



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**

**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**

**ESCUELA DE INGENIERÍA EN SISTEMAS**

**“ANÁLISIS DE RENDIMIENTO DE FRAMEWORKS PARA INTERFACES ENTRE PRISM Y**

**MVVM LIGHT APLICADO A LA EMPRESA CYBERTRONIC”**

**“TESIS DE GRADO PREVIA OBTENCIÓN DEL TÍTULO DE**

**INGENIERO EN SISTEMAS INFORMÁTICOS”**

**MARÍA FERNANDA GIRÓN BONILLA**

**VERÓNICA JANETH ARGUELLO PAZMIÑO**

**RIOBAMBA – ECUADOR**

**2014**

*Agradecemos en primera instancia a Dios por que con su misericordia e inmenso amor nos ha permitido llegar a este punto importante en nuestras vidas; a nuestros padres por brindarnos su apoyo incondicional en cada una de nuestra etapas, a nuestros familiares y amigos por sus palabras, consejos y ayudas en cada uno de nuestros pasos, a nuestros profesores en especial al Dr. Julio Santillán, por sus criterios en el presente trabajo, Ing. Jaime Zapata, Ing. Ivonne Rodríguez, por su tiempo compartido, sus valiosas sugerencias, a la escuela de ingeniería en sistemas por ser forjadora en nuestros estudios y a la empresa CYBERTONIC por permitirnos realizar la parte aplicativa de la presente investigación.*

**María Fernanda Girón Bonilla**

**Verónica Janeth Arguello Pazmiño**

*Dedico este proyecto de tesis a Dios, a mis padres, a mi familia. A Dios porque siempre ha estado conmigo en mi vida cuidándome y brindando sus bendiciones, a mis padres Marcelo y Marisol quienes han sido los pilares desde mi niñez hasta el día de hoy, a mi hermana Belén que con su apoyo me ha ayudado a cumplir con esta meta, a mi tío Marco y en especial a mi hija Sophia, además dedico a todos quienes han creído en mí depositando su confianza en cada reto de mi vida sin dudar de mi capacidad e inteligencia.*

**María Fernanda Girón Bonilla**

*La concepción de este proyecto de tesis dedico a Dios, a mis padres, a mi esposo e hijo motores fundamentales que inundan mi corazón con ganas de seguir adelante venciendo obstáculos que a lo largo del camino se presentan y que me cobijan en sus ramas de protección, apoyo, confianza pero sobre todo demostrándome su amor.*

**Verónica Janeth Arguello Pazmiño**

## FIRMAS RESPONSABLES Y NOTAS

**NOMBRES**

**FIRMA**

**FECHA**

ING. IVÁN MENES CAMEJO

**DECANO DE LA FACULTAD**

**DE INFORMÁTICA Y**

**ELECTRÓNICA**

---

---

ING. JORGE HUILCA

**DIRECTOR DE LA ESCUELA**

**DE INGENIERÍA EN**

**SISTEMAS**

---

---

DR. JULIO SANTILLÁN

**DIRECTOR DE TESIS**

---

---

DR. NARCISA SALAZAR

**MIEMBRO DE TESIS**

---

---

**DIRECTOR DEL CENTRO DE**

**DOCUMENTACIÓN**

---

---

## **RESPONSABILIDAD DEL AUTOR**

“Nosotros María Fernanda Girón Bonilla y Verónica Janeth Arguello Pazmiño, somos los responsables de las ideas, doctrinas y resultados expuestos en esta Tesis de Grado, y el patrimonio intelectual de la misma pertenece a la Escuela Superior Politécnica de Chimborazo.”

---

María Fernanda Girón Bonilla

---

Verónica Janeth Arguello Pazmiño

## ÍNDICE DE ABREVIATURAS

**DCL:** Lenguaje de control de datos.

**DDL:** Lenguaje de definición de datos.

**DML:** Lenguaje de manipulación de datos.

**DSDL:** Lenguaje de definición del almacenamiento de los datos.

**GDI:** Deposito de interfaz Gráfica.

**IDE:** Entorno de desarrollo integrado

**MVC:** Modelo Vista Controlador.

**MVVM:** Modelo Vista Vista Modelo

**SGBD:** Sistema de gestión de bases de datos relacionales

**USA:** Estados Unidos de América

**WPF:** Base de Presentación de Windows (Windows Presentation Foundation)

# ÍNDICE GENERAL

RESPONSABILIDAD DEL AUTOR.....	- 5 -
ÍNDICE DE ABREVIATURAS .....	- 6 -
ÍNDICE GENERAL .....	- 7 -
ÍNDICE DE FIGURAS .....	- 10 -
ÍNDICE DE TABLAS.....	- 11 -
INTRODUCCIÓN.....	- 12 -
CAPÍTULO I .....	- 14 -
MARCO REFERENCIAL .....	- 14 -
1.1.    Antecedentes .....	- 14 -
1.2.    Justificación del Proyecto de Tesis.....	- 15 -
1.2.1.    Justificación Teórica .....	- 15 -
1.2.2.    Justificación Práctica .....	- 16 -
1.3.    Objetivos .....	- 16 -
1.3.1.    Objetivo General .....	- 16 -
1.3.2.    Objetivos Específicos.....	- 17 -
1.4.    Hipótesis.....	- 17 -
1.5.    Métodos y Técnicas.....	- 17 -
1.5.1.    Métodos .....	- 17 -
1.5.2.    Técnicas.....	- 18 -
CAPÍTULO II .....	- 19 -
MARCO TEORICO.....	- 19 -
2.1. Framework .....	- 19 -
2.2. Base de Datos.....	- 19 -
2.2.1. Microsoft SQL Server.....	- 20 -
2.3. Microsoft Visual Studio .....	- 21 -
2.3.1. Visual Basic.....	- 22 -
2.3.2. Visual C#.....	- 22 -
2.4. Ciclo de Vida Software .....	- 22 -
2.5. Arquitectura del Software.....	- 23 -

2.5.1. Patrón de Diseño Modelo Vista Vista Modelo .....	- 24 -
2.5.2. Capas de Aplicación.....	- 25 -
2.5.3. Ventajas de las Capas .....	- 26 -
2.6. Windows Presentation Foundation .....	- 27 -
2.7. Rendimiento del Framework.....	- 27 -
2.8. Framework Prism .....	- 27 -
2.8.1. Ventajas.....	- 28 -
2.8.2. Componentes .....	- 30 -
2.8.3. Diseño de la Aplicación .....	- 30 -
2.9. Framework MVVM Light .....	- 32 -
2.9.1. Ventajas.....	- 32 -
2.9.1. Componentes.....	- 33 -
CAPÍTULO III .....	- 35 -
ANÁLISIS DE RENDIMIENTO DE LOS FRAMEWORKS PRIMS Y MVVM LIGHT .....	- 35 -
3.1. Definición de los parámetros a comparar.....	- 36 -
3.2. Métodos técnicas y procedimientos.....	- 38 -
3.2.1. Métodos .....	- 38 -
3.2.2. Técnicas y procedimientos.....	- 38 -
3.3. Análisis Comparativo.....	- 56 -
3.4. Puntajes Alcanzados.....	- 57 -
3.5 Interpretación .....	- 59 -
3.6 Resultados del Análisis .....	- 64 -
3.7. Comprobación de la Hipótesis .....	- 66 -
CAPÍTULO IV .....	- 71 -
DESARROLLO DEL SISTEMA VAMEG SOFT PARA LA EMPRESA CYBERTRONIC.....	- 71 -
4.1. Gestión del Proyecto.....	- 71 -
4.2. Desarrollo de la Metodología XP.....	- 72 -
4.2.1. Fase I: Planificación .....	- 72 -
4.2.2. Fase II: Diseño del software .....	- 87 -
4.2.3. Fase III: Codificación.....	- 97 -
4.2.4. Fase IV: Pruebas de Funcionamiento .....	- 98 -
CONCLUSIONES .....	- 108 -
RECOMENDACIONES.....	- 109 -

