forma numérica y grafica mediante estos se realiza su respectiva interpretación.

La calificación máxima del parámetro de comparación se establece mediante la suma de los valores máximos de calificación de cada indicador. Por lo tanto:

$$C_{MAX} = \sum_{i=0}^n VM_i = 10 + 15 + 5 = 30$$

La calificación del patrón MVVM en el parámetro de productividad se calcula mediante las sumatoria de las calificaciones de los índices de líneas de código, complejidad y disponibilidad de información. Como resultado tenemos:

$$C_{MVVM} = \sum_{i=0}^n V_i = 10 + 15 + 0.38 = \mathbf{25.38\%}$$

La calificación para el patrón MVC se calcula de la misma forma que para el patrón MVVM. Como resultado tenemos:

$$C_{MVC} = \sum_{i=0}^n V_i = 3.04 + 5.625 + 5 = \mathbf{13.67\%}$$

Los resultados de cumplimiento de los patrones de servicios web se expresan en la Figura III-8:

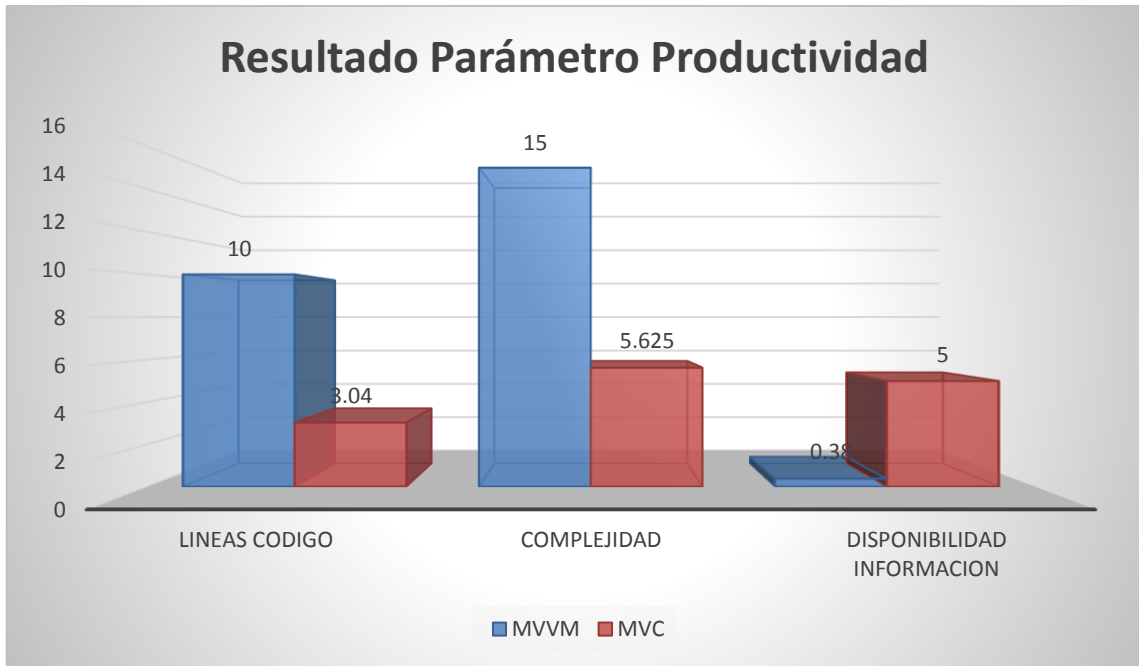


FIGURA III-8 RESULTADO ÍNDICES DE PRODUCTIVIDAD

Los valores en porcentaje del parámetro de productividad se presentan en la Figura III-9

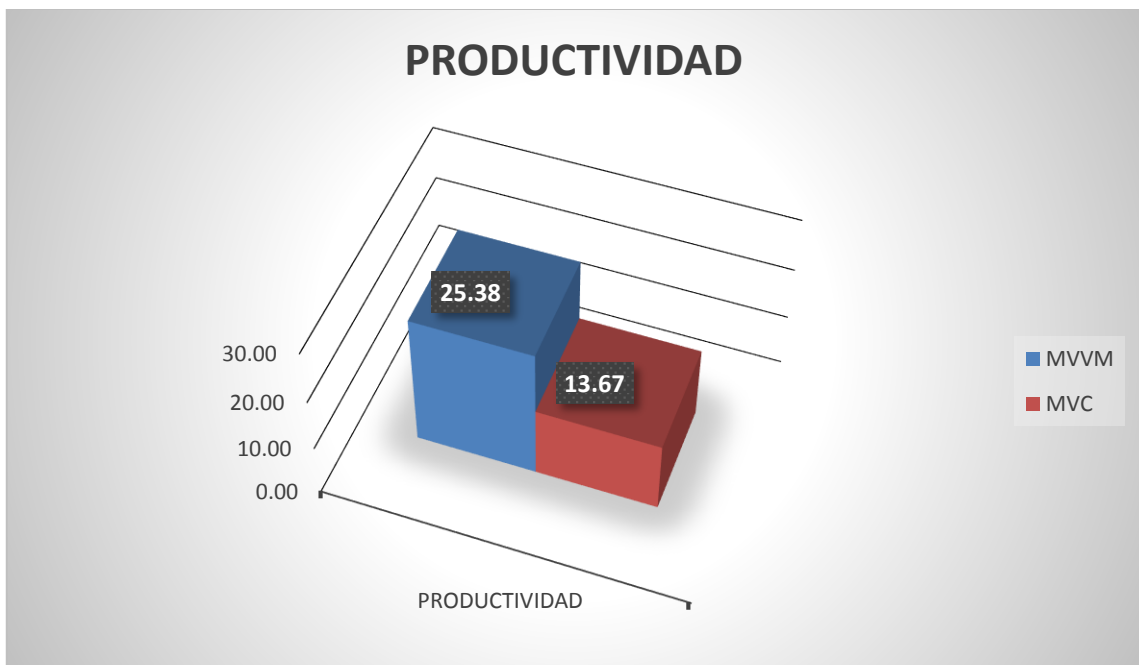


FIGURA III-9 PORCENTAJE DE CUMPLIMIENTO PARÁMETRO PRODUCTIVIDAD

3.5.3.2 Interpretación de resultados

En la tabla III-XXXI se ha determinado el valor cualitativo, cuantitativo y gráfico de acuerdo a porcentajes que se obtuvieron en las mediciones de cada índice de productividad.

Los resultados obtenidos dan a conocer que el patrón MVC posee un 25.38 % en su nivel de cumplimiento lo que es equivalente a Excelente y su calificación cuantitativa es de 5, a diferencia del patrón MVVM que posee un 13.67 % en su nivel de cumplimiento lo que es equivalente a Regular su calificación cuantitativa es de 2.

3.6 DEMOSTRACIÓN DE LA HIPÓTESIS

La Hipótesis Planteada es:

La arquitectura Modelo Vista Modelo de Vista presenta mejores condiciones en rendimiento y productividad para el desarrollo de aplicaciones de software profesionales.

Con los datos obtenidos de la Tabla III-XXXIII haciendo uso de la estadística inferencial se comprueba que el patrón MVVM presenta mejores condiciones de rendimiento y productividad cumpliendo con el 95.1% de sus niveles de cumplimiento con respecto al patrón MVC que cumple con el 41.6% de sus niveles de cumplimiento, por lo tanto, la hipótesis planteada queda comprobada con nivel de significancia al 95% y porcentaje de error al 5%.

En la Tabla III-XXXIII se muestra en resumen las calificaciones de cada parámetro de comparación así como su porcentaje y su calificación máxima.

Se obtuvo en porcentaje de la medición de los indicadores del patrón MVVM para el parámetro de rendimiento un porcentaje de 69.70% en las mediciones de cada indicador,

con respecto al patrón MVC que muestra un valor de 27.94% para el parámetro de rendimiento.

Para el parámetro de productividad la medición de sus indicadores propuso un valor del 25.38% y 13.67% para los patrones MVVM y MVC respectivamente.

TABLA III-XXXIII VALORES DE LOS NIVELES DE CUMPLIMIENTO DE CADA PARÁMETRO E INDICADOR PARA MVC Y MVVM

Variable	Parámetro	Indicador	MVVM	MVC	Calificación Máxima
RENDIMIENTO	Indicadores Claves	Promedio de Tiempo de Página	15	0.73	15
		Páginas por segundo	10	7.96	10
		Solicitudes/segundo	10	0.17	10
	Tiempo de Respuesta de la pagina	Promedio de tiempo de Respuesta	15	3.65	15
	Controladores y agentes	Porcentaje del tiempo del procesador	10	5.42	10
		Mbyte disponible	9.69	10	10
PRODUCTIVIDAD	Curva de aprendizaje	Líneas de Código	10	3.04	10
		Complejidad	15	5.62	15
		Información Disponible	0.38	5	5
TOTAL			95.1	41,6	100

La Figura III-10 muestra los porcentajes obtenidos por los patrones MVC y MVVM para los parámetros de rendimiento y productividad. Se observa que en el patrón MVVM cumple los niveles de aceptación con el 95,1% a diferencia del patrón MVC que su nivel de cumplimiento es del 41,6%.

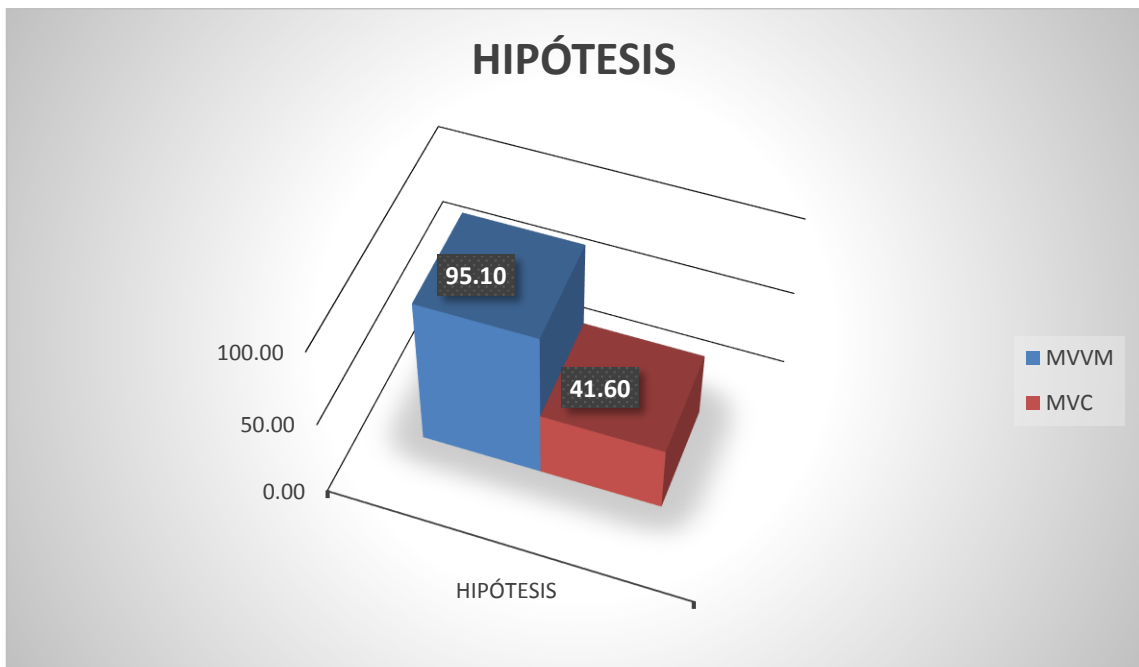


FIGURA III-10 NIVEL DE CUMPLIMIENTO DE LA HIPÓTESIS

CAPÍTULO IV

4 SISTEMA AUTOMATIZADO DE INVENTARIO Y FACTURACIÓN

En el presente capítulo se detalla el desarrollo con el patrón MVVM para el sistema web de Gestión de ventas para la empresa VIDRIALUM, usando la metodología Scrum.

Esta metodología se seleccionó debido a que:

- No hay fases. Éstas pasan a ser tareas que se ejecutan cuando se necesitan. No se hace primero el diseño del concepto o los requisitos, más tarde el análisis, luego el desarrollo, etc.
- Se empieza a trabajar sin el detalle cerrado de lo que se va a producir. Se parte de la visión general. El descubrimiento paulatino durante el desarrollo, y las circunstancias que se irán produciendo en el entorno, dibujarán el detalle de forma paralela al desarrollo. (17)
- Posee un equipo multidisciplinario y auto organizado.
- Adaptativo a los cambios.

En la Figura IV-11 se muestra el proceso que lleva acabo la metodología Scrum

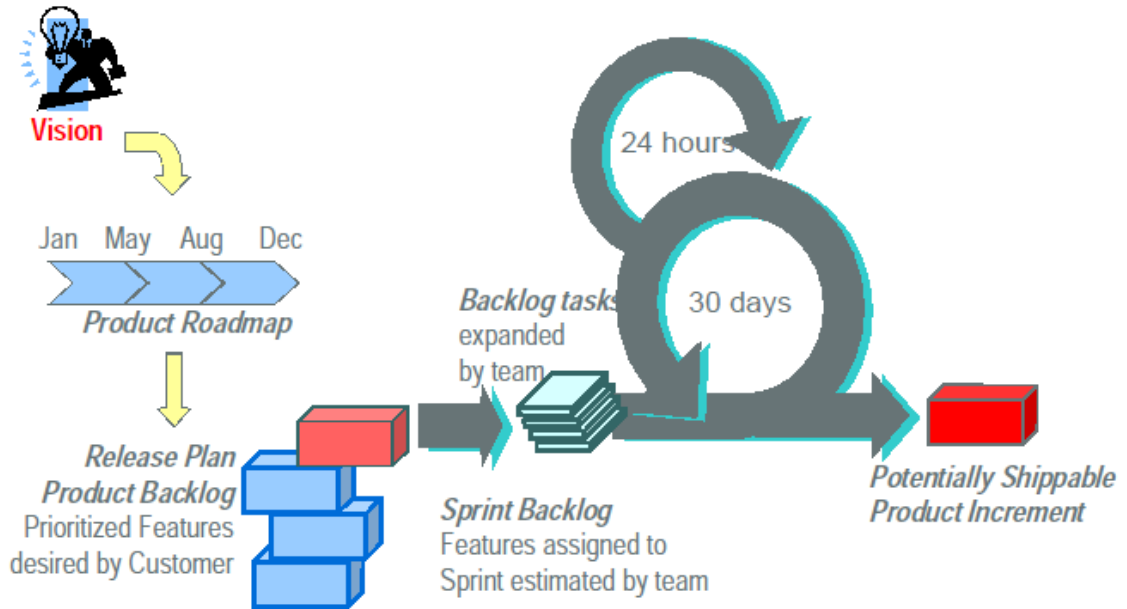


FIGURA IV-11 PROCESO SCRUM

Fuente: CollabNet

Se usará la estructura ágil de Scrum propuesto por Juan Palacio en su libro “Flexibilidad con Scrum” (6) está compuesto de 5 etapas:

1. Concepto.
2. Especulación.
3. Exploración.
4. Revisión.
5. Cierre.

4.1 CONCEPTO

Se realizó para la empresa VIDRIALUM que se encuentran en la ciudad de Riobamba su dirección es Avenida la Prensa 18-95 y Rey Cacha, en el cual se desarrolló y se

implementó el sistema de gestión para empleados, proveedores, clientes, facturación de ventas y compras de productos, de inventario y Reportes.

Las herramientas utilizadas para el desarrollo son:

- Microsoft Visual Studio 2012 Ultimate.
- Microsoft SQL Server 2008.
- Microsoft Project.

4.1.1 INFORMACIÓN GENERAL

El sistema web desarrollado como caso práctico de esta investigación usando el patrón MVVM es para la empresa VIDRIALUM.

Empresa riobambeña que se dedica a la distribución de vidrio y aluminio. El aluminio les entrega la empresa ecuatoriana CEDAL. Además también se dedica a la venta de suministros para la construcción con aluminio y vidrio.

Su matriz se encuentra ubicada en la ciudad de Riobamba en las calles Av. La Prensa y Rey Cacha, la sucursal en la ciudad de Guaranda. El organigrama estructural de la empresa se detalla en la Figura IV-12 Organigrama VIDRIALUM.