RESÚMEN .....	- 110 -
SUMARY .....	- 111 -
BIBLIOGRAFÍA.....	- 113 -

## ÍNDICE DE FIGURAS

FIGURA II. 1 Capas de aplicación MVVM .....	- 26 -
FIGURA II. 2: Diseño de la aplicación de PRISM .....	- 31 -
FIGURA II. 3: Regiones de la Shell en PRISM .....	- 31 -
FIGURA III. 4: Valores Porcentuales de los parámetros a usar .....	- 37 -
FIGURA III. 5: Análisis del Parámetro de Disponibilidad de Información.....	- 60 -
FIGURA III. 6: Comparación del parámetro Líneas de Código.....	- 60 -
FIGURA III. 7: Comparación del parámetro Tiempo de Desarrollo .....	- 61 -
FIGURA III. 8: Comparación del parámetro Uso del CPU .....	- 62 -
FIGURA III. 9: Comparación del parámetro Uso de la Memoria RAM .....	- 63 -
FIGURA III. 10: Comparación del parámetro Tiempo de Respuesta .....	- 63 -
FIGURA III. 11: Comparación del parámetro Tiempo de Mantenimiento .....	- 64 -
FIGURA III. 12: Totales Alcanzados en la Comparación .....	- 66 -
FIGURA III. 13: Chi Cuadrado.....	- 67 -
FIGURA III. 14: Comprobación de Hipótesis.....	- 69 -
FIGURA IV. 15: Fases de la Metodología XP .....	- 72 -
FIGURA IV. 16: Flujo de Proceso del Sistema.....	- 74 -
FIGURA IV. 17: Usuarios del Sistema VAMEG SOFT .....	- 77 -
FIGURA IV. 18: Rol de Usuario Administrador .....	- 77 -
FIGURA IV. 19: Rol de Usuario Gerente .....	- 77 -
FIGURA IV. 20: Rol Usuario Secretaria .....	- 78 -
FIGURA IV. 21: Rol de Usuario Contador .....	- 78 -
FIGURA IV. 22: Rol de Usuario Bodeguero.....	- 78 -
FIGURA IV. 23: Diseño de Base de Datos .....	- 87 -
FIGURA IV. 24: Ingreso al Sistema.....	- 88 -
FIGURA IV. 25: Menú del Sistema .....	- 88 -
FIGURA IV. 26: Movimiento de Datos del Cliente .....	- 89 -
FIGURA IV. 27: Buscar Datos del Cliente .....	- 89 -
FIGURA IV. 28: Facturación Venta.....	- 90 -
FIGURA IV. 29: Movimiento Datos Proveedor .....	- 90 -
FIGURA IV. 30: Buscar Proveedor .....	- 91 -
FIGURA IV. 31: Facturación Compra .....	- 91 -
FIGURA IV. 32: Movimiento Datos Categoría.....	- 92 -
FIGURA IV. 33: Movimiento Datos Subcategoría .....	- 92 -
FIGURA IV. 34: Movimiento Datos Marca.....	- 92 -
FIGURA IV. 35: Movimiento Datos Producto .....	- 93 -
FIGURA IV. 36: Movimiento Datos Usuario.....	- 93 -
FIGURA IV. 37: Diagrama de Bloques y Despliegue del Sistema.....	- 97 -
FIGURA IV. 38: Codificación .....	- 98 -

## ÍNDICE DE TABLAS

TABLA III. I: Valores de los Parámetros a usar .....	- 37 -
TABLA III. II: Requerimientos prototipo .....	- 39 -
TABLA III. III: Toma de mediciones.....	- 40 -
TABLA III. IV: Disponibilidad de la información.....	- 56 -
TABLA III. V: Valores de las Medias de los prototipos.....	- 58 -
TABLA III. VI: Análisis Parámetro Disponibilidad de la Información.....	- 59 -
TABLA III. VII: Valores para la comprobación análisis .....	- 65 -
TABLA III. VIII: Datos Observados Chi Cuadrado .....	- 68 -
TABLA III. IX: Valores esperados.....	- 68 -
TABLA III. X: Valores de contingencia.....	- 69 -
TABLA IV. XI: Historia de Usuario Ingreso al módulo Administración.....	- 79 -
TABLA IV. XII: Historia de Usuario Ingreso al módulo Facturación .....	- 80 -
TABLA IV. XIII: Historia de Usuario Ingreso al módulo de Inventario.....	- 80 -
TABLA IV. XIV: Historia de Usuario Ingresar, actualizar y eliminar clientes.....	- 81 -
TABLA IV. XV: Historia de Usuario Ingresar, actualizar y eliminar proveedores .....	- 81 -
TABLA IV. XVI: Historia de Usuario Ingresar, actualizar y eliminar productos .....	- 82 -
TABLA IV. XVII: Historia de Usuario Ingresar, actualizar y eliminar marcas.....	- 82 -
TABLA IV. XVIII: Historia de Usuario Ingresar, actualizar y eliminar categorías.....	- 83 -
TABLA IV. XIX: Historia de Usuario Ingresar, actualizar y eliminar subcategorías .....	- 83 -
TABLA IV. XX: Historia de Usuario realizar búsquedas de clientes, proveedores, productos.-	- 84 -
TABLA IV. XXI: Historia de Usuario Obtener reportes de las compras, ventas, clientes, proveedores .....	- 84 -
TABLA IV. XXII: Historia de Usuario Obtener reportes de las cuentas por cobrar y pagar .....	- 85 -
TABLA IV. XXIII: Diccionario de Datos.....	- 94 -
TABLA IV. XXIV: Ingreso, modificación y eliminación de clientes .....	- 99 -
TABLA IV. XXV: Ingreso, modificación y eliminación de proveedores .....	- 100 -
TABLA IV. XXVI: Ingreso, modificación y eliminación de productos .....	- 101 -
TABLA IV. XXVII: Ingreso, modificación y eliminación de categoría.....	- 102 -
TABLA IV. XXVIII: Ingreso, modificación y eliminación de subcategoría .....	- 103 -
TABLA IV. XXIX: Ingreso, modificación y eliminación de marcas .....	- 104 -
TABLA IV. XXX: Búsqueda de clientes, proveedores, productos.....	- 105 -
TABLA IV. XXXI: Reportes de las compras, ventas, productos, clientes y proveedores.....	- 106 -
TABLA IV. XXXII: Información sobre las cuentas por cobrar y pagar.....	- 107 -

## INTRODUCCIÓN

El presente trabajo de investigación de tesis previo a la obtención del título de Ingeniería en Sistemas Informáticos, trata de “ANÁLISIS DE RENDIMIENTO DE FRAMEWORKS PARA INTERFACES ENTRE PRISM Y MVVM LIGHT APLICADO EN LA EMPRESA CYBERTRONIC.”.