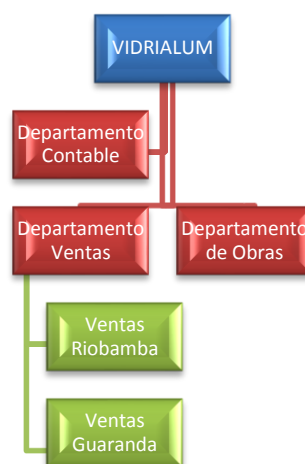


FIGURA IV-12 ORGANIGRAMA VIDRIALUM

Es una empresa consolidada con 30 años de experiencia en el mercado de ventas de aluminio y vidrio prestando sus servicios a empresas públicas y privadas. La compañía especializada en ofrecerle soluciones integrales en Aluminio y Vidrio para el hogar y la empresa, cuentan con una amplia gama de productos nacionales e importados, lo cual nos permite ofrecerle varios diseños y calidades, convirtiéndonos en una de las mejores alternativas del sector. El gerente propietario es el Ingeniero David Santillán Gerente

4.1.2 DESCRIPCIÓN DEL PRODUCTO

El sistema informático que se ha propuesto para desarrollar es el Sistema SADIF Sistema Automatizado de Inventario y Facturación, el mismo que ayudará a llevar un mejor control de las diferentes actividades que se realizan en la empresa SAHID asociada de la empresa VIDRIALUM acerca de informes de inventario, clientes, ventas, compras, Utilidad por producto, cierre de caja.

Ayudando a la agilización de las diferentes operaciones cotidianas que se lleve en la empresa, y principalmente a no tener pérdida de la información.

El sistema nos ofrece una plataforma completa para la administración de datos, por lo que el sistema actuara de forma independiente.

4.1.3 FUNCIONES DEL PRODUCTO

Las funciones del sistema web son:

- Administrar Empleados.
- Administrar Clientes.
- Administrar Proveedores.
- Administrar Productos.
- Administrar Ventas.
- Administrar Compras.

- Administrar Precios

4.1.4 CARACTERÍSTICAS DE USUARIO

El sistema cuenta con tres tipos de usuario Final:

- Cajero: personas encargadas de realizar las ventas, debe poseer conocimientos básicos de informática, se considera que debe poseer conocimiento mínimo de uso del sistema “Nivel Operativo”.
- Contadora: personas encargadas de realizar la contabilidad de la empresa posee conocimientos medio de informática y uso avanzado del sistema “Nivel Táctico”.
- Administrador: personas que se encargan de la administración de la empresa posee conocimientos medios de informático y uso avanzado del sistema “Nivel Administrativo”.

4.1.5 SUPUESTOS Y DEPENDENCIAS

El sistema deberá ser instalado en el sistema operativo que distribuye Microsoft, es decir, Windows XP, Vista y/o Windows 7.

Si se utiliza el motor de base de datos SQL server 2008, se utilizara el SQL AGENT y Jobs para ejecutar respaldos de la base de datos la misma que se almacenara en una carpeta en el Disco principal,

4.2 ESPECULACIÓN

En este apartado se expone los requerimientos funcionales, requerimientos de rendimiento y limitaciones de Diseño.

4.2.1 REQUERIMIENTOS FUNCIONALES

Estos requerimientos están divididos en 4 sprint con duración de tres semanas cada uno

Sprint 1

- El sistema debe permitir el ingreso mediante autenticación de usuarios.
- Como administrador o contador deseo ingresar los datos de los empleados.
- Como administrador o contador deseo modificar los datos de los empleados.
- Como administrador, contador o cajero deseo ingresar los datos de los clientes.
- Como administrador, contador o cajero deseo modificar los datos de los clientes.
- Como administrador o contador deseo ingresar los datos de los proveedores.
- Como administrador o contador deseo modificar los datos de los proveedores.
- Como administrador o contador deseo ingresar los datos de los productos.
- Como administrador o contador deseo modificar los datos de los productos.

Sprint 2

- Como administrador o contador deseo poder ingresar una factura de compras.
- Como administrador o cajero deseo poder ingresar una factura de venta.
- Como administrador deseo poder modificar la utilidad de los precios de cada producto identificándolos para la venta como unidad, docena, ciento, millar.
- Como administrador deseo poder realizar búsquedas de productos, empleados, clientes, proveedores.

- Como administrador deseo poder imprimir la venta realizada las veces que sean necesario.

Sprint 3

- Como administrador deseo poder anular Facturas de ventas.
- Como administrador deseo poder cambiar la clave de ingreso al sistema de los empleados.
- Como administrador deseo visualizar un reporte de los Kardex de todos los productos.
- Como administrador deseo visualizar un reporte de los Kardex por producto requerido.
- Como administrador deseo visualizar un reporte de las ventas realizadas por fechas.
- Como administrador deseo visualizar un reporte de compras realizadas por fechas.

Sprint 4

- Como administrador deseo visualizar un reporte de utilidad de los productos vendidos.
- Como administrador deseo visualizar un reporte de inventario por fecha seleccionada.

4.2.2 REQUERIMIENTO DE RENDIMIENTO

- La infraestructura de red así como sus terminales deben cumplir con normas de rendimiento.
- El servidor de base de datos, deberá tener un respaldo apropiado, así como personal técnico listo para cualquier eventualidad.

4.2.3 LIMITACIONES DE DISEÑO

Interfaz clara y precisa cubres los requerimientos de la organización y formularios, ambiente por defecto el Windows.

Se debe garantizar la existencia de hardware que evita algún inconveniente o retraso del mismo.

4.3 EXPLORACIÓN

En este apartado se describirá las tareas realizada durante el periodo de prácticas las mismas que están conformadas por el Product Backlog el mismo que consta de todos los requerimientos del sistema, además de los Sprint las misma que dividen en iteraciones a dichos requerimientos.

4.3.1 PRODUCT BACKLOG

El Product Backlog representa todo lo necesario para desarrollar y lanzar un producto exitoso. Se trata de una lista de todas las características, funciones, tecnologías, mejoras y correcciones de errores que constituyen los cambios que se harán al producto para futuras versiones. (18).

Para el Product Backlog del sistema SADIF se ha determinado las siguientes fechas:

Fecha de inicio: 8 de abril de 2013

Fecha de Fin: 8 de agosto de 2013

En la Tabla IV-I se muestra en detalle el Product Backlog para el proyecto SADIF:

TABLA IV-I PRODUCT BACKLOG PROYECTO SADIF

PRODUCT BACKLOG PROYECTO: SADIF					
ID.	Rango	Puntos	Título	Estado	Iteración
1	1	30	Creación de Capas de infraestructura, acceso a datos, lógica de negocios y presentación	Completo	1
2	2	3	El sistema debe permitir el ingreso mediante autenticación de usuarios.	Completo	1
3	2	4	Como administrador o contador deseo ingresar los datos de los empleados.	Completo	1
4	2	3	Como administrador o contador deseo modificar los datos de los empleados.	Completo	1
5	2	4	Como administrador, contador o cajero deseo ingresar los datos de los clientes.	Completo	1
6	2	3	Como administrador, contador o cajero deseo modificar los datos de los clientes.	Completo	1
7	2	4	Como administrador o contador deseo ingresar los datos de los proveedores.	Completo	1
8	2	3	Como administrador o contador deseo modificar los datos de los proveedores.	Completo	1
9	2	4	Como administrador o contador deseo ingresar los datos de los productos.	Completo	1
10	2	3	Como administrador o contador deseo modificar los datos de los productos.	Completo	1
11	3	10	Como administrador o contador deseo poder ingresar una factura de compras.	Completo	2
12	3	10	Como administrador o cajero deseo poder ingresar una factura de venta.	Completo	2
13	3	7	Como administrador deseo poder modificar la utilidad de los precios de cada producto identificándolos para la venta como unidad, docena, ciento, millar.	Completo	2
14	3	5	Como administrador deseo poder realizar búsquedas de productos, empleados, clientes, proveedores.	Activo	2
15	3	8	Como administrador deseo poder imprimir la venta realizada las veces que sean necesarios.	Activo	2
16	4	8	Como administrador deseo poder anular Facturas de ventas.	Activo	3
17	4	3	Como administrador deseo poder cambiar la clave de ingreso al sistema de los empleados.	Activo	3
18	4	6	Como administrador deseo visualizar	Activo	3

			un reporte de los Kardex de todos los productos.		
19	4	6	Como administrador deseo visualizar un reporte de los Kardex por producto requerido.	Activo	3
20	4	9	Como administrador deseo visualizar un reporte de las ventas realizadas por fechas.	Activo	3
21	4	6	Como administrador deseo visualizar un reporte de compras realizadas por fechas.	Activo	3
22	5	9	Como administrador deseo visualizar un reporte de utilidad de los productos vendidos.	Activo	4
23	5	9	Como administrador deseo visualizar un reporte de inventario por fecha seleccionada.	Activo	4

4.3.2 SPRINT 1

En esta iteración se describirá las tareas que se realizara en la creación de las Capas de infraestructura, acceso a datos, lógica de negocios y presentación para la iteración 1.

Las fechas de inicio y finalización del sprint 1 son:

Fecha de Inicio: 8 de abril de 2013

Fecha de Fin: 26 de abril de 2013

En la Tabla IV-II que se encuentra en la sección de anexos III se muestra las tareas realizadas en cada historia de usuario determinadas para este sprint.

4.3.3 SPRINT 2

En esta iteración se describirá las tareas que se realizara en la creación de las Capas de infraestructura, acceso a datos, lógica de negocios y presentación para la iteración 2.

La fecha de inicio y final de este sprint son:

Fecha de inicio: 29 de abril de 2013

Fecha de Fin: 17 de mayo de 2013

En la Tabla IV-III que se encuentra en la sección de anexos III se muestra las tareas realizadas en cada historia de usuario determinadas para este sprint.

4.3.4 SPRINT 3

En esta iteración se describirá las tareas que se realizara en la creación de las Capas de infraestructura, acceso a datos, lógica de negocios y presentación para la iteración 3

La fecha de inicio y final de este sprint son:

Fecha de inicio: 20 de mayo 2013

Fecha de Fin: 7 de junio 2013

En la Tabla IV-IV que se encuentra en la sección de anexos III se muestra las tareas realizadas en cada historia de usuario determinadas para este sprint.

4.3.5 SPRINT 4

En esta iteración se describirá las tareas que se realizara en la creación de las Capas de infraestructura, acceso a datos, lógica de negocios y presentación para la iteración 4.

La fecha de inicio y final de este sprint son:

Fecha de inicio: 10 de junio de 2013

Fecha de Fin: 28 de junio de 2013

En la Tabla IV-V que se encuentra en la sección de anexos III se muestra las tareas realizadas en cada historia de usuario determinadas para este sprint.

4.4 REVISIÓN

El equipo y los usuarios revisan las funcionalidades construidas hasta ese momento. Trabajan y operan con el producto real para determinar su alineación y dirección con el objetivo.

4.4.1 ESPECIFICACIONES DE PERSONAS Y EQUIPAMIENTO PARA EL PROYECTO

Se describe las personas y equipo utilizado para el desarrollo del proyecto su costo de uso y la tasa estándar de precio para el desarrollo del sistema SADIF. En la figura 4-3 se describe las personas y los equipos utilizados para el proyecto


		Nombre del recurso	Tipo	Iniciales	Capacidad máxima	Costo/Uso	Acumu	Tasa estándar
1		Jefe de proyecto	Trabajo	J	100%	\$ 0,00	Fin	\$ 7,50/hora
2		Programador	Trabajo	P	100%	\$ 0,00	Fin	\$ 5,50/hora
3		Desarrollador	Trabajo	D	100%	\$ 0,00	Fin	\$ 5,50/hora
4		Usuario	Trabajo	U	25%	\$ 0,00	Fin	\$ 0,00/hora
5		Un servidor	Trabajo	U	300%	\$ 1,50	Prorrat	\$ 0,00/hora
6		Computadora 1	Trabajo	C	100%	\$ 0,90	Prorrat	\$ 0,00/hora
7		Computadora 2	Trabajo	C	100%	\$ 0,90	Prorrat	\$ 0,00/hora
8		Suministros	Material	S		\$ 0,75	Prorrat	\$ 0,00

FIGURA IV-13 PERSONAL Y EQUIPO PROYECTO SADIF

4.4.2 DESARROLLO DE LA BASE DE DATOS DEL SISTEMA SADIF

Para el sistema SADIF se realizaron los siguientes procedimientos para el desarrollo de la base de datos utilizada:

- Análisis: se realizado un estudio a fondo del proceso de ventas e inventario que representará la base de datos y la aplicación.
- Desarrollo del modelo relacional: está basado en la teoría de conjuntos de las relaciones, haciendo que los datos se estructuren lógicamente en formas de relaciones también conocidas como tablas.
- Diseño de la base de datos: se realizó mediante script Transact-SQL el diseño de las base de datos TestFacin.

En los anexos se presenta el esquema TestFacin, explicando las relaciones y llaves primarias de la base de datos. Además se creó procedimientos almacenados para realizar con más seguridad las tareas hechas por el usuario hacia la base de datos.

4.4.2.1 RESPALDOS DE LA BASE DE DATOS

El procedimiento de sacar respaldos de la base de datos es muy importante ya que se debe a este procedimiento que la información es almacenada en un lugar segura que debido a varios factores la información almacenada en la base de datos no es segura estos factores pueden contribuir al daño de los equipos a la mala utilización del sistemas y específicamente de la base de datos.

Es por esta razón que realizar respaldos de la base de datos de una manera automática es importante y sencillo dependiendo de la base de datos que se utiliza sea esta SQL Express o el SQL Estándar, los dos son motor de base de datos SQL pero su manera de administrar los respaldos son muy diferentes.

A continuación se explicara de manera detallada la manera de realizar respaldos en los dos motores de base de datos antes mencionada.

Desde SQL Server 2005 podemos programar una tarea para realizar backups de una o varias base de datos con la herramienta SQL AGENT Job Scheduling.

Para esto debemos seleccionar el servidor sobre el que vamos a trabajar y después realizamos los siguientes pasos:

- Seleccionamos el SQL Server AGENT expandimos seleccionamos Job pulsamos botón derecho y seleccionamos New Job.
- En la pantalla que nos aparece ubicamos el nombre que se llamara nuestro Job.
- Seleccionamos la pestaña step damos clic en nuevo. Al hacer esto se abrirá una nueva pantalla, en ella añadiremos el nombre, seleccionamos en el campo Type

“Transact-SQL script” en la cual ubicaremos el script que se muestra en los anexos y seleccionamos la base de datos que deseamos programar nuestro step. Pulsaremos el botón OK y el programa volverá a la pantalla anterior.

- Ahora, para configurar cuando queremos que se ejecute nuestra tarea y para esto debemos especificar esto en la pestaña de Horarios o “Schedule”.
- Al realizar esto, se abrirá la pantalla “New Job Schedule” y donde debemos especificar:
 - El nombre del horario.
 - El tipo de planificación en el campo “Schedule Type”.
 - La frecuencia con la que queremos hacer nuestra tarea en las pestañas “Frecuencia” y “Daily frequency”.
 - La duración del Job indicando su fecha de inicio y si se desea su fecha de fin.

4.4.3 PROGRAMACIÓN DEL SISTEMA SADIF

La programación de los sistemas de producción permite determinar los procesos del negocio de la organización bajo estudio, y de describir sus flujos de trabajo mediante diagramas de casos de uso que son identificados y estructurados a partir de las actividades de cada proceso. Además, las reglas del negocio son identificadas e incluidas en un glosario, como parte de la especificación de datos y actividades. Un aspecto destacable es el hecho de que los casos de uso se realizan en paralelo, haciendo más fácil la identificación y especificación de casos de uso adecuados. Este proceso está siendo experimentando en un organismo de tamaño medio de la Administración Autonómica.

DEFINICIÓN DE CASO DE USO Y DIAGRAMAS

Los casos de usos y sus diagramas se los encontrará en la sección Casos de Uso en los Anexos Anexo caso de uso y diagramas.

En los siguientes apartados se detalla de manera literal la función de cada caso de uso.

CASO DE USO 1

El sistema debe permitir el ingreso mediante autenticación de usuarios.

Este caso de uso representa la manera en que los usuarios ya sean estos cajero, contador o administrador ingresan al sistema mediante su cedula y su contraseña, la cedula esta validada en caso de error muestra un mensaje de error de la misma manera si no logra ingresar a la tercera vez el sistema se cierra automáticamente emitiendo un mensaje de error; el caso de uso se expone de manera detallada en el anexo caso de uso autenticación de usuarios.

Se detalla además los diagramas de caso de uso que son las representaciones graficas de los caso de uso que explica de manera gráfica la funcionalidad del requerimiento este diagrama se encuentra detallado en los anexos Diagramas “Diagramas de caso de uso de autenticación”.

CASO DE USO 2

Como administrador o contador deseo ingresar los datos de los empleados.

En este caso de uso representa la manera en que el administrador o contador ingresan los datos de los empleados al sistema dichos datos registrado son: Cedula, nombre, apellido, contraseña, fecha de nacimiento, email, dirección, teléfono, celular, estado civil, cargas, sexo, cargo y salario; donde existen datos que son obligatorios para el sistema en los cuales al no ser asignados el sistema emite un mensaje de advertencia, y

en los campos que no son obligatorios y se los deja vacío el sistema asigna valores por defecto. El caso de uso esta detallado en los anexos Caso de Uso Ingreso del empleado.

En los anexos Diagramas de caso de uso gestión de empleados se detalla el diagrama de caso de uso de una manera más sencilla de entender al usuario.

CASO DE USO 3

Como administrador o contador deseo modificar los datos de los empleados.

En este caso se detalla la manera en que el usuario administrador o contador modifican los datos de empleados, los mismos que deben buscar al empleado ingresado anteriormente seleccionarlo para poder modificar los datos que se desee. El sistema valida datos obligatorios y cedula correcta. El caso de uso se detalla en los anexos caso de uso modificar datos empleados y su diagrama en los anexos diagrama de caso de uso gestión de empleados.

CASO DE USO 4

Como administrador, contador o cajero deseo ingresar los datos de los clientes.

El caso de uso representa la manera en que el usuario administrador, cajero o contador ingresan los datos del cliente llamaremos cliente a la persona que realiza una compra a la empresa, dichos datos son: RUC o Cédula, Nombre y Apellido, teléfono, celular, dirección, existen datos que son obligatorios estos campos están pintados de color tamarindo estos al no ser asignados el sistema emite un mensaje de advertencia, y los campos que no son obligatorios el sistema les asigna valores por defecto. El caso de uso esta detallado en los anexos Caso de uso ingresar datos clientes. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso gestión clientes.

CASO DE USO 5

Como administrador, contador o cajero deseo modificar los datos de los clientes.

En este caso se detalla la manera en que el usuario administrador, cajero o contador modifican los datos de los clientes, los mismos que deben buscar al empleado ingresado anteriormente seleccionarlo para poder modificar los datos que se desee. El sistema valida datos obligatorios y cedula correcta. El caso de uso esta detallado en los anexos Caso de uso modificar datos clientes. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso gestión clientes.

CASO DE USO 6

Como administrador o contador deseo ingresar los datos de los proveedores.

El caso de uso representa la manera en que el usuario administrador o contador ingresan los datos del proveedor llamaremos proveedor a la persona a la que se le realiza una compra para la empresa, dichos datos son: RUC o Cédula, Nombre y Apellido, teléfono, celular, dirección, Nombre Comercial, fax, email, estado existen datos que son obligatorios estos campos están pintados de color tamarindo estos al no ser asignados el sistema emite un mensaje de advertencia, y los campos que no son obligatorios el sistema les asigna valores por defecto. El caso de uso esta detallado en los anexos Caso de uso ingresar datos proveedores. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso gestión proveedores.

CASO DE USO 7

Como administrador o contador deseo modificar los datos del proveedor.

En este caso se detalla la manera en que el usuario administrador o contador modifican los datos de los proveedores, los mismos que deben buscar al proveedor ingresado anteriormente seleccionarlo para poder modificar los datos que se desee. El sistema

valida datos obligatorios y cedula correcta. El caso de uso esta detallado en los anexos Caso de uso modificar datos proveedor. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso gestión proveedor.

CASO DE USO 8

Como administrador o contador deseo ingresar los datos de los productos.

El caso de uso representa la manera en que el usuario administrador o contador ingresan los datos del producto llamaremos producto al objeto que la empresa se dedica a vender o comprar los datos que se deben ingresar son: Fecha de registro, código, nombre, nombre corto stock máximo y stock mínimo existen datos que son obligatorios estos campos están pintados de color tamarindo estos al no ser asignados el sistema emite un mensaje de advertencia, y los campos que no son obligatorios el sistema les asigna valores por defecto. El caso de uso esta detallado en los anexos Caso de uso ingresar datos productos. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso gestión productos.

CASO DE USO 9

Como administrador o contador deseo modificar los datos de los productos.

En este caso se detalla la manera en que el usuario administrador o contador modifican los datos de los productos, los mismos que deben buscar al producto ingresado anteriormente seleccionarlo para poder modificar los datos que se desee. El sistema valida datos obligatorios y cedula correcta. El caso de uso esta detallado en los anexos Caso de uso modificar datos productos. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso gestión productos.

CASO DE USO 10

Como administrador o contador deseo poder ingresar una factura de compras.

En este caso se detalla la manera en que el usuario administrador ingresa una factura de compra que se realiza a un proveedor de ciertos productos estos productos deben estar almacenados en el sistema antes de ingresar la factura de compra así como el proveedor. Los datos que se debe ingresar son: Numero de factura, fecha de ingreso, Cedula del proveedor, Tipo de pago el mismo que se refiere si el pago se realiza en efectivo, transferencia o cheque, Estado el que se refiere pendiente o pagado la factura, Numero de cheque y Banco, además de todos los productos comprados con su respectivo precio sin impuestos.

Existen datos que se registra automáticamente como es el nombre del proveedor al ingresar el número de cedula y el subtotal, valor neto, IVA y total. El caso de uso esta detallado en los anexos Caso de uso comprar producto. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso comprar producto.

CASO DE USO 11

Como administrador o cajero deseo poder ingresar una factura de venta.

En este caso de uso se detalla la manera en que el administrador o cajero realiza una venta solicitado por un cliente dicha factura consta de los datos del cliente número de factura fecha de venta, el estado de la factura, es decir, si el pago de la factura está pendiente o pagada además del empleado que realiza la venta y lo más importante los productos que desee el cliente estos productos pueden ser vendidos por unidad, docena, ciento y millar.

Existen campos que son obligatorias y se deberá tener cuidado con dichos campos. El caso de uso esta detallado en los anexos Caso de uso ventas. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso ventas.

CASO DE USO 12

Como administrador deseo poder modificar la utilidad de los precios de cada producto identificándolos para la venta como unidad, docena, ciento, millar.

En este caso de uso se detalla la manera en la que el administrador después de ingresar una factura de compras ingresa la utilidad de los precios por unidad de venta estas unidades son: Unidad, Docena, Ciento y Millar. El precio ingresado es sin impuesto y se basa con el precio promedio de los productos, este precio se obtiene del Kardex, además se obtiene el nombre del último proveedor del producto y el precio de compra este método se lo realiza producto por producto. El caso de uso esta detallado en los anexos Caso de uso modificar utilidad precios. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso modificar utilidad precios.

CASO DE USO 13

Como administrador deseo poder realizar búsquedas de productos, empleados, clientes, proveedores.

En este caso de uso se detalla la manera en la que los usuarios del sistema ya sean estos administradores, contador o cajero pueden realizar búsquedas de empleados clientes, proveedores y productos mediante filtros que el usuario selecciona estos filtros se basan en el nombre, cedula, apellido o código. El caso de uso esta detallado en los anexos Caso de uso buscar. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso buscar.

CASO DE USO 14

Como administrador deseo poder imprimir la venta realizada las veces que sean necesarios.

El caso de uso detalla la manera en la que el usuario puede imprimir una factura de venta almacenada las veces que desee solo con el número de factura. El caso de uso esta detallado en los anexos Caso de uso imprimir factura. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso imprimir factura.

CASO DE USO 15

Como administrador deseo poder anular Facturas de ventas.

El caso de uso muestra como el administrador puede anular facturas de venta solo ingresando el número de factura que desee anular, el sistema se encarga de aumentar la cantidad de productos vendidos para evitar inconsistencias los inventarios. El caso de uso esta detallado en los anexos Caso de uso anular factura. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso anular factura.

CASO DE USO 16

Como administrador deseo poder cambiar la clave de ingreso al sistema de los empleados

El caso de usos explica la manera en la que el administrador cambia la clave de los empleados mediante la cedula del empleado y se le asigna una nueva clave para el uso del sistema; El caso de uso esta detallado en los anexos Caso de uso modificar clave empleados.

CASO DE USO 17

Como administrador deseo visualizar un reporte de los Kardex de todos los productos

El caso de uso muestra la manera en la que el administrador puede observar, imprimir o archivar el Kardex de los movimientos contables de la empresa de cada uno de los productos que la empresa ha realizado algún tipo de transacción ya sea esta de compra venta o devolución en venta; El caso de uso esta detallado en los anexos Caso de uso reporte Kardex. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso reporte Kardex.

CASO DE USO 18

Como administrador deseo visualizar un reporte de los Kardex por producto requerido.

El caso de uso muestra la manera en la que el administrador puede observar, imprimir o archivar el Kardex de los movimientos contables de la empresa de uno de los productos que la empresa ha realizado algún tipo de transacción ya sea esta de compra venta o devolución en venta, dicho reportes se mostrara cuando el administrador ingrese el código del producto que desee. El caso de uso esta detallado en los anexos Caso de uso reporte Kardex por producto requerido. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso reporte Kardex por producto requerido.

CASO DE USO 19

Como administrador deseo visualizar un reporte de las ventas realizadas por fechas.

El caso de uso muestra el informe que el administrador visualiza ventas realizadas en un rango de fechas, dicho informe le sirve al administrador para cuadrar el dinero en ventas

del sistema con el dinero que el empleado le entrega esto se lo conoce como cuadro de caja. El caso de uso esta detallado en los anexos Caso de uso reporte venta por fecha. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso reporte ventas fecha.

CASO DE USO 20

Como administrador deseo visualizar un reporte de compras realizadas por fechas.

El caso de uso muestra el informe que el administrador visualiza compras realizadas en un rango de fechas, dicho informe le sirve al administrador para saber las compras que ha realizado. El caso de uso esta detallado en los anexos Caso de uso reportes compra fecha. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso reporte compra fecha.

CASO DE USO 21

Como administrador deseo visualizar un reporte de utilidad de los productos vendidos.

El caso de uso muestra el informe en la que se detalla la cantidad de productos vendidos, el precio de compra el precio de venta la utilidad en porcentaje y la utilidad en dinero este informe le sirve al administrador saber que producto es el más vendido y cual producto nos ofrece mejor utilidad de venta para próximas compras y ventas. El caso de uso esta detallado en los anexos Caso de uso reporte utilidad productos. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso reporte utilidad productos.

CASO DE USO 22

Como administrador deseo visualizar un reporte de inventario por fecha seleccionada

El caso de uso muestra la manera en la que el administrador puede visualizar el total de productos que posee la empresa en un rango de fechas dicho informe le sirve para tener un control de los productos existentes en el sistema con los reales. El caso de uso esta detallado en los anexos Caso de uso reporte inventario por fecha. El diagrama de caso de uso esta detallado en los anexos Diagrama de caso de uso reporte inventario por fecha.

4.4.4 CAPACITACIÓN SOBRE EL USO DEL SISTEMA

Para el sistema desarrollado en las en la empresa VIDRIALUM fue necesaria la respectiva capacitación al personal que interactúa con la aplicación, razón por la cual se realizaron cursos de capacitación en los cuales tenemos los siguientes detalles:

DETALLES

Destinatarios: Personal Laboral de la empresa VIDRIALUM

Fechas Del 1 al 31 de julio del 2013

Horario: Lunes a Viernes 15:00 a 18:00 horas

4.4.5 PRUEBAS E INSTALACIÓN

Para el sistema desarrollado se realizó la instalación y pruebas que las realizaron el personal laboral de la empresa VIDRIALUM, estos procesos nos permiten manejar errores que en el desarrollo no se tuvieron en cuenta; en dichos procesos tenemos los siguientes detalles:

Fecha: del 1 de agosto al 10 de agosto de 2013

CONCLUSIONES

1. El estudio comparativo entre los patrones MVVM y MVC ha permitido determinar que el patrón MVVM presenta mejores condiciones de rendimiento y productividad con respecto al patrón MVC ambos patrones desarrollados con tecnología .Net.
2. El modelo y la vista cumple las mismas funciones tanto para el patrón MVVM como para el patrón MVC; la diferencia más distintiva del patrón MVC la acción del usuario afecta directamente al controlador mientras que en el patrón MVVM la acción afecta directamente a la vista.
3. El patrón MVVM presenta mejores condiciones de rendimiento y productividad cumpliendo con el 95.1% de sus niveles de cumplimiento con respecto al patrón MVC que cumple con el 41.6% de sus niveles de cumplimiento con esto la hipótesis queda demostrada con un nivel de significancia del 95% y un porcentaje de error al 5%.
4. Debido al cumplimiento más alto de niveles de rendimiento y productividad se seleccionó el patrón MVVM para el desarrollo del sistema SADIF para la empresa VIDRIALUM.
5. El desarrollo del sistema SADIF usando el patrón MVVM permitió un mejor control sobre el inventario, ventas y compras para la empresa VIDRIALUM.

RECOMENDACIONES

1. Se recomienda una investigación exhaustiva acerca del patrón MVVM y sus diferentes herramientas para el desarrollo en específico LigthSwitch.
2. El resultado de la Tesis: “ESTUDIO COMPARATIVO DE LOS PATRONES PARA INTERFACES DE USUARIO MVVM Y MVC. APLICADO AL DESARROLLO DEL SITIO DE GESTIÓN DE VENTAS PARA VIDRIALUM” se use como line base para estudios posteriores sea para el patrón MVVM o MVC.
3. La creación de sitios web usando el patrón MVVM debido a que posee mejores niveles de rendimiento y productividad en aplicaciones .Net.
4. Se recomienda identificar correctamente las ventajas que tiene el Controlador del patrón MVC y el ModelView del patrón MVVM, con la finalidad de comprender las diferencia entre ambos.
5. El uso del patrón MVVM con la herramienta LigthSwitch se puede utilizar para el desarrollo de cualquier proyecto con un tiempo de entrega corto.

RESUMEN

La investigación compara los patrones para interfaces de usuario Modelo Vista Controlador (MVC) y el Modelo Vista Modelo de Vista (MVVM) usando tecnologías .Net para el desarrollo del sistema para gestión de ventas y compras para la empresa VIDRIALUM. La utilización del método científico inductivo para reconocer, obtener y medir los resultado de las variables rendimiento y productividad de las pruebas de los prototipos creados para el patrón MVC y MVVM, llegando a la conclusión que el patrón MVVM presenta mejores prestaciones en comparación con MVC, además se usó las técnicas de observación participativa y juego de rol para la recolección de datos.

Para analizar y desarrollar se utilizó: Sistema operativo: Windows 7 Ultimate de 64 bits, Procesador: Intel® Core™ i5-2430 CPU@ 2.40 GHz, Memoria: 8 Gigabytes, Maquina: Sony VAIO VPCEG25FL y para las pruebas de rendimiento se utilizó: Herramientas de medición de rendimiento Visual Studio 2012 Ultimate.

Los resultados obtenidos y de acuerdo a las mediciones realizadas muestra que: que el patrón MVVM presenta mejores condiciones de rendimiento y productividad cumpliendo con el 95.1% de sus niveles de cumplimiento con respecto al patrón MVC que cumple con el 41.6% de sus niveles de cumplimiento referente al rendimiento y a la productividad con nivel de significancia del 95% y un margen de error del 5%.

Se diseñó el Sistema Automatizado de Inventario y facturación (SADIF) para la empresa VIDRIALUM usando el patrón MVVM y el Framework Ligthswitch, ya que dicho patrón presenta un nivel en rendimiento y productividad más alto que el patrón MVC. Este sistema ofrece un mejor manejo en compras, ventas y un mayor control en el inventario de sus productos.

Palabras Claves: Patrón, MVC, MVVM, Rendimiento, Productividad

SUMMARY

The research compares the patterns for user interfaces Model View Controller (MVC) and Model View View Model (MVVM) using technologies .Net development management system for sales and purchases for the company VIDRIALUM. The use of the inductive scientific method to recognize, collect and measure the outcome variables of performance and productivity testing of prototypes created for the MVC and MVVM pattern, concluding that the MVVM pattern provides better performance compared to MVC, further participant observation techniques and role play to data collection was used.

To analyze and develop used: Operating System: Windows 7 Ultimate 64-bit

Processor: Intel ® Core ™ i5-2430 CPU@2.40 GHz Memory: 8 Gigabytes, Used: Sony VAIO VPCEG25FL and performance tests was used: Performance Measurement Tools Visual Studio 2012 Ultimate.