Es un estudio y análisis investigativo a fondo del rendimiento de diferentes Frameworks para obtener como resultado el más óptimo para interfaces.

Como precedente cuando se trata de desarrollar aplicaciones en lenguaje .Net, la mayoría de desarrolladores implementan sus proyectos en el Frameworks que viene por defecto o en el que tenga mayor conocimiento, sin realizar una investigación previa sobre los Frameworks que se ajusten a las necesidades y a su vez ofrezcan el mejor rendimiento y estabilidad.

El presente trabajo contiene los siguientes capítulos:

En el **capítulo 1** se presenta el planteamiento de la investigación antecedentes, hipótesis, métodos y técnicas; es todo el marco referencial para el desarrollo de la tesis.

En el **capítulo 2** se detalla los aspectos teóricos motivos de la investigación conceptos, terminologías referenciadas al objeto de estudio.

El **capítulo 3** trata del desarrollo del análisis de rendimiento de los Frameworks donde se seleccionan las herramientas para determinar cual tiene el mejor desempeño, sometiendo al sistema a diferentes escenarios de pruebas.

Se han determinado los indicadores e índices que sirven para analizar los Frameworks, finalizando con la demostración de la hipótesis.

En el **capítulo 4** se detalla la parte aplicativa de la tesis, contiene los requisitos de ingeniería de software, estándares de desarrollo, todo referente a la empresa CYBERTRONIC. El manejo de usuarios será mediante un control de acceso de usuario y contraseña, se maneja artículos, proveedores, clientes de la empresa

# CAPÍTULO I

## MARCO REFERENCIAL

### 1.1. Antecedentes

Existen varias herramientas y lenguajes de programación para el desarrollo de aplicaciones de calidad. El verdadero problema está, en que arquitectura utilizar y cuáles son los posibles Frameworks que soportan dicho requerimiento.

Se ha adoptado la utilización de diferentes Frameworks porque permiten afrontar de forma estructurada y según los últimos patrones, aplicaciones de visualización de datos con interfaces de usuario; entre estos Frameworks podemos mencionar “Prism, Cinch, Caliburn. Micro, ReactiveUI, MVVM LIGHT”, entre otros.

Se ha adoptado realizar el análisis de rendimiento de dos Frameworks de todos los que se menciona por medio de características comparables y se ha escogido Prism y MVVM LIGHT.

Hemos buscado un problema con cierto nivel de complejidad para ello se requiere desarrollar un sistema que cumpla con los requerimientos planeados en dicho problema, y así poder reflejar las funcionalidades que ofrecen los Frameworks.

En la actualidad la Empresa CYBERTRONIC requieren de un sistema para gestión y administración de la información de la empresa, y que la misma se encuentre de forma confiable.

Para el desarrollo de este sistema se pretende utilizar el Framework que provea el mejor rendimiento luego de efectuar la comparación

## **1.2. Justificación del Proyecto de Tesis**

### **1.2.1. Justificación Teórica**

- El presente proyecto de tesis plantea la utilización de Frameworks que proporcionen aplicaciones estructuradas con la arquitectura Modelo Vista Vista Modelo además de un estándar para el desarrollo de aplicaciones informáticas dentro de la Empresa CYBERTRONIC a través de Frameworks que por medio de estos se establecerá una forma más eficiente de desarrollar aplicaciones.
- Framework Prism proporciona una guía para ayudar de forma fácil el diseño y la construcción flexible de aplicaciones, proporciona orientación y una biblioteca reutilizable, ayuda al desarrollo de aplicaciones modulares y a comprender, aplicar y utilizar patrones de diseño.

- Framework MVVM LIGHT, ayuda a separar la vista de su modelo que crea aplicaciones que son más limpio y más fácil de mantener y ampliar. También crea aplicaciones comprobables y le permite tener una capa de interfaz de usuario.
- El principal objetivo de utilizar Frameworks es construir una alternativa sencilla y fácil, además de un entorno de interfaz de usuario que llame la atención y que atraiga al usuario.

### **1.2.2. Justificación Práctica**

El desarrollo de la aplicación se realizará tanto en Prism como en MVVM LIGHT Frameworks de desarrollo para interfaces que nos proporcionarán un gran rendimiento

El desarrollo del sistema VAMEG SOFT se implementará para la gestión y administración de la empresa CYBERTRONIC dividiendo así los siguientes módulos:

- Módulo de Inventarios
- Módulo de Facturación
  - Compras
  - Ventas
- Módulo de administración del sistema

## **1.3. Objetivos**

### **1.3.1. Objetivo General**

Realizar un análisis de rendimiento de Frameworks para interfaces entre Prism y MVVM Light aplicado en la Empresa CYBERTRONIC

### **1.3.2. Objetivos Específicos**

- Estudiar los Frameworks en Microsoft .NET Open source MVVM
- Determinar los parámetros de rendimiento para el desarrollo de aplicaciones entre Prism y MVVM Light.
- Evaluar los Frameworks y mediante pruebas de carga seleccionar el mejor conforme a los parámetros de rendimiento.
- Implementar la aplicación software VAMEG SOFT en base al Framework seleccionado para la Empresa CYBERTRONIC.

### **1.4. Hipótesis**

La implementación de una aplicación con Prism es el más eficiente en Rendimiento, que una aplicación desarrollada en MVVM Light.

### **1.5. Métodos y Técnicas**

#### **1.5.1. Métodos**

El método utilizado como guía para la presente investigación es el método científico, el cual contempla los siguientes puntos:

- ✓ El planteamiento del problema que en este caso es el análisis comparativo de los Frameworks Prism y MVVM Light
- ✓ El apoyo del proceso previo a la formulación de la Hipótesis
- ✓ Levantamiento de información necesaria
- ✓ Análisis e interpretación de Resultados

- ✓ Proceso de Comprobación de la Hipótesis

Para el desarrollo y la implementación de la aplicación del sistema VAMEG SOFT para la gestión y administración de la empresa CYBERTRONIC se empleara la metodología ágil XP.

### **1.5.2. Técnicas**

Para la recopilación de la información necesaria que sustente el presente trabajo de investigación, se ha establecido como técnica lo siguiente:

- ✓ Revisión de los Frameworks que se van a analizar
- ✓ Observación
- ✓ Técnicas de comprobación de la hipótesis
- ✓ Prototipo

## **CAPÍTULO II**

### **MARCO TEORICO**

A continuación se detalla la información documental para confeccionar el diseño metodológico de la investigación

#### **2.1. Framework**

“Un Framework, se refiere a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación” [1]. Se considera un framework a una aplicación que se puede configurar añadiendo líneas de código para que tenga una estructura completa.