The results obtained and arrangements made to measurements shows that: the MVVM pattern has better performance and productivity complying with 95.1% of its level of compliance with respect to the MVC pattern that meets the 41.6% of compliance levels regarding performance and productivity.

Automated Inventory and Billing System (SADIF) was designed for the company using the MVVM pattern VIDRIALUM and Lighthswitch Framework, since this pattern has a level of performance and higher productivity than the MVC pattern.

This system offers better handling on purchases, sales and more control in the inventory of your products.

GLOSARIO TÉRMINOS

MVC: Modelo Vista Controlador.

MVVM: Modelo Vista Modelo de Vista.

MVP: Modelo Vista Presentador.

POCO: Plain Old CLR Objects.

GoF: Gans of Four.

HMVC: Hierarchical Model View Controler.

MVA: Modelo Vista Adaptador.

DTO: Data Transfer Object.

MVP: Most Value Professional.

WPF: Windows Presentation Fundation.

XAML: extensible Application Markup Language

APLICACIONES WEB: aplicación web es aquellas aplicaciones que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador.

Desviación Estándar: La desviación estándar o desviación típica es una medida de centralización o dispersión para variables de razón (ratio o cociente) y de intervalo, de gran utilidad en la estadística descriptiva. Se define como la raíz cuadrada de la varianza. Junto con este valor, la desviación típica es una medida (cuadrática) que informa de la media de distancias que tienen los datos respecto de su media aritmética, expresada en las mismas unidades que la variable.

RAM: La memoria principal o RAM (Random Access Memory, Memoria de Acceso Aleatorio) es donde el computador guarda los datos que está utilizando en el momento presente. El almacenamiento es considerado temporal por que los datos y programas permanecen en ella mientras que la computadora este encendida o no sea reiniciada.

HISTORIAS DE USUARIO: Una historia de usuario es una representación de un requerimiento de software escrito en una o dos frases utilizando el lenguaje común del usuario.

HTML: es el lenguaje de marcado predominante para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes.

ANEXOS

ANEXOS PRUEBAS RENDIMIENTO PRODUCTIVIDAD

Grabadora Prueba Rendimiento

The screenshot shows the Web Test Recorder interface. On the left, a list of recorded requests is visible, including URLs like `http://localhost:52479/default.htm` and `http://localhost:52479/TestFacinData.svc/Productoes`. The main area displays the details of a recorded test scenario titled "Gestion Compra".

Gestion Compra *

Numero Factura: 001 Proveedor: 0603748956

Fecha Compra: 24/05/2014 Nombre Apellido: prueba prueba

Hora Compra: 19:38 Razon Social: los alamos uno

Estado: PAGADO Direccion: los alamos uno

Tipo Pago: EFECTIVO

Compra Detalles

PRODUCTO	NOMBRE PRODUCTO	CANTIDAD	PRECIO UNIDAD	TOTAL PRODUCTO
☺ C.RE.01	Rem cel nat.	1000	0,01	10,00
☺ C.RE.02	Rem cel br.	1000	0,02	20,00
☺ C.RE.03	Rem Arc 5/32 x 1/2 Nat	500	0,01	5,00
☺ C.RE.04	Rem 1/8 nat.	1500	0,02	30,00
☺ C.RE.05	Rem 1/8 br.	755	0,01	7,55

Subtotal: 72,55 Descuento: 0 Valor Neto: 72,55 Valor Iva: 8,71 Total Pago: 81,26

Buttons: Guardar, Actualizar

Ejecutar Prueba de Rendimiento

The screenshot shows the Microsoft Visual Studio interface with a performance test scenario named "WebTest1.webtest" loaded in the Performance Explorer. The scenario is configured with several requests and validation rules.

WebTest1.webtest

- Request: `http://localhost:52479/default.htm`
- Request: `http://go.microsoft.com/fwlink/`
- Request: `http://localhost:52479/Web/SalesSystemSupport.Client.xap`
- Request: `http://localhost:52479/TestFacinData.svc/Microsoft_LightSwitch_GetCanInformation`
- Request: `http://localhost:52479/Web/Microsoft-LightSwitch-Security-ServerGenerated-Implementation-AuthenticationService.svc/binary/GetAuthenticationInfo`
- Request: `http://localhost:52479/TestFacinData.svc/Proveedores`
- Request: `http://localhost:52479/TestFacinData.svc/Microsoft_LightSwitch_GetCanInformation`
- Request: `http://localhost:52479/TestFacinData.svc/Productoes`
- Request: `http://localhost:52479/TestFacinData.svc/Sbatch`
- Request: `http://localhost:52479/TestFacinData.svc/CompraCabeceras(320)`
- Validation Rules:
 - Response URL
 - Response Time Goal

ANEXO I PRUEBAS RENDIMIENTO MVC

DATOS DEL INDICADOR PORCENTAJE DE TIEMPO DE DESCARGA DE LA PÁGINA PARA EL PATRÓN MVC

CategoryName	CounterName	ComputedValue
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	43.08025
LoadTest:Page	Avg. Page Time	10.63597
LoadTest:Page	Avg. Page Time	0.06499997
LoadTest:Page	Avg. Page Time	0.01954166
LoadTest:Page	Avg. Page Time	0.01352631
LoadTest:Page	Avg. Page Time	0.01274193
LoadTest:Page	Avg. Page Time	45.2633
LoadTest:Page	Avg. Page Time	1.431167
LoadTest:Page	Avg. Page Time	3.152462
LoadTest:Page	Avg. Page Time	78.08872
LoadTest:Page	Avg. Page Time	4.071312
LoadTest:Page	Avg. Page Time	35.11979
LoadTest:Page	Avg. Page Time	21.75685
LoadTest:Page	Avg. Page Time	14.23584
LoadTest:Page	Avg. Page Time	12.64529
LoadTest:Page	Avg. Page Time	12.82977
LoadTest:Page	Avg. Page Time	6.374299
LoadTest:Page	Avg. Page Time	10.65872
LoadTest:Page	Avg. Page Time	9.090273
LoadTest:Page	Avg. Page Time	6.895182
LoadTest:Page	Avg. Page Time	2.660752
LoadTest:Page	Avg. Page Time	0.01747348
LoadTest:Page	Avg. Page Time	1.029565
LoadTest:Page	Avg. Page Time	1.689307
LoadTest:Page	Avg. Page Time	3.596462
LoadTest:Page	Avg. Page Time	7.200116
LoadTest:Page	Avg. Page Time	5.317765

DATOS DEL INDICADOR PÁGINA POR SEGUNDO PARA EL PATRÓN MVC

CategoryName	CounterName	ComputedValue
Memory	Pages/sec	0
Memory	Pages/sec	187.0598
Memory	Pages/sec	3132.366
Memory	Pages/sec	4506.565
Memory	Pages/sec	5123.412
Memory	Pages/sec	1692.133
Memory	Pages/sec	119.2711
Memory	Pages/sec	1.597679
Memory	Pages/sec	1.601737
Memory	Pages/sec	1.798029
Memory	Pages/sec	1.602906
Memory	Pages/sec	154.8978
Memory	Pages/sec	52.88758
Memory	Pages/sec	0.1996953
Memory	Pages/sec	1.802431
Memory	Pages/sec	1.598189
Memory	Pages/sec	266.6618
Memory	Pages/sec	1.602154
Memory	Pages/sec	13.22145
Memory	Pages/sec	3.208849
Memory	Pages/sec	1.797202
Memory	Pages/sec	1.602732
Memory	Pages/sec	1.597628
Memory	Pages/sec	5.391483
Memory	Pages/sec	3.205359
Memory	Pages/sec	206.6569
Memory	Pages/sec	107.8943
Memory	Pages/sec	1.602102
Memory	Pages/sec	293.2466
Memory	Pages/sec	57.89075
Memory	Pages/sec	0
Memory	Pages/sec	1.803062
Memory	Pages/sec	52.80289
Memory	Pages/sec	3.011912
Memory	Pages/sec	1.596913
Memory	Pages/sec	1.60331
Memory	Pages/sec	1.79713
Memory	Pages/sec	1.602667
Memory	Pages/sec	1.597679

DATOS DEL INDICADOR SOLICITUDES SEGUNDO PARA EL PATRÓN

MVC

CategoryName	CounterName	ComputedValue
LoadTest:Request	Requests/Sec	0
LoadTest:Request	Requests/Sec	0
LoadTest:Request	Requests/Sec	0
LoadTest:Request	Requests/Sec	0
LoadTest:Request	Requests/Sec	0
LoadTest:Request	Requests/Sec	0
LoadTest:Request	Requests/Sec	68.4
LoadTest:Request	Requests/Sec	1236.8
LoadTest:Request	Requests/Sec	370.6
LoadTest:Request	Requests/Sec	342.6
LoadTest:Request	Requests/Sec	320
LoadTest:Request	Requests/Sec	1.2
LoadTest:Request	Requests/Sec	4.8
LoadTest:Request	Requests/Sec	4
LoadTest:Request	Requests/Sec	6.2
LoadTest:Request	Requests/Sec	412.8
LoadTest:Request	Requests/Sec	131.2
LoadTest:Request	Requests/Sec	106.2
LoadTest:Request	Requests/Sec	873.4
LoadTest:Request	Requests/Sec	560.8
LoadTest:Request	Requests/Sec	197.8
LoadTest:Request	Requests/Sec	93.2
LoadTest:Request	Requests/Sec	168.2
LoadTest:Request	Requests/Sec	270.2
LoadTest:Request	Requests/Sec	322.4
LoadTest:Request	Requests/Sec	249.6
LoadTest:Request	Requests/Sec	478.2
LoadTest:Request	Requests/Sec	518.8
LoadTest:Request	Requests/Sec	441.6
LoadTest:Request	Requests/Sec	438.8
LoadTest:Request	Requests/Sec	320.6
LoadTest:Request	Requests/Sec	546
LoadTest:Request	Requests/Sec	605.8
LoadTest:Request	Requests/Sec	852.4
LoadTest:Request	Requests/Sec	915.8
LoadTest:Request	Requests/Sec	742
LoadTest:Request	Requests/Sec	752.6
LoadTest:Request	Requests/Sec	475.6

DATOS DEL INDICADOR PORCENTAJE DE TIEMPO DE RESPUESTA

PARA EL PATRÓN MVC

CategoryName	CounterName	ComputedValue
LoadTest:Request	Avg. Response Time	0
LoadTest:Request	Avg. Response Time	0
LoadTest:Request	Avg. Response Time	0
LoadTest:Request	Avg. Response Time	0
LoadTest:Request	Avg. Response Time	0
LoadTest:Request	Avg. Response Time	0
LoadTest:Request	Avg. Response Time	25.65777
LoadTest:Request	Avg. Response Time	1.339241
LoadTest:Request	Avg. Response Time	3.287541
LoadTest:Request	Avg. Response Time	3.799822
LoadTest:Request	Avg. Response Time	0.4514975
LoadTest:Request	Avg. Response Time	3.867833
LoadTest:Request	Avg. Response Time	0.01954166
LoadTest:Request	Avg. Response Time	1.6056
LoadTest:Request	Avg. Response Time	0.01274193
LoadTest:Request	Avg. Response Time	3.75281
LoadTest:Request	Avg. Response Time	0.07798628
LoadTest:Request	Avg. Response Time	0.4046685
LoadTest:Request	Avg. Response Time	5.009466
LoadTest:Request	Avg. Response Time	0.2437707
LoadTest:Request	Avg. Response Time	3.287332
LoadTest:Request	Avg. Response Time	2.900944
LoadTest:Request	Avg. Response Time	1.707681
LoadTest:Request	Avg. Response Time	1.658805
LoadTest:Request	Avg. Response Time	1.921341
LoadTest:Request	Avg. Response Time	1.19401
LoadTest:Request	Avg. Response Time	1.351479
LoadTest:Request	Avg. Response Time	1.229486
LoadTest:Request	Avg. Response Time	1.113765
LoadTest:Request	Avg. Response Time	0.4813929
LoadTest:Request	Avg. Response Time	0.004618216
LoadTest:Request	Avg. Response Time	0.1467113
LoadTest:Request	Avg. Response Time	0.3140644
LoadTest:Request	Avg. Response Time	0.5417958
LoadTest:Request	Avg. Response Time	0.6211135
LoadTest:Request	Avg. Response Time	0.4683779
LoadTest:Request	Avg. Response Time	0.03706165

**DATOS DEL INDICADOR PORCENTAJE DE TIEMPO DE PROCESADOR
PARA EL PATRÓN MVC**

CategoryName	CounterName	ComputedValue
Processor	% Processor Time	0
Processor	% Processor Time	57.39646
Processor	% Processor Time	31.82113
Processor	% Processor Time	41.16715
Processor	% Processor Time	46.53807
Processor	% Processor Time	75.08204
Processor	% Processor Time	98.66645
Processor	% Processor Time	95.90489
Processor	% Processor Time	98.52893
Processor	% Processor Time	72.55271
Processor	% Processor Time	47.23819
Processor	% Processor Time	36.12011
Processor	% Processor Time	41.09168
Processor	% Processor Time	32.02583
Processor	% Processor Time	34.73996
Processor	% Processor Time	52.99379
Processor	% Processor Time	40.51696
Processor	% Processor Time	48.02843
Processor	% Processor Time	83.99992
Processor	% Processor Time	45.23784
Processor	% Processor Time	47.95894
Processor	% Processor Time	45.51058
Processor	% Processor Time	46.66642
Processor	% Processor Time	51.76826
Processor	% Processor Time	60.2604
Processor	% Processor Time	52.13653
Processor	% Processor Time	76.22139
Processor	% Processor Time	71.81195
Processor	% Processor Time	66.18481
Processor	% Processor Time	51.79131
Processor	% Processor Time	50.16588
Processor	% Processor Time	52.13653
Processor	% Processor Time	65.08458
Processor	% Processor Time	74.61762
Processor	% Processor Time	87.17943
Processor	% Processor Time	71.67223
Processor	% Processor Time	66.26124
Processor	% Processor Time	50.95518

DATOS DEL INDICADOR MBYTES DISPONIBLE PARA EL PATRÓN MVC

CategoryName	CounterName	ComputedValue
Memory	Available MBytes	1717
Memory	Available MBytes	1695
Memory	Available MBytes	1687
Memory	Available MBytes	1665
Memory	Available MBytes	1648
Memory	Available MBytes	1441
Memory	Available MBytes	1177
Memory	Available MBytes	902
Memory	Available MBytes	473
Memory	Available MBytes	1206
Memory	Available MBytes	1831
Memory	Available MBytes	1891
Memory	Available MBytes	1942
Memory	Available MBytes	1943
Memory	Available MBytes	1943
Memory	Available MBytes	1965
Memory	Available MBytes	2006
Memory	Available MBytes	2007
Memory	Available MBytes	2066
Memory	Available MBytes	2062
Memory	Available MBytes	2064
Memory	Available MBytes	2116
Memory	Available MBytes	2113
Memory	Available MBytes	2110
Memory	Available MBytes	2113
Memory	Available MBytes	2112
Memory	Available MBytes	2108
Memory	Available MBytes	2109
Memory	Available MBytes	2110
Memory	Available MBytes	2103
Memory	Available MBytes	2080
Memory	Available MBytes	2079
Memory	Available MBytes	2083
Memory	Available MBytes	2085
Memory	Available MBytes	2123
Memory	Available MBytes	2125
Memory	Available MBytes	2125
Memory	Available MBytes	2124
Memory	Available MBytes	2124

ANEXO II PRUEBAS PRODUCTIVIDAD MVVM

class CompraDetalle

Namespace: LightSwitchApplication

Most Complex: Cantidad_Validate 2

Methods: 5

Regions: 1

class CompraCabecera

Namespace: LightSwitchApplication

Most Complex: Descuento_Changed 1

Methods: 5

ANEXO III PRUEBAS RENDIMIENTO MVVM

DATOS DEL INDICADOR PORCENTAJE DE TIEMPO DE DESCARGA DE LA PÁGINA PARA EL PATRÓN MVVM

CategoryName	CounterName	ComputedValue
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	2.196254
LoadTest:Page	Avg. Page Time	4.476461
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	0
LoadTest:Page	Avg. Page Time	0.03466652
LoadTest:Page	Avg. Page Time	0.03283323
LoadTest:Page	Avg. Page Time	0.03728563
LoadTest:Page	Avg. Page Time	0.03864444
LoadTest:Page	Avg. Page Time	0.03579485
LoadTest:Page	Avg. Page Time	0.03438595
LoadTest:Page	Avg. Page Time	0.03279486
LoadTest:Page	Avg. Page Time	0.03709091
LoadTest:Page	Avg. Page Time	0.2260593
LoadTest:Page	Avg. Page Time	0.04046456
LoadTest:Page	Avg. Page Time	0.04083582
LoadTest:Page	Avg. Page Time	0.03241572
LoadTest:Page	Avg. Page Time	0.04171153
LoadTest:Page	Avg. Page Time	0.03395294
LoadTest:Page	Avg. Page Time	0.03625925
LoadTest:Page	Avg. Page Time	0.03129411
LoadTest:Page	Avg. Page Time	0.03199999
LoadTest:Page	Avg. Page Time	0.03031815
LoadTest:Page	Avg. Page Time	0.03639996
LoadTest:Page	Avg. Page Time	0.03299998
LoadTest:Page	Avg. Page Time	0.02834996
LoadTest:Page	Avg. Page Time	0.03597296
LoadTest:Page	Avg. Page Time	0.03964863
LoadTest:Page	Avg. Page Time	0.03899998
LoadTest:Page	Avg. Page Time	0.03291378