#### **2.2. Base de Datos**

“La Base de datos es un conjunto de información almacenada en memoria auxiliar que permite acceso directo y un conjunto de programas que manipulan esos datos.” [2]

Una base de datos se la entiende como un almacén de información la cual permite guardar muchos datos de manera organizada para posteriormente usar más

fácilmente. Toda esta información es almacenada en forma de registro para optimizar la misma.

### **2.2.1. Microsoft SQL Server**

“Es un sistema de gestión de bases de datos relacionales (SGBD) basada en el lenguaje Transact –SQL, capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea”. [3]

Adicionalmente, Microsoft ha puesto gratuitamente a disposición de todo el mundo una versión reducida de MS SQL Server llamada SQL Server Express Edition cuyas principales limitaciones son que no soporta bases de datos superiores a 4 GB de tamaño, únicamente utiliza un procesador y un Gb de RAM, y no cuenta con Agente de SQL Server

Microsoft SQL Server presenta algunas características:

- ✓ Escalabilidad, estabilidad y seguridad.
- ✓ Soporta procedimientos almacenados.
- ✓ Incluye también un potente entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.
- ✓ Permite trabajar en modo cliente-servidor donde la información y datos se alojan en el servidor y las terminales o clientes de la red sólo acceden a la información.
- ✓ Además permite administrar información de otros servidores de datos.

## 2.3. Microsoft Visual Studio

“Es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows.

Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic.NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros.” [4]

A partir de la versión 2002 Visual Studio permite a los desarrolladores crear aplicaciones, sitios y aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma. NET. Así se pueden crear aplicaciones que se intercomunican entre estaciones de trabajo, páginas web y dispositivos móviles.

A partir de la versión 2005 Microsoft ofrece gratuitamente las Express Editions. Estas son varias ediciones básicas separadas por lenguajes de programación o plataforma enfocadas para novatos y entusiastas. Estas ediciones son iguales al entorno de desarrollo comercial pero sin características avanzadas. Las ediciones que hay son:

- ✓ Visual Basic Express Edition
- ✓ Visual C# Express Edition
- ✓ Visual C++ Express Edition
- ✓ Visual J# Express Edition (Visual J# desapareció en Visual Studio 2008)
- ✓ Visual Web Developer Express Edition (para programar en ASP.NET).

### **2.3.1. Visual Basic**

“Es un ambiente gráfico de desarrollo de aplicaciones para el sistema operativo Microsoft Windows. Las aplicaciones creadas con Visual Basic están basadas en objetos y son manejadas por eventos. Visual Basic se deriva del lenguaje Basic, el cual es un lenguaje de programación estructurado. Sin embargo, Visual Basic emplea un modelo de programación manejada por eventos”. [5].

Es un lenguaje que se creó con la finalidad de simplificar la programación con un ambiente de desarrollo fácil, se puede utilizar para la edición de código fuente en todas las versiones para Windows, de esta manera compilar un código nativo obteniendo una mejora de rendimiento considerado que soporta características propias de los lenguajes orientados a objetos.

### **2.3.2. Visual C#**

“Es un lenguaje de programación que se ha diseñado para compilar diversas aplicaciones que se ejecutan en .NET Framework” [6].

Está diseñado para la infraestructura del lenguaje común, incluye mejoras de otros lenguajes siendo este independiente de los otros tipos de lenguajes, contiene tipos de datos integrados, caracteres que facilitan la escritura con el teclado de símbolos, se utilizan aplicaciones. Net siendo así simple eficaz para el desarrollo de las mismas.

## **2.4. Ciclo de Vida Software**

Representa el desarrollo del software desde su etapa inicial hasta su etapa final

- ✓ **Análisis:** Representa el alcance del proyecto; es decir que es lo que debe tener el nuevo sistema.
- ✓ **Diseño:** En esta etapa se especifican los requerimientos, características esenciales del sistema además que se fundamenta la arquitectura que se empleara la cual nos permite tener una visión del software.
- ✓ **Desarrollo:** Se elige un entorno de desarrollo de fácil manejo.
- ✓ **Pruebas:** Se verifican que los requerimientos se cumplan según lo dispuesto.
- ✓ **Mantenimiento:** Es la mejora del sistema, además se pueden adaptar nuevos requerimientos del software si éste lo necesitara posteriormente. En esta etapa se consume desde un 40 a 80% de los recursos por eso es la etapa más importante de este ciclo; es decir que se repiten todas las etapas al añadir un nuevo requerimiento teniendo como resultado la utilización de gran cantidad de tiempo y de recursos económicos.

## 2.5. Arquitectura del Software

Se denomina arquitectura lógica porque contiene un conjunto de patrones que se diseña con base a diseños y restricciones; es decir son los componentes que llevan a cabo una tarea, sus interfaces y la forma de comunicación entre ellos.

“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 llevan a cabo la funcionalidad especificada, cumpliendo con criterios de calidad, seguridad, disponibilidad, usabilidad, mantenibilidad” [7].

### 2.5.1. Patrón de Diseño Modelo Vista Vista Modelo

“Es un patrón de diseño de aplicaciones para desacoplar código de interfaz de usuario y código que no sea de interfaz de usuario. Con MVVM, defines la interfaz de usuario de forma declarativa (por ejemplo, mediante XAML) y usas el marcado de enlace de datos para vincularla a otras capas que contengan datos y comandos de usuario. La infraestructura de enlace de datos proporciona un acoplamiento débil que mantiene sincronizados la interfaz de usuario y los datos vinculados, y que enruta todas las entradas de usuario a los comandos apropiados.”[8]

El patrón MVVM organiza el código de tal forma que es posible cambiar partes individuales sin que los cambios afecten a las demás partes. Esto presenta numerosas ventajas, como las siguientes:

- ✓ Permite un estilo de codificación exploratorio e iterativo.
- ✓ Simplifica las pruebas unitarias.
- ✓ Permite aprovechar mejor herramientas de diseño como Expression Blend.
- ✓ Admite la colaboración en equipo.

Por el contrario, una aplicación con una estructura más convencional utiliza el enlace de datos únicamente en controles de lista y texto, y responde a la entrada del usuario mediante el control de eventos expuestos por los controles.

“Los controladores de eventos están acoplados a los controles y suelen contener código que manipula la interfaz de usuario directamente. Esto hace que sea difícil o

imposible reemplazar un control sin tener que actualizar el código de control de eventos.” [8].

MVVM se considera como la evolución del patrón MVC (Modelo vista Controlador), el cual intenta facilitar el trabajo de quien diseña la interfaz de usuario con el que genera el código que la sustenta; este patrón consta de tres partes muy diferentes puesto que sus capas tienen una forma de comunicación: la vista y la vista modelo una comunicación bidireccional mientras que el modelo solo recibe órdenes de la vista modelo.

### 2.5.2. Capas de Aplicación

Las capas del MVVM se las puede clasificar en las siguientes:

- “La capa de **modelo** incluye todo el código que implementa la lógica principal de la aplicación y define los tipos requeridos para modelar el dominio de la aplicación. Esta capa es completamente independiente de las capas de vista y modelo de vista.
- La capa de **vista** define la interfaz de usuario que utiliza marcado declarativo. El marcado de enlace de datos define la conexión entre componentes específicos de la interfaz de usuario y diversos miembros de modelo de vista (y, en ocasiones, de modelo).
- La capa de **modelo de vista** proporciona destinos de enlace de datos para la vista. En muchos casos, el modelo de vista expone el modelo directamente o proporciona miembros que encapsulan miembros de modelo específicos. El

modelo de vista también puede definir miembros para realizar un seguimiento de los datos que son relevantes para la interfaz de usuario pero no para el modelo, como el orden de visualización de una lista de elementos.”[8]

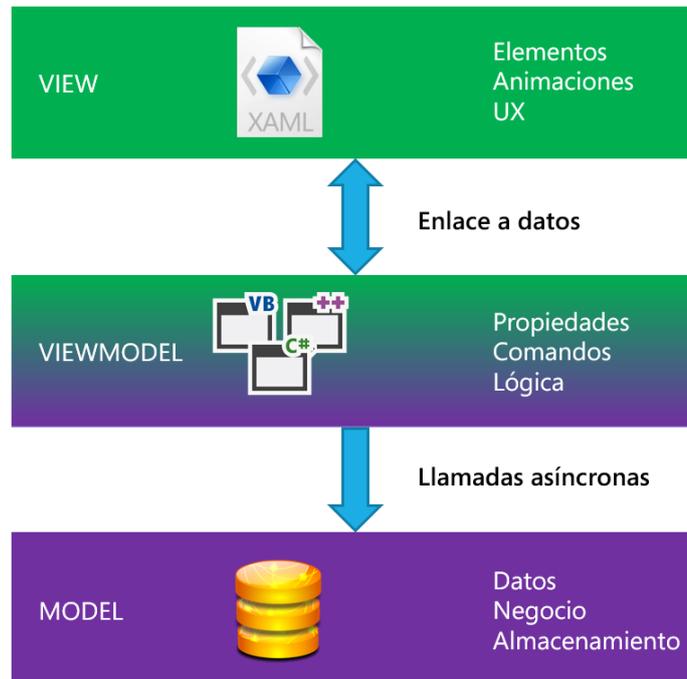


FIGURA II. 1 Capas de aplicación MVVM

FUENTE: <http://mvpway.files.wordpress.com/2013/04/mvvm.png>

### 2.5.3. Ventajas de las Capas

- “Permite facilitar la comprensión del código. Esto se debe a que el código de características específicas a menudo es independiente de otro código, lo que facilita su aprendizaje y permite reutilizarlo en otras aplicaciones.
- Facilita la realización de pruebas automatizadas del código que no corresponde a la interfaz de usuario. Microsoft Visual Studio admite proyectos de pruebas unitarias, que permiten comprobar el diseño del código durante el desarrollo, así como identificar y diagnosticar errores.

- Permite aislar el impacto de los cambios, de forma que es mucho menos arriesgado probar nuevas características, corregir errores e incorporar las contribuciones de los colaboradores.” [8]

Por tanto se puede concluir que se acopla a cambios y a un diagnóstico de errores.

## **2.6. Windows Presentation Foundation**

“Windows Presentation Foundation (WPF o Base de Presentación de Windows), es una tecnología desarrollada por Microsoft que permite potenciar las capacidades de desarrollo de interfaces de interacción integrando y ampliando las mejores características de las aplicaciones Windows y de las aplicaciones web.

Windows Presentation Foundation es un subsistema gráfico para el renderizado de interfaces de usuario de aplicaciones para Windows. Fue diseñado para eliminar las dependencias en el subsistema GDI.” [9]

## **2.7. Rendimiento del Framework**

El rendimiento se considera como la efectividad o la eficiencia de lograr un resultado, de una tarea, se puede medir el rendimiento tanto de hardware como de software y así poder obtener un resultado confiable.

## **2.8. Framework Prism**

“El pasado Febrero de 2012, Microsoft liberó la versión 4.1 de Prism, un Framework pensado para facilitar la creación de aplicaciones con WPF y Silverlight. Sobre todo,

permite afrontar de forma estructurada y según los últimos patrones, aplicaciones de visualización de datos con interfaces de usuario ricas y con un negocio complejo.” [10]

Prism por su facilidad y flexibilidad en WPF hace que el desarrollo de las aplicaciones sea más sencilla para el programador ya sea de escritorio, Silverlight Rich Internet, Applications (RIA) y Windows Phone 7. Por esta razón Prism presenta:

- ✓ **“Modularidad:** Particionar los componentes para crear aplicaciones a partir de módulos integrados pero independientes.
- ✓ **Interfaz de usuario:** desacoplamiento de los componentes de UI con respecto al resto de la aplicación.
- ✓ **Servicios horizontales** (log, authentication,etc) separados de servicios verticales (business logic). Esto facilita la creación de capas realmente desacopladas.”[10]

### 2.8.1. Ventajas

Prism presenta las siguientes ventajas:

- ✓ **“Reutilizar.-** Promueve la reutilización al permitir que los componentes y servicios que se desarrollen con facilidad, probados e integrados en una o más aplicaciones. La reutilización se puede lograr a nivel de componentes a través de la reutilización de los componentes de la unidad probada que pueden ser fácilmente descubiertos e integrados en tiempo de ejecución a través de la inyección de dependencia, y en el nivel de aplicación a través del uso de

módulos que encapsulan las capacidades a nivel de aplicación que pueden ser reutilizados en todas las aplicaciones.

- ✓ **Extensibilidad.**- Prism ayuda a crear aplicaciones que sean fáciles de extender mediante la gestión de dependencias de los componentes, lo que los componentes sean más fácilmente integrados o reemplazados con implementaciones alternativas en tiempo de ejecución, y proporcionando la capacidad para descomponer una aplicación en módulos que se pueden actualizar y desplegar de forma independiente. Muchos de los componentes en la propia Biblioteca Prisma también se puede ampliar o sustituir.
- ✓ **Flexibilidad.**- Prism ayuda a crear aplicaciones flexibles al permitir que sean más fáciles de actualizar a medida que se desarrollan y se integran nuevas capacidades. Prism también permite a las aplicaciones de WPF a desarrollar el uso de servicios y componentes comunes, lo que permite que la aplicación se despliega y se consume en la forma más adecuada. También permite que las aplicaciones proporcionan experiencias diferentes según la función o configuración.
- ✓ **Equipo de Desarrollo.**- Prism promueve el desarrollo del equipo al permitir que equipos separados para desarrollar e incluso despliegan diferentes partes de la aplicación de forma independiente. Prisma ayuda a minimizar las dependencias entre equipos y permite que los equipos se centran en diferentes áreas funcionales (como el diseño de la interfaz de usuario, la aplicación lógica de negocio, y el desarrollo de código de infraestructura), o en diferentes áreas

funcionales de nivel empresarial (por ejemplo, el perfil, las ventas, el inventario, etc.).