DATOS DEL INDICADOR PÁGINA POR SEGUNDO PARA EL PATRÓN

MVVM

CategoryName	CounterName	ComputedValue
Memory	Pages/sec	0
Memory	Pages/sec	22.05616
Memory	Pages/sec	3.401795
Memory	Pages/sec	0.1999342
Memory	Pages/sec	0
Memory	Pages/sec	1.804744
Memory	Pages/sec	0
Memory	Pages/sec	0
Memory	Pages/sec	0
Memory	Pages/sec	0.4000391
Memory	Pages/sec	0
Memory	Pages/sec	0.4001894
Memory	Pages/sec	0.2000106
Memory	Pages/sec	0
Memory	Pages/sec	0.6001763
Memory	Pages/sec	0.6004742
Memory	Pages/sec	28.39883
Memory	Pages/sec	11.58605
Memory	Pages/sec	3.594313
Memory	Pages/sec	4.809079
Memory	Pages/sec	2.393656
Memory	Pages/sec	4.412142
Memory	Pages/sec	3.5947
Memory	Pages/sec	5.00672
Memory	Pages/sec	1.398319
Memory	Pages/sec	2.601931
Memory	Pages/sec	0.7993576
Memory	Pages/sec	0.3999474
Memory	Pages/sec	0.399618
Memory	Pages/sec	1.604452
Memory	Pages/sec	0.3995336
Memory	Pages/sec	0.2003276
Memory	Pages/sec	0.5990333
Memory	Pages/sec	0.3998575
Memory	Pages/sec	2.40065
Memory	Pages/sec	1.402594
Memory	Pages/sec	2.794889
Memory	Pages/sec	4.608142

DATOS DEL INDICADOR SOLICITUD/SEGUNDO PARA EL PATRÓN

MVVM

CategoryName	CounterName	ComputedValue
LoadTest:Request	Requests/Sec	0
LoadTest:Request	Requests/Sec	53.6
LoadTest:Request	Requests/Sec	23.4
LoadTest:Request	Requests/Sec	0
LoadTest:Request	Requests/Sec	0
LoadTest:Request	Requests/Sec	0
LoadTest:Request	Requests/Sec	0
LoadTest:Request	Requests/Sec	0
LoadTest:Request	Requests/Sec	0
LoadTest:Request	Requests/Sec	0
LoadTest:Request	Requests/Sec	0
LoadTest:Request	Requests/Sec	0
LoadTest:Request	Requests/Sec	1.2
LoadTest:Request	Requests/Sec	1.2
LoadTest:Request	Requests/Sec	1.4
LoadTest:Request	Requests/Sec	9
LoadTest:Request	Requests/Sec	7.8
LoadTest:Request	Requests/Sec	11.4
LoadTest:Request	Requests/Sec	15.6
LoadTest:Request	Requests/Sec	24.2
LoadTest:Request	Requests/Sec	23.6
LoadTest:Request	Requests/Sec	25.4
LoadTest:Request	Requests/Sec	26.8
LoadTest:Request	Requests/Sec	17.8
LoadTest:Request	Requests/Sec	20.8
LoadTest:Request	Requests/Sec	17
LoadTest:Request	Requests/Sec	10.8
LoadTest:Request	Requests/Sec	10.2
LoadTest:Request	Requests/Sec	4.8
LoadTest:Request	Requests/Sec	4.4
LoadTest:Request	Requests/Sec	3
LoadTest:Request	Requests/Sec	3.8
LoadTest:Request	Requests/Sec	4
LoadTest:Request	Requests/Sec	7.4
LoadTest:Request	Requests/Sec	7.4
LoadTest:Request	Requests/Sec	9
LoadTest:Request	Requests/Sec	11.6
LoadTest:Request	Requests/Sec	16.4
LoadTest:Request	Requests/Sec	14.2

DATOS DEL INDICADOR PORCENTAJE DE TIEMPO DE RESPUESTA

PARA EL PATRÓN MVVM

CategoryName	CounterName	ComputedValue
LoadTest:Request	Avg. Response Time	0
LoadTest:Request	Avg. Response Time	2.196254
LoadTest:Request	Avg. Response Time	4.476461
LoadTest:Request	Avg. Response Time	0
LoadTest:Request	Avg. Response Time	0
LoadTest:Request	Avg. Response Time	0
LoadTest:Request	Avg. Response Time	0
LoadTest:Request	Avg. Response Time	0
LoadTest:Request	Avg. Response Time	0
LoadTest:Request	Avg. Response Time	0
LoadTest:Request	Avg. Response Time	0
LoadTest:Request	Avg. Response Time	0.03466666
LoadTest:Request	Avg. Response Time	0.0328333
LoadTest:Request	Avg. Response Time	0.03728569
LoadTest:Request	Avg. Response Time	0.03864444
LoadTest:Request	Avg. Response Time	0.03579486
LoadTest:Request	Avg. Response Time	0.03438596
LoadTest:Request	Avg. Response Time	0.03279487
LoadTest:Request	Avg. Response Time	0.03709091
LoadTest:Request	Avg. Response Time	0.2260593
LoadTest:Request	Avg. Response Time	0.04046457
LoadTest:Request	Avg. Response Time	0.04083582
LoadTest:Request	Avg. Response Time	0.03241573
LoadTest:Request	Avg. Response Time	0.04171154
LoadTest:Request	Avg. Response Time	0.03395294
LoadTest:Request	Avg. Response Time	0.03625926
LoadTest:Request	Avg. Response Time	0.03129411
LoadTest:Request	Avg. Response Time	0.03199999
LoadTest:Request	Avg. Response Time	0.03031817
LoadTest:Request	Avg. Response Time	0.03639999
LoadTest:Request	Avg. Response Time	0.03299998
LoadTest:Request	Avg. Response Time	0.02835
LoadTest:Request	Avg. Response Time	0.03597297
LoadTest:Request	Avg. Response Time	0.03964864
LoadTest:Request	Avg. Response Time	0.03899999
LoadTest:Request	Avg. Response Time	0.03291379
LoadTest:Request	Avg. Response Time	0.03232927
LoadTest:Request	Avg. Response Time	0.03876056

**DATOS DEL INDICADOR PORCENTAJE DE TIEMPO DE PROCESADOR
PARA EL PATRÓN MVVM**

CategoryName	CounterName	ComputedValue
Processor	% Processor Time	0
Processor	% Processor Time	92.94657
Processor	% Processor Time	39.11525
Processor	% Processor Time	38.60575
Processor	% Processor Time	37.11297
Processor	% Processor Time	35.48687
Processor	% Processor Time	28.50739
Processor	% Processor Time	23.13524
Processor	% Processor Time	27.35423
Processor	% Processor Time	32.40247
Processor	% Processor Time	25.49124
Processor	% Processor Time	27.04969
Processor	% Processor Time	31.97931
Processor	% Processor Time	33.4943
Processor	% Processor Time	30.0993
Processor	% Processor Time	32.90001
Processor	% Processor Time	39.21868
Processor	% Processor Time	33.63333
Processor	% Processor Time	38.20867
Processor	% Processor Time	40.95536
Processor	% Processor Time	45.39852
Processor	% Processor Time	46.80208
Processor	% Processor Time	33.21887
Processor	% Processor Time	45.79414
Processor	% Processor Time	31.59389
Processor	% Processor Time	32.14254
Processor	% Processor Time	27.27239
Processor	% Processor Time	26.25334
Processor	% Processor Time	23.24123
Processor	% Processor Time	29.05372
Processor	% Processor Time	21.42821
Processor	% Processor Time	37.39101
Processor	% Processor Time	36.39426
Processor	% Processor Time	26.79094
Processor	% Processor Time	25.28701
Processor	% Processor Time	31.47509
Processor	% Processor Time	31.49319
Processor	% Processor Time	27.65924

**DATOS DEL INDICADOR PORCENTAJE DE TIEMPO DE PROCESADOR
PARA EL PATRÓN MVVM**

CategoryName	CounterName	ComputedValue
Memory	Available MBytes	1323
Memory	Available MBytes	1260
Memory	Available MBytes	1246
Memory	Available MBytes	1247
Memory	Available MBytes	1249
Memory	Available MBytes	1250
Memory	Available MBytes	1250
Memory	Available MBytes	1252
Memory	Available MBytes	1252
Memory	Available MBytes	1252
Memory	Available MBytes	1252
Memory	Available MBytes	1252
Memory	Available MBytes	1253
Memory	Available MBytes	1254
Memory	Available MBytes	1250
Memory	Available MBytes	1268
Memory	Available MBytes	2000
Memory	Available MBytes	2177
Memory	Available MBytes	2164
Memory	Available MBytes	2158
Memory	Available MBytes	2158
Memory	Available MBytes	2157
Memory	Available MBytes	2155
Memory	Available MBytes	2165
Memory	Available MBytes	2159
Memory	Available MBytes	2158
Memory	Available MBytes	2158
Memory	Available MBytes	2159
Memory	Available MBytes	2161
Memory	Available MBytes	2162
Memory	Available MBytes	2162
Memory	Available MBytes	2163
Memory	Available MBytes	2163
Memory	Available MBytes	2160
Memory	Available MBytes	2159
Memory	Available MBytes	2159
Memory	Available MBytes	2160
Memory	Available MBytes	2161

ANEXO IV PRUEBAS PRODUCTIVIDAD MVC

class CompraController

Class CompraController.

Namespace: SalesSupportSystem.Controllers

Most Complex: CabeceraCompra 8

Fields: 7

Properties: 2

Methods: 17

Regions: 2

ANEXO SPRINT

TABLA 0-II SPRINT 1

ITERACIÓN 1							
PROYECTO: SADIF							
ID	TIPO	RANGO	TITULO	PUNTOS	ASIGNADO	ESTADO	ESTIMACIÓN HORAS
1	Inicio	1	Inicio del Proyecto	30	Javier Vallejo	AC ¹ /C ²	160
	Tarea		Reunión con el coordinador del proyecto	1	Javier Vallejo	C	5
	Tarea		Diseño de la base de datos	1	Javier Vallejo	C	5
	Tarea		Pruebas a la base de datos	1	Javier Vallejo	C	5
	Tarea		Adaptación módulo de inventario	2	Javier Vallejo	C	5
	Tarea		Creación del plan de generación de datos	1	Javier vallejo	C	5
	Tarea		Creación del proyecto de infraestructura y	2	Javier Vallejo	C	40

¹ AC: ACTIVO

² C: COMPLETO

			acceso a datos				
	Tarea		Creación del proyecto de Infraestructura y lógica de negocios	2	Javier Vallejo	C	45
	Tarea		Creación del proyecto de infraestructura y Presentación	2	Javier Vallejo	C	50
2	Caso de Usuario	2	El sistema debe permitir el ingreso mediante autenticación de usuarios.	3	Javier Vallejo	AC/C	8
	Tarea		Generar el acceso a datos	1	Javier Vallejo	C	2
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	C	2
	Tarea		Generar la capa de presentación	1	Javier Vallejo	C	2
	Tarea		Pruebas Unitarias	1	Javier Vallejo	C	1
	Tarea		Generar la documentación	1	Javier Vallejo	C	1
3	Caso de Usuario	2	Como administrador o contador deseo ingresar los datos de los empleados.	4	Javier Vallejo	AC/C	8
	Tarea		Generar el acceso a datos	1	Javier Vallejo	C	2

	Tarea		Generar la lógica de negocios	1	Javier Vallejo	C	2
	Tarea		Generar la capa de presentación	2	Javier Vallejo	C	2
	Tarea		Pruebas Unitarias	1	Javier Vallejo	C	1
	Tarea		Generar la documentación	1	Javier Vallejo	C	1
4	Caso de Usuario	2	Como administrador o contador deseo modificar los datos de los empleados.	3	Javier Vallejo	AC/C	8
	Tarea		Generar el acceso a datos	1	Javier Vallejo	C	2
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	C	2
	Tarea		Generar la capa de presentación	2	Javier Vallejo	C	2
	Tarea		Pruebas Unitarias	1	Javier Vallejo	C	1
	Tarea		Generar la documentación	1	Javier Vallejo	C	1
5	Caso de Usuario	2	Como administrador, contador o cajero deseo ingresar los datos de los clientes.	4	Javier Vallejo	AC/C	8
	Tarea		Generar el acceso a datos	1	Javier Vallejo	C	2
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	C	2

	Tarea		Generar la capa de presentación	2	Javier Vallejo	C	2
	Tarea		Pruebas Unitarias	1	Javier Vallejo	C	1
	Tarea		Generar la documentación	1	Javier Vallejo	C	1
6	Caso de Usuario	2	Como administrador, contador o cajero deseo modificar los datos de los clientes.	3	Javier Vallejo	AC/C	8
	Tarea		Generar el acceso a datos	1	Javier Vallejo	C	2
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	C	2
	Tarea		Generar la capa de presentación	2	Javier Vallejo	C	2
	Tarea		Pruebas Unitarias	1	Javier Vallejo	C	1
	Tarea		Generar la documentación	1	Javier Vallejo	C	1
7	Caso de Usuario	2	Como administrador o contador deseo ingresar los datos de los proveedores.	4	Javier Vallejo	AC/C	8
	Tarea		Generar el acceso a datos	1	Javier Vallejo	C	2
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	C	2
	Tarea		Generar la capa de presentación	2	Javier Vallejo	C	2

	Tarea		Pruebas Unitarias	1	Javier Vallejo	C	1
	Tarea		Generar la documentación	1	Javier Vallejo	C	1
8	Caso de Usuario	2	Como administrador o contador deseo modificar los datos de los proveedores.	3	Javier Vallejo	AC/C	8
	Tarea		Generar el acceso a datos	1	Javier Vallejo	C	2
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	C	2
	Tarea		Generar la capa de presentación	2	Javier Vallejo	C	2
	Tarea		Pruebas Unitarias	1	Javier Vallejo	C	1
	Tarea		Generar la documentación	1	Javier Vallejo	C	1
9	Caso de Usuario	2	Como administrador o contador deseo ingresar los datos de los productos.	4	Javier Vallejo	AC/C	8
	Tarea		Generar el acceso a datos	1	Javier Vallejo	C	2
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	C	2
	Tarea		Generar la capa de presentación	2	Javier Vallejo	C	2
	Tarea		Pruebas Unitarias	1	Javier Vallejo	C	1

	Tarea		Generar la documentación	1	Javier Vallejo	C	1
10	Caso de Usuario	2	Como administrador o contador deseo modificar los datos de los productos.	3	Javier Vallejo	AC/C	8
	Tarea		Generar el acceso a datos	1	Javier Vallejo	C	2
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	C	2
	Tarea		Generar la capa de presentación	2	Javier Vallejo	C	2
	Tarea		Pruebas Unitarias	1	Javier Vallejo	C	1
	Tarea		Generar la documentación	1	Javier Vallejo	C	1

TABLA IV-III SPRINT 2

ITERACIÓN 2							
PROYECTO: SADIF							
ID	TIPO	RANGO	TITULO	PUNTOS	ASIGNADO	ESTADO	ESTIMACIÓN HORAS
1	Caso de usuario	3	Como administrador o contador deseo poder ingresar una factura de compras.	10	Javier Vallejo	AC³/C⁴	10
	Tarea		Generar el acceso a datos	1	Javier Vallejo	C	2
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	C	2
	Tarea		Generar la capa de presentación	2	Javier Vallejo	C	2
	Tarea		Pruebas Unitarias	1	Javier Vallejo	C	2
	Tarea		Generar la documentación	1	Javier Vallejo	C	2
2	Caso de usuario	3	Como administrador o cajero deseo poder ingresar una factura de venta.	10	Javier Vallejo	AC/C	10
	Tarea		Generar el acceso a datos	1	Javier Vallejo	C	2

³ AC: ACTIVO

⁴ C: COMPLETO

	Tarea		Generar la lógica de negocios	1	Javier Vallejo	C	2
	Tarea		Generar la capa de presentación	2	Javier Vallejo	C	2
	Tarea		Pruebas Unitarias	1	Javier Vallejo	C	2
	Tarea		Generar la documentación	1	Javier Vallejo	C	2
3	Caso de usuario	3	Como administrador deseo poder modificar la utilidad de los precios de cada producto identificándolos para la venta como unidad, docena, ciento, millar.	7	Javier Vallejo	AC/C	6
	Tarea		Generar el acceso a datos	1	Javier Vallejo	C	1
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	C	1
	Tarea		Generar la capa de presentación	2	Javier Vallejo	C	1
	Tarea		Pruebas Unitarias	1	Javier Vallejo	C	1
	Tarea		Generar la documentación	1	Javier Vallejo	C	2
4	Caso de usuario	3	Como administrador deseo poder realizar búsquedas de productos, empleados, clientes, proveedores.	5	Javier Vallejo	AC/C	7
	Tarea		Generar el acceso a datos	1	Javier Vallejo	AC/C	2

	Tarea		Generar la lógica de negocios	1	Javier Vallejo	AC/C	2
	Tarea		Generar la capa de presentación	2	Javier Vallejo	AC/C	1
	Tarea		Pruebas Unitarias	1	Javier Vallejo	AC/C	1
	Tarea		Generar la documentación	1	Javier Vallejo	AC/C	1
5	Caso de usuario	3	Como administrador deseo poder imprimir la venta realizada las veces que sean necesarios.	8	Javier Vallejo	AC/C	8
	Tarea		Generar el acceso a datos	1	Javier Vallejo	AC/C	2
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	AC/C	2
	Tarea		Generar la capa de presentación	2	Javier Vallejo	AC/C	2
	Tarea		Pruebas Unitarias	1	Javier Vallejo	AC/C	1
	Tarea		Generar la documentación	1	Javier Vallejo	AC/C	1

TABLA IV-IIIIV SPRINT 3

ITERACIÓN 3							
PROYECTO: SADIF							
ID	TIPO	RANGO	TITULO	PUNTOS	ASIGNADO	ESTADO	ESTIMACIÓN HORAS
1	Caso de usuario	4	Como administrador deseo poder anular Facturas de ventas.	4	Javier Vallejo	AC⁵/C⁶	5
	Tarea		Generar el acceso a datos	1	Javier Vallejo	AC	1
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	AC	1
	Tarea		Generar la capa de presentación	2	Javier Vallejo	AC	1
	Tarea		Pruebas Unitarias	1	Javier Vallejo	AC	1
	Tarea		Generar la documentación	1	Javier Vallejo	AC	1
2	Caso de usuario	4	Como administrador deseo poder cambiar la clave de ingreso al sistema de los empleados.	2	Javier Vallejo	AC/C	5
	Tarea		Generar el acceso a datos	1	Javier Vallejo	AC	1

⁵ AC: ACTIVO

⁶ C: COMPLETO

	Tarea		Generar la lógica de negocios	1	Javier Vallejo	AC	1
	Tarea		Generar la capa de presentación	2	Javier Vallejo	AC	1
	Tarea		Pruebas Unitarias	1	Javier Vallejo	AC	1
	Tarea		Generar la documentación	1	Javier Vallejo	AC	1
3	Caso de usuario	4	Como administrador deseo visualizar un reporte de los Kardex de todos los productos	7	Javier Vallejo	AC/C	6
	Tarea		Generar el acceso a datos	1	Javier Vallejo	AC	1
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	AC	1
	Tarea		Generar la capa de presentación	2	Javier Vallejo	AC	1
	Tarea		Pruebas Unitarias	1	Javier Vallejo	AC	1
	Tarea		Generar la documentación	1	Javier Vallejo	AC	2
4	Caso de usuario	4	Como administrador deseo visualizar un reporte de los Kardex por producto requerido.	6	Javier Vallejo	AC/C	5
	Tarea		Generar el acceso a datos	1	Javier Vallejo	AC	1
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	AC	1
	Tarea		Generar la capa de presentación	2	Javier Vallejo	AC	1

	Tarea		Pruebas Unitarias	1	Javier Vallejo	AC	1
	Tarea		Generar la documentación	1	Javier Vallejo	AC	1
5	Caso de usuario	4	Como administrador deseo visualizar un reporte de las ventas realizadas por fechas.	6	Javier Vallejo	AC/C	5
	Tarea		Generar el acceso a datos	1	Javier Vallejo	AC	1
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	AC	1
	Tarea		Generar la capa de presentación	2	Javier Vallejo	AC	1
	Tarea		Pruebas Unitarias	1	Javier Vallejo	AC	1
	Tarea		Generar la documentación	1	Javier Vallejo	AC	1
6	Caso de usuario	4	Como administrador deseo visualizar un reporte de compras realizadas por fechas.	9	Javier Vallejo	AC/C	8
	Tarea		Generar el acceso a datos	1	Javier Vallejo	AC	2
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	AC	2
	Tarea		Generar la capa de presentación	2	Javier Vallejo	AC	2
	Tarea		Pruebas Unitarias	1	Javier Vallejo	AC	1
	Tarea		Generar la documentación	1	Javier Vallejo	AC	1

TABLA 0III-IV SPRINT 4

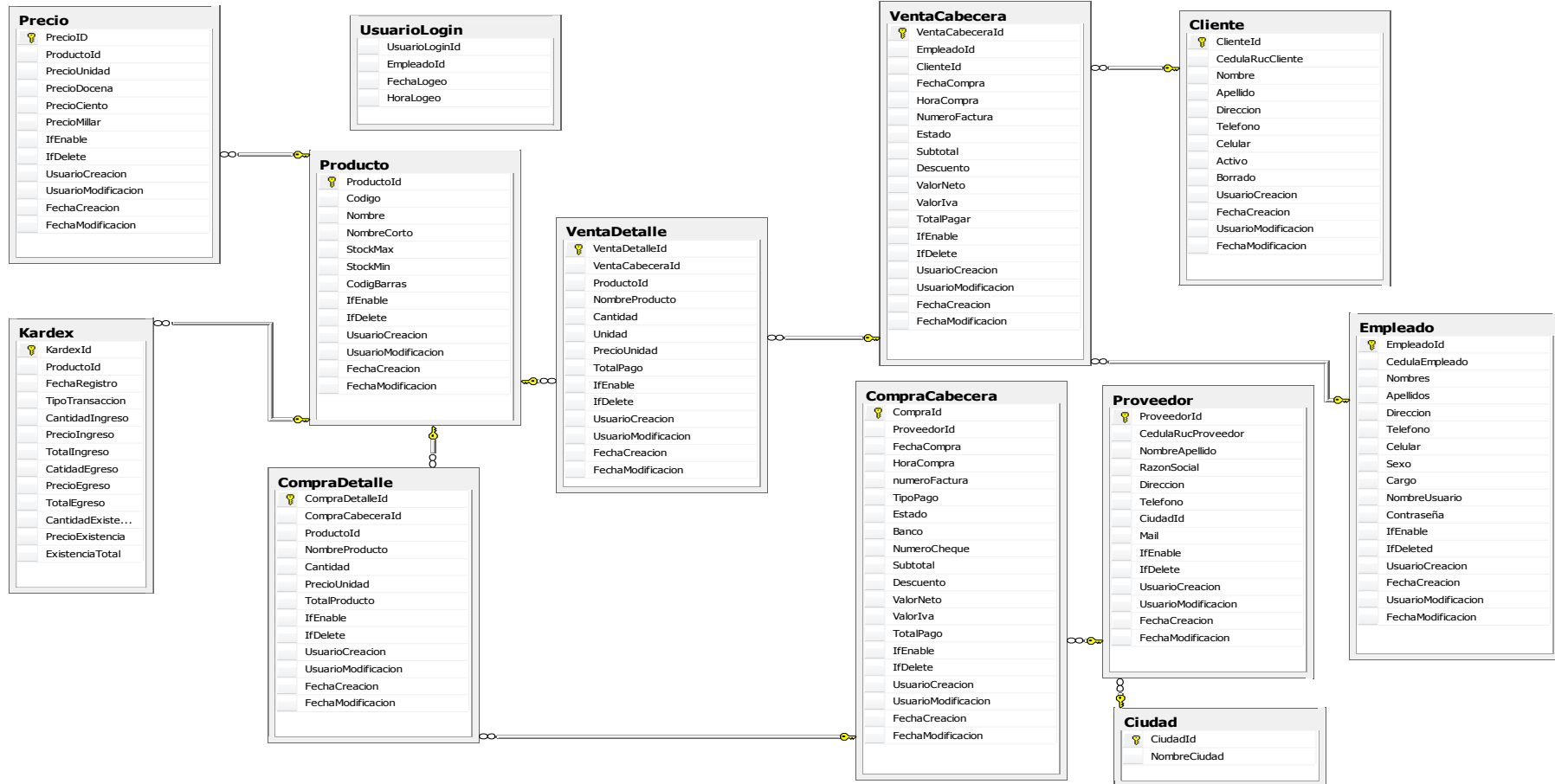
ITERACIÓN 4							
PROYECTO: SADIF							
ID	TIPO	RANGO	TITULO	PUNTOS	ASIGNADO	ESTADO	ESTIMACIÓN HORAS
1	Caso de usuario	5	Como administrador deseo visualizar un reporte de utilidad de los productos vendidos.	9	Javier Vallejo	AC⁷/C⁸	5
	Tarea		Generar el acceso a datos	1	Javier Vallejo	AC	1
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	AC	1
	Tarea		Generar la capa de presentación	2	Javier Vallejo	AC	1
	Tarea		Pruebas Unitarias	1	Javier Vallejo	AC	1
	Tarea		Generar la documentación	1	Javier Vallejo	AC	1
2	Caso de usuario	5	Como administrador deseo visualizar un reporte de inventario por fecha seleccionada	9	Javier Vallejo	AC/C	5

⁷ AC: ACTIVO

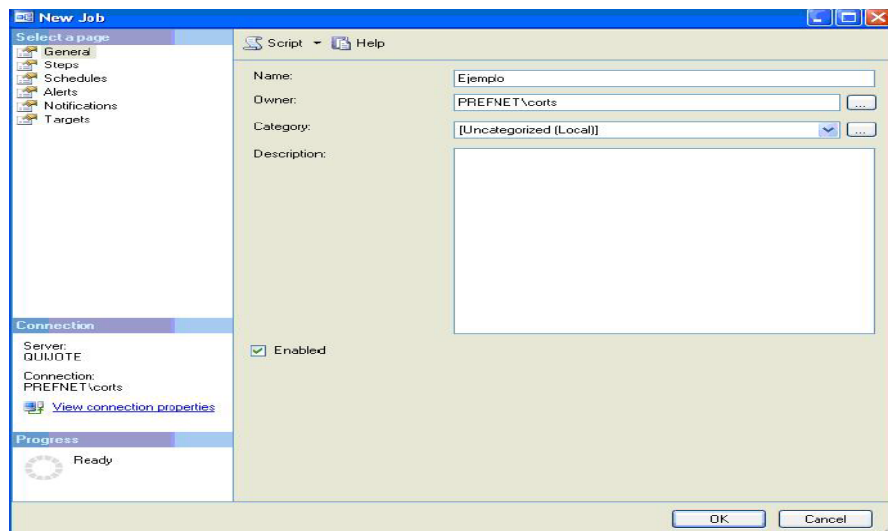
⁸ C: COMPLETO

	Tarea		Generar el acceso a datos	1	Javier Vallejo	AC	1
	Tarea		Generar la lógica de negocios	1	Javier Vallejo	AC	1
	Tarea		Generar la capa de presentación	2	Javier Vallejo	AC	1
	Tarea		Pruebas Unitarias	1	Javier Vallejo	AC	1
	Tarea		Generar la documentación	1	Javier Vallejo	AC	1

BASE DATOS TESTFACIN



PESTAÑA VENTANA DEL SQL AGENT SERVER



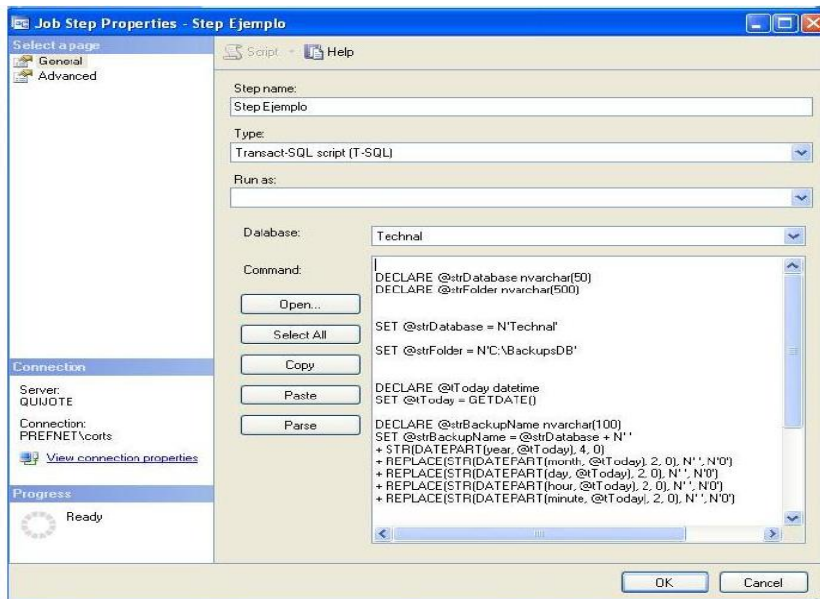
SCRIPT RESPALDO DE BASE DE DATOS

```
DECLARE @BD VARCHAR(100)
DECLARE @CARPETA VARCHAR (500)
DECLARE @HOY DATETIME
DECLARE @STRBK VARCHAR(500)
DECLARE @STRARCHIVO VARCHAR(600)

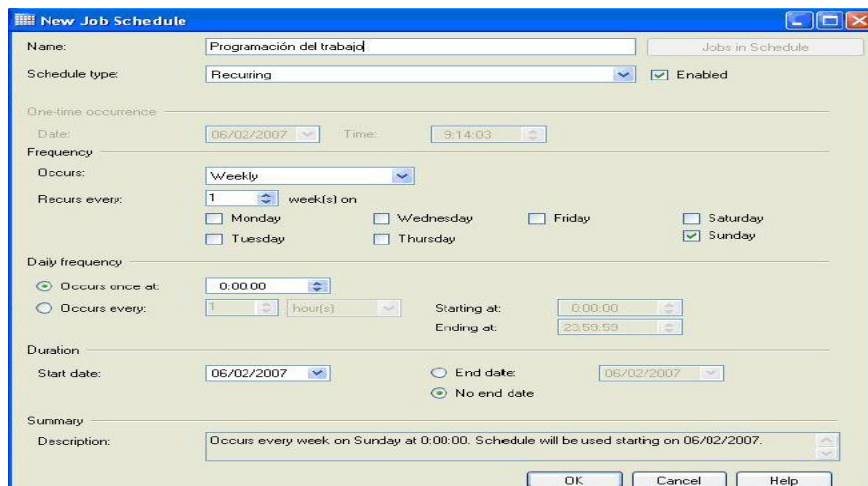
SET @BD= N'BD_INVENTARIO'
SET @CARPETA= N'C:\BACKUP'
SET @HOY=GETDATE()
SET @STRBK= @BD+ N' '+STR(DATEPART(YEAR,@HOY),4,0)
              +REPLACE(STR(DATEPART(MONTH,@HOY),2,0),N' ',N'0')
              +REPLACE(STR(DATEPART(DAY,@HOY),2,0),N' ',N'0')
              +REPLACE(STR(DATEPART(HOUR,@HOY),2,0),N' ',N'0')
              +REPLACE(STR(DATEPART(MINUTE,@HOY),2,0),N' ',N'0')

SET @STRARCHIVO=@CARPETA +N'\'+@STRBK+ N'.BAK'
BACKUP DATABASE @BD TO DISK= @STRARCHIVO
WITH NOFORMAT,INIT, SKIP,NAME=@STRBK
```

PESTAÑA VENTANA DEL PASO SQL SERVER AGENT



HORARIO DEL JOB



ANEXOS CASO DE USOS

CASO DE USO DE AUTENTICACIÓN DE USUARIOS

Identificador Caso de Uso	CU_ AUTENTICAR
Nombre del caso de Uso	Autenticar los usuarios.
Actores	Usuarios
Propósito	Verificar si un usuario está registrado en el sistema.
Visión General	El Sistema verifica si un usuario está registrado y le permite o niega el acceso a la aplicación
Tipo	Primario y Esencial
Referencia	Requerimiento 1
CURSO TÍPICO DE REFERENCIAS	
ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
5. El usuario utiliza la aplicación.	
CURSOS ALTERNOS	
ERROR: Cedula y/o Password son incorrectas, se indica el error y se visualiza la página de autenticación. Si ingresa su contraseña tres veces incorrectamente	

CASO DE USO INGRESO EMPLEADO

Identificador Caso de Uso	CU_ INGRESO EMPLEADO
Nombre del caso de Uso	Ingreso de empleados
Actores	Usuario: Administrador y Contador
Propósito	Ingresar los datos de los empleados.
Visión General	El sistemas permite ingresar los datos de un empleado validando la cedula y campos vacíos.
Tipo	Primario y Esencial
Referencia	Requerimiento 2
CURSO TÍPICO DE REFERENCIAS	
ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
5. El usuario da clic en el botón de empleados.	6. El sistema le muestra la venta de ingreso de datos del empleado.
7. El usuario ingresa los datos del empleado y da clic en el botón almacenar.	8. El sistema verifica si la cedula esta correcta. 9. Si la cedula esta correcta verifica los campos. 10. Si la cedula es incorrecta muestra un mensaje de error. 11. Si los campos son obligatorios: 12. Se emite un mensaje de advertencia: “Los campos son obligatorios”. 13. Si la cedula y los campos están correctos. 14. El sistema emite un mensaje: “Se ingresó correctamente”.
CURSOS ALTERNOS	
ERROR: Cedula es incorrecta. Campos obligatorios.	