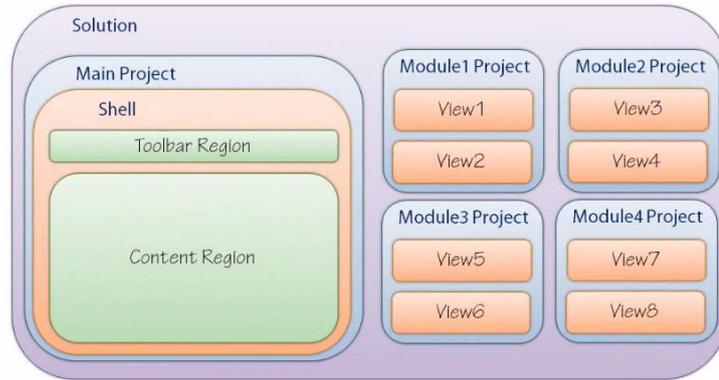
- ✓ **Calidad.**- Prism puede ayudar a aumentar la calidad de las aplicaciones al permitir que los servicios y componentes comunes a estar plenamente probados y puestos a disposición de los equipos de desarrollo. Además, al proporcionar implementaciones plenamente probadas de patrones de diseño comunes, y la orientación necesaria para aprovechar plenamente, Prism permite a los equipos de desarrollo se centren en sus requisitos de aplicación en lugar de implementar y probar código de infraestructura.” [11]

### 2.8.2. Componentes

1. **“Shell:** plantilla que define la estructura de la interfaz de usuario. Shell contiene varias regiones.
2. **Regiones :** Las regiones se utilizan para especificar la parte específica de la concha como elementos para inyectar vista en tiempo de ejecución
3. **Módulos:** Son las principales áreas funcionales de la aplicación. Cada módulo necesita ser independiente de otras,
4. **Vistas:** Módulos contiene el número de impresiones. Vistas en Prism se construyen con patrón de diseño MVVM.
5. **Boot-Flejadora:** Este componente se encarga de crear Shell y la inicialización de la aplicación. “[12]

### 2.8.3. Diseño de la Aplicación

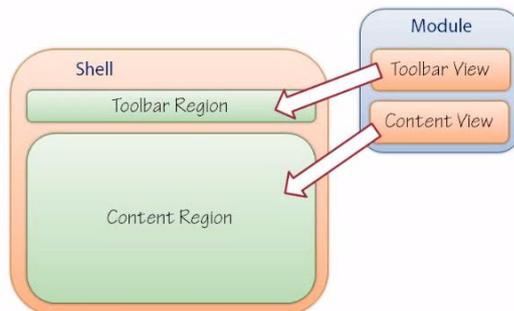
El diseño ideal de una aplicación de PRISM es la siguiente:



**FIGURA II. 2: Diseño de la aplicación de PRISM**

**FUENTE:** <http://www.codeproject.com/Articles/507077/PRISM-application-from-the-beginning-part-I>

“La solución tiene un proyecto principal que sólo contiene una Shell. Este Shell define regiones como queramos. La solución también contiene proyectos que representan módulos de la aplicación. Este módulo contiene los puntos de vista que van a ser inyectados en las regiones de la Shell” [12]



**FIGURA II. 3: Regiones de la Shell en PRISM**

**FUENTE:** <http://www.codeproject.com/Articles/507077/PRISM-application-from-the-beginning-part-I>

## 2.9. Framework MVVM Light

“MVVM Light es una implementación del patrón MVVM desarrollada por Laurent Bugnion (Galasoft) de gran éxito y que puede ser usada en proyectos WPF, Silverlight, Windows 8 y Windows Phone para acelerar y facilitar su desarrollo.

Las librerías de MVVM Light pueden ser añadidas a nuestro proyecto fácilmente mediante Nuget<sup>1</sup> y, para quien lo prefiera, también tenemos a nuestra disposición un toolkit con plantillas de proyectos de Visual Studio que las incorporan.” [13]

### 2.9.1. Ventajas

- ✓ “Proporciona toda la infraestructura necesaria para realizar la mayoría de proyectos que nos planteemos y nos va a permitir empezar a desarrollar aplicaciones usando el patrón MVVM sin apenas necesidad de desarrollar código adicional; sin duda un ahorro de tiempo
  - ✓ Posibilidad de utilizar snippets. <sup>2</sup>De este modo podremos fácilmente definir propiedades, comandos y demás con mucha más rapidez. Todos los snippets empiezan por “mvvm”, para que sean fácilmente encontrados mediante intellisense.
  - ✓ Facilita la edición de la interfaz de usuario con Expression Blend, que incluye la creación de datos en tiempo de diseño para permitir al diseñador mejorar su trabajo con controles enlazados a datos sin tener que depender de que el ViewModel se los sirva.
- MVVM Light en Visual Studio 2010 se debe hacer la instalación de Snippets

---

<sup>1</sup>“Nuget es un manejador de paquetes que permite instalar y actualizar librerías y herramientas en Visual Studio 2010, su instalación y la inserción de las referencias en los ficheros de configuración web.config y app.config.”

<sup>2</sup>“Snippet pequeña porción de código o texto de programación. Son utilizadas generalmente para minimizar la repetición de códigos, hacer más claros los algoritmos o permitir que una aplicación genere el código automáticamente.”

para que el Framework funcione sin dificultades ni errores al momento de realizar cualquier aplicación.” [13]

### 2.9.1. Componentes

Esta herramienta reúne bibliotecas y componentes auxiliares como:

- ✓ **“ViewModelBase:** Clase que se utiliza como clase base para el ViewModel modelo vista
- ✓ **Mensajero clase:** Que se utiliza para comunicar dentro de la aplicación.
- ✓ **MessageBase:** Una clase de mensaje simple, llevar información opcional sobre el remitente del mensaje.
- ✓ **GenericMessage:** Un mensaje simple con una propiedad de contenido
- ✓ **NotificationMessage:** Se utiliza para enviar una notificación (como una cadena) a un destinatario. Por ejemplo, guardar sus notificaciones lo más constante en una clase de notificaciones y luego enviar Notifications.Save a un destinatario.
- ✓ **NotificationMessage:** Igual que el anterior, pero con una propiedad de contenido genérico. Puede ser usado para pasar un parámetro al destinatario junto con la notificación.
- ✓ **NotificationMessageAction:** Envía una notificación a un destinatario y permite que el destinatario de la llamada al remitente.
- ✓ **NotificationMessageAction:** Envía una notificación a un destinatario y permite que el destinatario de la llamada al remitente con un parámetro genérico.
- ✓ **DialogMessage:** Se utiliza para solicitar que un destinatario (por lo general a View) muestra un cuadro de diálogo, y pasa el resultado al llamador (mediante

una devolución de llamada). El destinatario puede elegir cómo mostrar el cuadro de diálogo, ya sea con un cuadro de mensajes estándar, con una medida emergente, etc.

- ✓ **PropertyChangedMessage:** Se utiliza para transmitir que una propiedad cambia en el remitente. Cumple la misma función que el evento PropertyChanged, pero de un modo menos estricto.” [13]

## **CAPÍTULO III**

### **ANÁLISIS DE RENDIMIENTO DE LOS FRAMEWORKS PRIMS Y MVVM LIGHT**

El contenido que tiene este documento de investigación posee información acerca de los frameworks que ayudan a resolver algún tipo de problema con el diseño en las aplicaciones. Se ha estudiado cada uno de ellos para demostrar la forma de construirlos, para las diferentes proyecciones de mantenibilidad, eficiencia y flexibilidad.