CASO DE USO MODIFICAR EMPLEADO

Identificador Caso de Uso	CU_ MODIFICAR EMPLEADO
Nombre del caso de Uso	Modificar de empleados
Actores	Usuario: Administrador y Contador
Propósito	Modificar los datos de los empleados.
Visión General	El sistemas permite modificar los datos de un empleado validando la cedula y campos vacíos.
Tipo	Primario y Esencial
Referencia	Requerimiento 3
CURSO TÍPICO DE REFERENCIAS	
ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
5. El usuario da clic en el botón de buscar y selecciona la pestaña de empleados ingresa la cedula.	6. El sistema le muestra una tabla con los datos del empleado.
7. El usuario selecciona la fila en la que se encuentre los datos del empleado da clic izquierdo y escoge modificar.	8. El sistema muestra la venta de modificación de empleados.
9. El usuario ingresa los datos que desee modificar y da clic en el botón almacenar.	10. El sistema verifica si la cedula esta correcta. 11. Si la cedula esta correcta verifica los campos. 12. Si la cedula es incorrecta muestra un mensaje de error. 13. Si los campos son obligatorios: 14. Se emite un mensaje de advertencia: "Los campos son obligatorios". 15. Si la cedula y los campos están correctos. 16. El sistema emite un mensaje: 17. "Se ingresó correctamente"
CURSOS ALTERNOS	

ERROR: Cedula es incorrecta.
Campos obligatorios.

CASO DE USO INGRESO CLIENTES

Identificador Caso de Uso	CU_ INGRESO CLIENTES
Nombre del caso de Uso	Ingreso de clientes
Actores	Usuario: Administrador, Cajero y Contador
Propósito	Ingresar los datos de los clientes.
Visión General	El sistemas permite ingresar los datos de un cliente validando la cedula y campos vacíos.
Tipo	Primario y Esencial
Referencia	Requerimiento 4

CURSO TÍPICO DE REFERENCIAS

ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador cajero o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
5. El usuario da clic en el botón de clientes.	6. El sistema le muestra la venta de ingreso de datos del cliente.
7. El usuario ingresa los datos del cliente y da clic en el botón almacenar.	8. El sistema verifica si la cedula esta correcta. 9. Si la cedula esta correcta verifica los campos. 10. Si la cedula es incorrecta muestra un mensaje de error. 11. Si los campos son obligatorios: 12. Se emite un mensaje de advertencia: "Los campos son obligatorios". 13. Si la cedula y los campos están correctos. 14. El sistema emite un mensaje: "Se ingresó correctamente".

CURSOS ALTERNOS

ERROR: Cedula es incorrecta.
Campos obligatorios.

CASO DE USO MODIFICAR CLIENTE

Identificador Caso de Uso	CU_ MODIFICAR CLIENTE
Nombre del caso de Uso	Modificar Cliente
Actores	Usuario: Administrador Cajero y Contador
Propósito	Modificar los datos de los clientes.
Visión General	El sistemas permite modificar los datos de un cliente validando la cedula y campos vacíos.
Tipo	Primario y Esencial
Referencia	Requerimiento 5
CURSO TÍPICO DE REFERENCIAS	
ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador cajero o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
5. El usuario da clic en el botón de buscar y selecciona la pestaña de clientes e ingresa la cedula.	6. El sistema le muestra una tabla con los datos del cliente.
7. El usuario selecciona la fila en la que se encuentre los datos del cliente da clic izquierdo y escoge modificar.	8. El sistema muestra la venta de modificación de cliente.
9. El usuario ingresa los datos que desee modificar y da clic en el botón almacenar.	10. El sistema verifica si la cedula esta correcta. 11. Si la cedula esta correcta verifica los campos. 12. Si la cedula es incorrecta muestra un mensaje de error. 13. Si los campos son obligatorios: 14. Se emite un mensaje de advertencia: "Los campos son obligatorios". 15. Si la cedula y los campos están correctos.

	16. El sistema emite un mensaje: 17. “Se ingresó correctamente”
CURSOS ALTERNOS	
ERROR: Cedula es incorrecta. Campos obligatorios.	

CASO DE USO INGRESO PROVEEDORES

Identificador Caso de Uso	CU_ INGRESO PROVEEDOR
Nombre del caso de Uso	Ingreso de PROVEEDORES
Actores	Usuario: Administrador o Contador
Propósito	Ingresar los datos de los proveedores.
Visión General	El sistemas permite ingresar los datos de un proveedor validando la cedula y campos vacíos.
Tipo	Primario y Esencial
Referencia	Requerimiento 6

CURSO TÍPICO DE REFERENCIAS

ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador cajero o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
5. El usuario da clic en el botón de proveedores.	6. El sistema le muestra la venta de ingreso de datos del proveedor.
7. El usuario ingresa los datos del proveedor y da clic en el botón almacenar.	8. El sistema verifica si la cedula esta correcta. 9. Si la cedula esta correcta verifica los campos. 10. Si la cedula es incorrecta muestra un mensaje de error. 11. Si los campos son obligatorios: 12. Se emite un mensaje de advertencia: “Los campos son obligatorios”. 13. Si la cedula y los campos están correctos.

	14. El sistema emite un mensaje: “Se ingresó correctamente”.
CURSOS ALTERNOS	
ERROR: Cedula es incorrecta. Campos obligatorios.	

CASO DE USO MODIFICAR PROVEEDOR

Identificador Caso de Uso	CU_ MODIFICAR PROVEEDOR
Nombre del caso de Uso	Modificar proveedor
Actores	Usuario: Administrador o Contador
Propósito	Modificar los datos de los proveedores.
Visión General	El sistemas permite modificar los datos de un proveedor validando la cedula y campos vacíos.
Tipo	Primario y Esencial
Referencia	Requerimiento 7

CURSO TÍPICO DE REFERENCIAS

ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador cajero o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
5. El usuario da clic en el botón de buscar y selecciona la pestaña de proveedor e ingresa la cedula.	6. El sistema le muestra una tabla con los datos del proveedor.
7. El usuario selecciona la fila en la que se encuentre los datos del proveedor da clic izquierdo y escoge modificar.	8. El sistema muestra la venta de modificación de proveedor.
9. El usuario ingresa los datos que desee modificar y da clic en el botón almacenar.	10. El sistema verifica si la cedula esta correcta. 11. Si la cedula esta correcta verifica los campos. 12. Si la cedula es incorrecta muestra un mensaje de error. 13. Si los campos son obligatorios:

	<p>14. Se emite un mensaje de advertencia: “Los campos son obligatorios”.</p> <p>15. Si la cedula y los campos están correctos.</p> <p>16. El sistema emite un mensaje:</p> <p>17. “Se ingresó correctamente”</p>
CURSOS ALTERNOS	
ERROR: Cedula es incorrecta. Campos obligatorios.	

CASO DE USO INGRESO PRODUCTO

Identificador Caso de Uso	CU_ INGRESO PRODUCTO
Nombre del caso de Uso	Ingreso de productos
Actores	Usuario: Administrador o Contador
Propósito	Ingresar los datos de los productos.
Visión General	El sistema permite ingresar los datos de un producto validando los campos vacíos.
Tipo	Primario y Esencial
Referencia	Requerimiento 8

CURSO TÍPICO DE REFERENCIAS

ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador cajero o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
5. El usuario da clic en el botón de productos.	6. El sistema le muestra la venta de ingreso de datos del producto.
7. El usuario ingresa los datos del producto y da clic en el botón almacenar.	<p>8. Si los campos son obligatorios:</p> <p>9. Se emite un mensaje de advertencia: “Los campos son obligatorios”.</p> <p>10. Si los campos están correctos.</p> <p>11. El sistema emite un mensaje: “Se ingresó correctamente”.</p>

CURSOS ALTERNOS

ERROR:
Campos obligatorios.

CASO DE USO MODIFICAR PRODUCTOS

Identificador Caso de Uso	CU_ MODIFICAR PRODUCTOS
Nombre del caso de Uso	Modificar productos
Actores	Usuario: Administrador o Contador
Propósito	Modificar los datos de los productos.
Visión General	El sistema permite modificar los datos de un producto validando los campos vacíos.
Tipo	Primario y Esencial
Referencia	Requerimiento 9

CURSO TÍPICO DE REFERENCIAS

ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador cajero o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
5. El usuario da clic en el botón de buscar y selecciona la pestaña de productos e ingresa el código.	6. El sistema le muestra una tabla con los datos del producto.
7. El usuario selecciona la fila en la que se encuentre los datos del producto da clic izquierdo y escoge modificar.	8. El sistema muestra la venta de modificación de producto.
9. El usuario ingresa los datos que desee modificar y da clic en el botón almacenar.	10. Verifica los campos. 11. Si los campos son obligatorios: 12. Se emite un mensaje de advertencia: "Los campos son obligatorios". 13. Si los campos están correctos. 14. El sistema emite un mensaje: 15. "Se ingresó correctamente"

CURSOS ALTERNOS
ERROR: Campos obligatorios.

CASO DE USO COMPRA PRODUCTO

Identificador Caso de Uso	CR ADQUISICIÓN DE PRODUCTOS	
Nombre del caso de Uso	Adquisición de productos	
Actores	Administrador, Proveedor	
Propósito	Realizar la compra de productos que necesita la empresa.	
Visión General	El proveedor entregara al gerente el pedido y la factura emitida.	
Tipo	Primario y esencial	
Referencia	Requerimiento 10	
CURSO TÍPICO DE REFERENCIAS		
ACCIÓN DEL ACTOR(ADMINISTRADOR)	ACCIÓN DEL ACTOR(PROVEEDOR)	REPUESTA DE SISTEMA
1. El usuario administrador cajero o contador requiere acceder a la aplicación		2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password		4. Muestra la ventana principal de la aplicación
5. El gerente realiza un pedido al proveedor	6. El proveedor se encarga de entregar los productos y la factura	
7. Si el proveedor no está registrado el administrador se encarga de ingresar los datos.	8. El proveedor entrega sus datos.	9. El sistema verifica si la cedula esta correcta.

		<p>10. Si la cedula esta correcta verifica los campos.</p> <p>11. Si la cedula es incorrecta muestra un mensaje de error.</p> <p>12. Si los campos son obligatorios:</p> <p>13. Se emite un mensaje de advertencia: "Los campos son obligatorios".</p> <p>14. Si la cedula y los campos están correctos.</p> <p>15. El sistema emite un mensaje:</p> <p>16. "Se ingresó correctamente".</p>
17. El usuario da clic en el botón de Factura de compra		18. El sistema le muestra la ventana para el ingreso de la factura
19. El usuario ingresa el número de factura.		20. El sistema indica si la factura ya ha sido ingresado. Emite un mensaje de advertencia "La factura ya existe "
21. El usuario ingresa la cedula del proveedor		22. El sistema muestra el nombre del Proveedor automáticamente.
23. El usuario ingresa el estado de la factura y el tipo de pago		24. Si el tipo de pago es en cheque o transferencia el sistema habilita el campo de número de cheque o transferencia y el banco de donde se realizó.
25. El usuario ingresa el código de producto		<p>26. Si el producto existe muestra el nombre del producto.</p> <p>27. Si no existe emite un mensaje de advertencia "El producto no existe".</p>
28. Si el producto no existe el usuario se encarga de ingresar los datos del		<p>29. Si los campos son obligatorios:</p> <p>30. Se emite un mensaje de advertencia:</p>

producto.		<p>“Los campos son obligatorios”.</p> <p>31. Si los campos están correctos.</p> <p>32. El sistema emite un mensaje:</p> <p>33. “Se ingresó correctamente”.</p>
34. El usuario da clic en el botón de almacenar		<p>35. Si los campos son obligatorios:</p> <p>36. Se emite un mensaje de advertencia:</p> <p>“Los campos son obligatorios”.</p> <p>37. Si los campos están correctos.</p> <p>38. El sistema emite un mensaje:</p> <p>39. “Se ingresó correctamente”.</p>
CURSOS ALTERNOS		
<p>ERROR: Campos obligatorios.</p>		

CASO DE USO VENTAS

Identificador Caso de Uso	CR VENTA DE PRODUCTOS	
Nombre del caso de Uso	Venta de productos	
Actores	Administrador, Cajero, contador o Cliente	
Propósito	Realizar la venta de productos.	
Visión General	El cliente entregara al cajero el pedido.	
Tipo	Primario y esencial	
Referencia	Requerimiento 11	
CURSO TÍPICO DE REFERENCIAS		
ACCIÓN DEL ACTOR(CAJERO)	ACCIÓN DEL ACTOR(CLIENTE)	REPUESTA DE SISTEMA
1. El usuario administrador cajero o contador requiere acceder a la aplicación		2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password		4. Muestra la ventana principal de la aplicación
	5. El cliente se encarga de entregar el pedido	
6. Si el cliente no está registrado el cajero se encarga de ingresar los datos.	7. El cliente entrega sus datos.	8. El sistema verifica si la cedula esta correcta. 9. Si la cedula esta correcta verifica los campos.

		<p>10. Si la cedula es incorrecta muestra un mensaje de error.</p> <p>11. Si los campos son obligatorios:</p> <p>12. Se emite un mensaje de advertencia: “Los campos son obligatorios”.</p> <p>13. Si la cedula y los campos están correctos.</p> <p>14. El sistema emite un mensaje:</p> <p>15. “Se ingresó correctamente”.</p>
16. El usuario da clic en el botón de Ventas	17. El cliente le entrega la lista de productos que necesita	18. El sistema le muestra la ventana para el ingreso de la factura
19. El usuario ingresa el número de factura.		20. El sistema indica si la factura ya ha sido ingresado. Emite un mensaje de advertencia “La factura ya existe ”
21. El usuario da clic en el botón buscar cliente		22. El sistema muestra la ventana de búsqueda de cliente.
23. El usuario selecciona el método por el que desee buscar nombre, apellido o cedula		24. El sistema muestra una tabla con los datos del cliente.
25. El usuario selecciona la fila clic izquierdo agregar cliente		26. El sistema muestra la venta de Ventas con los datos del cliente.
27. El usuario ingresa el estado de la factura y el empleado que realizo la venta		
28. El usuario ingresa el código de producto		<p>29. Si el producto existe muestra el nombre del producto.</p> <p>30. Si no existe emite un mensaje de advertencia “El producto no existe”.</p>
31. Si el producto no existe el usuario se		32. Si los campos son obligatorios:

encarga de ingresar los datos del producto.		33. Se emite un mensaje de advertencia: “Los campos son obligatorios”. 34. Si los campos están correctos. 35. El sistema emite un mensaje: 36. “Se ingresó correctamente”.
37. El usuario da clic en el botón de almacenar		38. Si los campos son obligatorios: 39. Se emite un mensaje de advertencia: “Los campos son obligatorios”. 40. Si los campos están correctos. 41. El sistema emite un mensaje: 42. “Se ingresó correctamente”.
43. El usuario pide el pago correspondiente.	44. El cliente cancela la factura	
45. El cajero recauda el valor y entrega la factura		46. El sistema emite la factura correspondiente. 47. El sistema disminuye la cantidad de productos
CURSOS ALTERNOS		
ERROR: Campos obligatorios.		