En este capítulo se implementará un análisis de rendimiento comparando estadísticamente una aplicación con los framework Prism y MVVM Light; los parámetros a tomarse en cuenta, el desarrollo de los prototipos, la explicación de los puntajes alcanzados y su interpretación finalmente se realiza la comprobación de la hipótesis para determinar el mejor framework en el que se implementara la aplicación final.

Para la comprobación de la hipótesis se utilizara una técnica estadística que permitirá la obtención de resultados de cada uno de los requisitos desarrollados en los prototipos de estudio y expresados en valores numéricos, que permitirá determinar la mejor opción para implementar la aplicación VAMEG SOFT.

### **3.1. Definición de los parámetros a comparar**

Para conocer cuál de los frameworks en estudio es el más eficiente para el desarrollo de la aplicación se han seleccionado los factores que han sido determinados por:

- ✓ **Disponibilidad de Información.-** valora la existencia de información y conocimiento que se puede conseguir de cada framework por medios accesibles, como Internet, libros y manuales.
- ✓ **Número de líneas de código escritas.-** Este parámetro se lo analiza en el momento de desarrollar un requerimiento. Se contarán las líneas de código escritas para cada requerimiento.
- ✓ **Tiempo de desarrollo.-** Es el tiempo que se demora en implementar cada requerimiento.
- ✓ **Tiempo de respuesta.-** Cada prototipo posee un tiempo determinado en el momento de responder a una petición, se verifica la demora al ejecutar un requerimiento de los prototipos.
- ✓ **Uso de CPU.-** Se obtendrá usando el administrador de tareas y me da un número determinado al ejecutar un requerimiento del prototipo seleccionado.
- ✓ **Uso de Memoria RAM.-** representa la cantidad de memoria RAM que consume los requerimientos desarrollados sobre los prototipos.

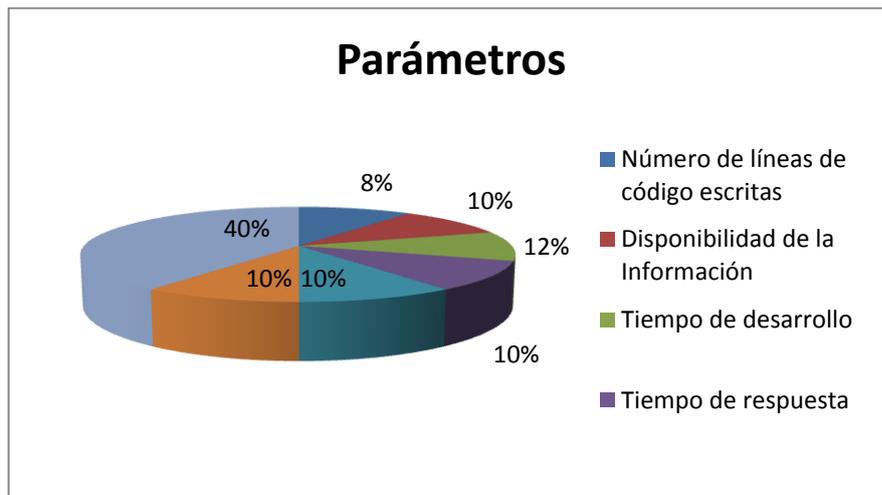
- ✓ **Tiempo de Mantenimiento.**- Se medirá el tiempo que toma desarrollar un nuevo requerimiento y adaptarlo a los prototipos.

La tabla III.I, detalla los parámetros y sus valores a usar dentro de la presente investigación.

**TABLA III. I: Valores de los Parámetros a usar**

Parámetro	Valor Porcentual
Número de líneas de código escritas	8%
Disponibilidad de la Información	10%
Tiempo de desarrollo	12%
Tiempo de respuesta	10%
Uso de CPU	10%
Uso de memoria RAM	10%
Tiempo de Mantenimiento	40%
<b>TOTAL</b>	<b>100%</b>

Para obtener el porcentaje para cada parámetro se debe tener a consideración el ciclo de vida del software y en especial la experiencia que se obtuvo en el estudio de estos nuevos frameworks es por esto que se ha colocado esta valoración. (Figura III. 4).



**FIGURA III. 4: Valores Porcentuales de los parámetros a usar**

Con los parámetros mencionados se evaluarán los frameworks y se obtendrá como resultado el más adecuado en el cual se desarrollara el sistema VAMEG SOFT.

## **3.2. Métodos técnicas y procedimientos**

En la investigación se determinan algunos métodos que son base esencial en el mismo.

A continuación se detallan los elementos:

### **3.2.1. Métodos**

En el presente estudio prevalece el método científico que plantea:

- ✓ Definición del Problema
- ✓ Descripción de la Hipótesis
- ✓ Recopilación de información para comprobar la hipótesis con el uso de experimentos
- ✓ Analiza resultados

### **3.2.2. Técnicas y procedimientos**

En esta etapa hay que tener en cuenta la hipótesis: “La implementación de una aplicación con Prism es el más eficiente en Rendimiento, que una aplicación desarrollada en MVVM Light”.

Para poder comprobar la hipótesis, se especifica la muestra y población que serán objeto de estudio. Se utilizara estadística descriptiva e inferencial para alcanzar este objetivo. [15]

## **Población**

La población se ha escogido de los requerimientos específicos de la aplicación de la empresa Cybertronic que se utilizan en el Módulo de Ventas.

## **Requerimientos**

**TABLA III. II: Requerimientos prototipo**

Código	Requerimiento
REQ3	El sistema debe permitir ingresar, actualizar y eliminar clientes
REQ5	El sistema debe permitir ingresar, actualizar y eliminar productos.
REQ6	El sistema debe permitir ingresar, actualizar y eliminar marca.
REQ7	El sistema debe permitir ingresar, actualizar y eliminar categoría
REQ8	El sistema debe permitir ingresar, actualizar y eliminar subcategoría
REQ9	El sistema debe permitir realizar las búsquedas de clientes, proveedores, productos

## **Muestra**

Para el análisis de rendimiento se procederá al desarrollo de prototipos del Módulo ventas con los frameworks Prism y Mvvm light respectivamente, para esto se ha realizado el análisis de los requisitos de la aplicación para la Empresa Cybertronic

### Toma de mediciones

Una vez desarrollados los prototipos, se procedió a tomar las medidas de los parámetros, a continuación se muestra en la siguiente tabla con los resultados obtenidos para los prototipos del módulo ventas con PRISM y MVVM LIGHT.