CASO DE USO MODIFICAR UTILIDAD PRECIOS

Identificador Caso de Uso	CU_ MODIFICAR UTILIDAD PRECIOS
Nombre del caso de Uso	Modificar utilidad precios
Actores	Usuario: Administrador o Contador
Propósito	Modificar la utilidad de precios de cada producto.
Visión General	El sistema permite modificar la utilidad de los precios de cada producto validando que los datos sean numéricos
Tipo	Primario y Esencial
Referencia	Requerimiento 12

CURSO TÍPICO DE REFERENCIAS

ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
5. El usuario da clic en el botón del menú ver escogemos la opción Precios	6. El sistema le muestra la ventana para ingresar los precios.
7. El usuario ingresa el código del producto que desee.	8. El sistema muestra el nombre del proveedor, el precio promedio y el precio de compra.
9. El usuario ingresa el porcentaje de utilidad dependiendo de la unidad sea esta Unidad, docena, Ciento y Millar	10. El sistema verifica si la utilidad es un valor numérico y no se encuentra vacío. 11. Si el sistema verifica que todo está correcto muestra el precio de venta de cada unidad dependiendo de la utilidad. 12. Si el sistema verifica que algún campo está vacío muestra un mensaje de error “No puede dejar campos vacío”

CURSOS ALTERNOS

ERROR: Cedula es incorrecta.
Campos obligatorios.

CASO DE USO BUSCAR

Identificador Caso de Uso	CU_ BÚSQUEDAS
Nombre del caso de Uso	Buscar un objeto deseado por el usuario
Actores	Usuario: Administrador Contador o Cajero
Propósito	Buscar un proveedor, producto, cliente o empleado
Visión General	El sistema buscar proveedores, clientes, empleados o productos mostrando nos sus datos
Tipo	Primario y Esencial
Referencia	Requerimiento 13
CURSO TÍPICO DE REFERENCIAS	
ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
5. El usuario da clic en el botón Buscar	6. El sistema le muestra la ventana para buscar siendo la primera opción la de cliente.
7. El usuario selecciona el objeto que desee buscar sean estos Cliente, Empleado, Proveedor, Producto; además el filtro por el que se dese buscar estos son Nombre, Apellido, Código o Cedula.	8. El sistema muestra una tabla con todos los datos del objeto seleccionado y por el filtro buscado
CURSOS ALTERNOS	
ERROR:	

CASO DE USO IMPRIMIR FACTURA

Identificador Caso de Uso	CU_ IMPRIMIR FACTURA
Nombre del caso de Uso	Imprimir factura
Actores	Usuario: Administrador Contador o Cajero
Propósito	Imprimir factura dado su número
Visión General	El sistema permite imprimir la factura las veces que se desee
Tipo	Primario y Esencial
Referencia	Requerimiento 14
CURSO TÍPICO DE REFERENCIAS	
ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
El usuario da clic en el botón “VER” del menú escogemos la opción Imprimir factura.	5. El sistema le muestra la ventana para ingresar el número de factura.
6. El usuario ingresa el número de factura que desee imprimir y da clic en el botón aceptar	7. El sistema muestra la factura que se buscó y manda a imprimir a la impresora por defecto.
CURSOS ALTERNOS	
ERROR: Validación del campo número.	

CASO DE USO ANULAR FACTURA

Identificador Caso de Uso	CU_ ANULAR FACTURA
Nombre del caso de Uso	Imprimir factura
Actores	Usuario: Administrador
Propósito	Anular factura dado su número
Visión General	El sistema permite anular una factura
Tipo	Primario y Esencial
Referencia	Requerimiento 15
CURSO TÍPICO DE REFERENCIAS	
ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
El usuario da clic en el botón “VER” del menú escogemos la opción Anular factura.	5. El sistema le muestra la ventana para ingresar el número de factura.
6. El usuario ingresa el número de factura que desee anular y da clic en el botón aceptar	7. El sistema anula la factura solo si el usuario es administrador si no lo es emite un mensaje de error: “ No tiene privilegios para anular” 8. El sistema emite valida si la factura ya fue anulada si ya fue anulada emite un mensaje de advertencia. “La factura ya fue anulada”. 9. Si la factura no fue anulada el sistema se encara de aumentar la cantidad de productos vendidos en la existencia de productos y emite un mensaje de información “La factura fue anulada correctamente”
CURSOS ALTERNOS	
ERROR: Validación del campo número.	

CASO DE USO CAMBIO DE CLAVE

Identificador Caso de Uso	CU_ CAMBIO CLAVE
Nombre del caso de Uso	Cambio de clave
Actores	Usuario: Administrador
Propósito	Cambiará la clave de ingreso
Visión General	El sistema permite cambiar la clave de los empleados para poder ingresar al sistema SADIF
Tipo	Primario y Esencial
Referencia	Requerimiento 16
CURSO TÍPICO DE REFERENCIAS	
ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
El usuario da clic en el botón "ARCHIVO" del menú escogemos la opción Cambio Clave.	5. El sistema le muestra la ventana para el cambio de clave.
6. El usuario ingresa el número de cedula y la nueva clave da clic en el botón aceptar.	7. El sistema emite un mensaje de información "cambio de clave correcto".
CURSOS ALTERNOS	
ERROR: Validación del campo cedula.	

CASO DE USO REPORTE KARDEX

Identificador Caso de Uso	CU_ REPORTE KARDEX
Nombre del caso de Uso	Reporte Kardex
Actores	Usuario: Administrador
Propósito	Ver el reporte de Kardex
Visión General	El sistema permite ver el reporte de todos los movimientos hechos por la empresa mediante su Kardex
Tipo	Primario y Esencial
Referencia	Requerimiento 17
CURSO TÍPICO DE REFERENCIAS	
ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
El usuario da clic en el botón "VER" del menú escogemos la opción Reportes y damos clic en el reporte Kardex.	5. El sistema le muestra la ventana para Visualizar el informe.
6. El usuario puede seleccionar imprimir o almacenar en diferentes formatos.	
CURSOS ALTERNOS	
ERROR:	

CASO DE USO REPORTE DE KARDEX POR PRODUCTO REQUERIDO

Identificador Caso de Uso	CU_ REPORTE KARDEX1
Nombre del caso de Uso	Reporte Kardex por producto requerido
Actores	Usuario: Administrador
Propósito	Ver el reporte de Kardex por producto requerido.
Visión General	El sistema permite ver el reporte de todos los movimientos hechos por la empresa mediante su Kardex dado un producto
Tipo	Primario y Esencial
Referencia	Requerimiento 18
CURSO TÍPICO DE REFERENCIAS	
ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
5. El usuario da clic en el botón “VER” del menú escogemos la opción Reportes y damos clic en el reporte Kardex por producto.	6. El sistema le muestra la ventana para ingresar el código del producto que se desee visualizar.
7. El ingresa el código del producto.	8. El sistema le muestra la ventana para Visualizar el informe
CURSOS ALTERNOS	
ERROR:	

CASO DE USO REPORTE VENTAS

Identificador Caso de Uso	CU_ REPORTE VENTAS
Nombre del caso de Uso	Reporte Ventas por fechas
Actores	Usuario: Administrador
Propósito	Ver el reporte de ventas por fecha
Visión General	El sistema permite ver el reporte de todas las ventas realizadas en fechas indicadas.
Tipo	Primario y Esencial
Referencia	Requerimiento 19
CURSO TÍPICO DE REFERENCIAS	
ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
El usuario da clic en el botón "VER" del menú escogemos la opción Reportes y damos clic en el reporte Ventas fecha.	5. El sistema le muestra la ventana para seleccionar desde que fecha y hasta que fecha se muestra las ventas.
6. El usuario selecciona las fechas y da clic en el botón de aceptar.	7. El sistema le muestra la ventana para Visualizar el informe
CURSOS ALTERNOS	
ERROR:	

CASO DE USO REPORTE COMPRAS

Identificador Caso de Uso	CU_ REPORTE COMPRAS
Nombre del caso de Uso	Reporte Compras por fechas
Actores	Usuario: Administrador
Propósito	Ver el reporte de compras por fecha
Visión General	El sistema permite ver el reporte de todas las compras realizadas en fechas indicadas.
Tipo	Primario y Esencial
Referencia	Requerimiento 20
CURSO TÍPICO DE REFERENCIAS	
ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
El usuario da clic en el botón “VER” del menú escogemos la opción Reportes y damos clic en el reporte Compras.	5. El sistema le muestra la ventana para seleccionar desde que fecha y hasta que fecha se muestra las compras.
6. El usuario selecciona las fechas y da clic en el botón de aceptar.	7. El sistema le muestra la ventana para Visualizar el informe
CURSOS ALTERNOS	
ERROR:	

CASO DE USO REPORTE UTILIDAD

Identificador Caso de Uso	CU_ REPORTE UTILIDAD
Nombre del caso de Uso	Reporte Utilidad por producto
Actores	Usuario: Administrador
Propósito	Ver el reporte de compras por fecha
Visión General	El sistema permite ver el reporte de la utilidad en dinero y porcentaje de productos comprados y vendidos.
Tipo	Primario y Esencial
Referencia	Requerimiento 21
CURSO TÍPICO DE REFERENCIAS	
ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
El usuario da clic en el botón "VER" del menú escogemos la opción Reportes y damos clic en el reporte Utilidad por Producto.	5. El sistema le muestra la ventana para Visualizar el informe
CURSOS ALTERNOS	
ERROR:	

CASO DE USO REPORTE INVENTARIO

Identificador Caso de Uso	CU_ REPORTE INVENTARIO
Nombre del caso de Uso	Reporte Inventario
Actores	Usuario: Administrador
Propósito	Ver el reporte de inventario por fecha
Visión General	El sistema permite ver inventario del sistema de los productos en un rango de fechas.
Tipo	Primario y Esencial
Referencia	Requerimiento 22
CURSO TÍPICO DE REFERENCIAS	
ACCIÓN DEL ACTOR	REPUESTA DE SISTEMA
1. El usuario administrador o contador requiere acceder a la aplicación	2. Muestra una ventana para que el usuario ingrese su cedula y Password.
3. El Usuario ingresa su cedula y Password	4. Muestra la ventana principal de la aplicación
El usuario da clic en el botón “VER” del menú escogemos la opción Reportes y damos clic en el reporte Inventario.	5. El sistema le muestra la ventana para seleccionar desde que fecha y hasta que fecha.
6. El usuario ingresa las fechas y da clic en aceptar.	7. El sistema permite visualizar el informe.
CURSOS ALTERNOS	
ERROR:	

ANEXO DIAGRAMAS

DIAGRAMA DE CASO DE USO AUTENTICACIÓN

-

DIAGRAMA DE CASO DE USO GESTIÓN EMPLEADOS

-

DIAGRAMA DE CASO DE USO GESTIÓN DE CLIENTES

✎

DIAGRAMA DE CASO DE USO GESTIÓN DE PROVEEDORES

✎

DIAGRAMA DE CASO DE USO GESTIÓN PRODUCTOS

...

DIAGRAMA DE CASO DE USO COMPRAS

■

DIAGRAMA DE CASO DE USO VENTAS

■

DIAGRAMA DE CASO DE USO MODIFICAR UTILIDAD PRECIOS

↵

DIAGRAMA DE CASO DE USO BÚSQUEDAS

↵

DIAGRAMA DE CASO DE USO IMPRIMIR FACTURA

=

DIAGRAMA DE CASO DE USO ANULAR FACTURA

=

DIAGRAMA DE CASO DE USO CAMBIO CLAVE

-

DIAGRAMA DE CASO DE USO REPORTE KARDEX

-



DIAGRAMA DE CASO DE USO REPORTE KARDEX POR PRODUCTO

■

DIAGRAMA DE CASO DE USO REPORTE VENTAS POR FECHA

■

DIAGRAMA DE CASO DE USO REPORTE COMPRAS POR FECHAS

="

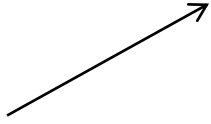


DIAGRAMA DE CASO DE USO REPORTE PRODUCTO UTILIDAD

="

DIAGRAMA DE CASO DE USO REPORTE INVENTARIO

4

BIBLIOGRAFÍA

- (1). GAMMA, ERICH., Y OTROS. *Design Patterns*. Estados Unidos : Addison Wesley, 2005.
- (2). Wikipedia.org. wikipedia.org. [En línea] 7 de 5 de 2014. [Citado el: 10 de 10 de 2013.] http://es.wikipedia.org/wiki/Patr%C3%B3n_de_dise%C3%B1o.
- (3). Trygve Reenskaug, James O. Coplien. artima developer. [En línea] 20 de 03 de 2009. [Citado el: 10 de 10 de 2013.] http://www.artima.com/articles/dci_vision.html.
- (4). Yii Software LLC. yii framework. [En línea] [Citado el: 10 de 10 de 2013.] <http://www.yiiframework.com/doc/guide/1.1/en/basics.best-practices>.
- (5). Wikipedia.org. Wikipedia. [En línea] 7 de 5 de 2014. [Citado el: 10 de 11 de 2013.] <http://es.wikipedia.org/wiki/Modelo%20%80%93vista%20%80%93controlador#Historia>.
- (6). Estruch, Juan José. <http://jjestruch.wordpress.com/>. [En línea] 21 de 2 de 2012. [Citado el: 25 de 10 de 2013.] <http://jjestruch.wordpress.com/2012/02/21/arquitectura-ddd-domain-driven-design-asp-net-mvc/>.
- (7). Fuertes Ortega V., Guevara Morocho J. *Análisis del patrón Modelo Vista Controlador implementado en lenguajes de software libre para el desarrollo de aplicaciones web*. Riobamba : s.n., 2010.
- (8). Saucedo, Luis Antonio. <http://es.scribd.com/>. [En línea] 11 de 03 de 2014. [Citado el: 07 de 06 de 2014.] <http://es.scribd.com/doc/211851362/51278468-Modelo-Vista-Controlador>.
- (9). <http://patronesdisenoo.blogspot.com/>. [En línea] 8 de 07 de 2013. [Citado el: 30 de 11 de 2013.] <http://patronesdisenoo.blogspot.com/2013/07/modelo-vista-controlador-mvc.html>.

- (10). *WPF Apps With The Model-View-ViewModel Design Pattern*. Smith, Josh. 2009, MSDN Magazine.
- (11). Llorante, Cesar; Castro, Unai; Ramos, Miguel; Nelson Javier. *Guia de Arquitectura N Capas orientada al dominio con Net. 4.0*. España : Krasis Consulting, 2010.
- (12). Wikipedia.org. Wikipedia.org. *Wikipedia.org*. [En línea] 8 de 1 de 2013. [Citado el: 21 de 2 de 2014.]
http://es.wikipedia.org/wiki/Pruebas_de_rendimiento_del_software.
- (13). MSDN Microsoft. Microsoft Developer Network. [En línea] Microsoft, 01 de 01 de 2012. [Citado el: 25 de 10 de 2013.] [http://msdn.microsoft.com/es-es/library/bb385914\(v=vs.110\).aspx](http://msdn.microsoft.com/es-es/library/bb385914(v=vs.110).aspx).
- (14). MSDN Microsoft . Microsoft Developer Network. [En línea] Microsoft, 01 de 01 de 2012. [Citado el: 01 de 12 de 2013.] [http://msdn.microsoft.com/es-es/library/dd293540\(v=vs.110\).aspx](http://msdn.microsoft.com/es-es/library/dd293540(v=vs.110).aspx).
- (15). Wayne W., Daniel. Prueba de Hipotesis. [aut. libro] Daniel Wyane W. *Bioestadística, Base para el Analisis de las ciencias de la salud*. s.l. : LimusaWiley, 2008.
- (16). Meier, J.D; Farre, Carlos; Bansode, Bansode, Prashant; Barber, Scott; Rea, Dennis. *Performance Testing Guidance for Web Applications*. Estados Unidos : Microsoft Corporation, 2007.
- (17). Palacio, Juan. *Flexibilidad con Scrum*. s.l. : safeCreative, 2008.
- (18). Lopez, Marcelo; Stura, Nestor; Soria, Jose Luis. *Scrum Guide*. s.l. : scrum.org, 2010.