TABLA III. III: Toma de mediciones

PRISM						
	LINEAS DE CODIGO	Tiempo de desarrollo(h)	Uso de CPU(%)	Uso de la RAM (MB)	Tiempo de Respuesta(s)	Tiempo Mantenimiento(h)
REQ3	260	24	4.156	58.188	1.5	1.2
REQ5	696	46	8.612	68.455	3	2.25
REQ6	486	12.15	3.028	52.987	1.4	0.15
REQ7	235	9	4.556	57.528	2.3	1.6
REQ8	286	11	3.288	56.020	1.2	0.6
REQ9	340	18	5.292	115.644	2.5	3.2
MVVM LIGHT						
	LINEAS DE CODIGO	Tiempo de desarrollo(h)	Uso de CPU (%)	Uso de la RAM (MB)	Tiempo de Respuesta(s)	Tiempo Mantenimiento(h)
REQ3	290	28.4	5.312	58.352	1.3	1.5
REQ5	776	68.4	4.516	59.972	3.7	2
REQ6	200	16.15	3.228	60.660	1.6	0.25
REQ7	234	12	5.767	56.408	2.8	1.15
REQ8	150	13.15	4.623	75.584	1.4	0.45
REQ9	315	21.15	5.982	213.560	3.4	2.4

### Cálculo de la media poblacional

Para el cálculo de la media se tomaron las medidas de los valores de cada uno de los parámetros considerados dentro del análisis estadístico comparativo.

## PROTOTIPO PRISM

### NÚMERO DE LÍNEAS DE CÓDIGO

#### Media

$$\bar{X} = \frac{X_1 + X_2 + X_3 + \dots + X_n}{N}$$

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{N}$$

$$X = \frac{1903}{6} = 317,166$$

#### Varianza

$$s^2 = \sum_{i=1}^n \frac{(xi)^2 - nx^2}{n}$$

$$s^2 = 31411,22$$

#### Desviación estándar

$$s = \sqrt{s^2}$$

$$s = 177,23$$

### Intervalo de confianza

$$IC = x \pm z (Sx) \quad Sx = \frac{S}{\sqrt{n}}$$

$$Ic = 317,166 \pm 1,96 * 72,35$$

$$IC = 317,166 \pm 141,81$$

$$IC = [175,356; 458,976]$$

### TIEMPO DE DESARROLLO

Media

$$\bar{X} = \frac{X_1 + X_2 + X_3 + \dots + X_n}{N}$$

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{N}$$

$$X = \frac{120,15}{6} = 20,02$$

Varianza

$$s^2 = \frac{\sum_{i=1}^n (xi)^2 - nx^2}{n}$$

$$s^2 = 160,14$$

### Desviación estándar

$$S = \sqrt{S}$$

$$S = 12,65$$

### Intervalo de confianza

$$IC = \bar{x} \pm 2 (S_x) \quad S_x = \frac{S}{\sqrt{n}}$$

$$Ic = 20,02 \pm 1,96 * 5,166$$

$$IC = 20,02 \pm 10,13$$

$$IC = [9,89; 30,15]$$

### USO DEL CPU

Media

$$\bar{X} = \frac{X_1 + X_2 + X_3 + \dots + X_n}{N}$$

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{N}$$

$$X = \frac{28,932}{6} = 4,822$$

### Varianza

$$s^2 = \sum_{i=1}^n \frac{(xi)^2 - nx^2}{n}$$

$$s^2 = 3,445$$

### Desviación estándar

$$s = \sqrt{s^2}$$

$$s = 1,856$$

### Intervalo de confianza

$$IC = x \pm z (Sx) \quad Sx = \frac{s}{\sqrt{n}}$$

$$Ic = 4,822 \pm 1,96 * 0,757$$

$$IC = 4,822 \pm 1,485$$

$$IC = [3,336; 6,307]$$

## USO DE MEMORIA RAM

### Media

$$\bar{X} = \frac{X_1 + X_2 + X_3 + \dots + X_R}{N}$$

$$\bar{X} = \frac{\sum_{i=1}^R X_i}{N}$$

$$X = \frac{408,822}{6} = 68,137$$

### Varianza

$$s^2 = \frac{\sum_{i=1}^n (xi)^2 - nx^2}{n}$$

$$s^2 = 474,148$$

### Desviación estándar

$$S = \sqrt{s}$$

$$S = 21,77$$

### Intervalo de confianza

$$IC = x \pm z (Sx) \quad Sx = \frac{S}{\sqrt{n}}$$

$$Ic = 68,137 \pm 1,96 * 4,66$$

$$IC = 68,137 \pm 9,146$$

$$IC = [58,991; 77,283]$$

## TIEMPO DE RESPUESTA

### Media

$$\bar{X} = \frac{X_1 + X_2 + X_3 + \dots + X_n}{N}$$

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{N}$$

$$X = \frac{11,9}{6} = 1,983$$

### Varianza

$$s^2 = \sum_{i=1}^n \frac{(xi)^2 - nx^2}{n}$$

$$s^2 = 0,432$$

### Desviación estándar

$$s = \sqrt{s^2}$$

$$s = 0,658$$

### Intervalo de confianza

$$IC = x \pm z (Sx) \quad Sx = \frac{s}{\sqrt{n}}$$

$$Ic = 1,983 \pm 1,96 * 0,26$$

$$IC = 1,983 \pm 0,509$$

$$IC = [1,474; 2,492]$$

### TIEMPO DE MANTENIMIENTO

Media

$$\bar{x} = \frac{X_1 + X_2 + X_3 + \dots + X_n}{N}$$

$$\bar{x} = \frac{\sum_{i=1}^n X_i}{N}$$

$$X = \frac{5,45}{6} = 0,91$$

Varianza

$$s^2 = \sum_{i=1}^n \frac{(xi)^2 - nx^2}{n}$$

$$s^2 = 0,377$$

### Desviación estándar

$$s = \sqrt{S}$$

$$s = 0,614$$

### Intervalo de confianza

$$IC = x \pm z (Sx) \quad Sx = \frac{s}{\sqrt{n}}$$

$$Ic = 0,91 \pm 1,96 * 0,25$$

$$IC = 0,91 \pm 0,491$$

$$IC = [0,419; 1,401]$$

### PROTOTIPO MVVM LIGHT

### NÚMERO DE LÍNEAS DE CÓDIGO

#### Media

$$\bar{x} = \frac{X_1 + X_2 + X_3 + \dots + X_n}{N}$$

$$\bar{x} = \frac{\sum_{i=1}^n X_i}{N}$$

$$X = \frac{1965}{6} = 327,5$$

### Varianza

$$s^2 = \sum_{i=1}^n \frac{(xi)^2 - nx^2}{n}$$

$$s^2 = 43203,25$$

### Desviación estándar

$$S = \sqrt{s}$$

$$S = 207,85$$

### Intervalo de confianza

$$IC = x \pm z (Sx) \quad Sx = \frac{S}{\sqrt{n}}$$

$$Ic = 327,5 \pm 1,96 * 84,85$$

$$IC = 327,5 \pm 166.31$$

$$IC = [161,19; 493,81]$$

### TIEMPO DE DESARROLLO

Media

$$\bar{x} = \frac{X_1 + X_2 + X_3 + \dots + X_n}{N}$$

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{N}$$

$$X = \frac{159,25}{6} = 26,54$$

### Varianza

$$s^2 = \sum_{i=1}^n \frac{(x_i)^2 - nx^2}{n}$$

$$s^2 = 380,66$$

### Desviación estándar

$$S = \sqrt{s}$$

$$S = 19,51$$

### Intervalo de confianza

$$IC = x \pm 2 (Sx) \quad Sx = \frac{S}{\sqrt{n}}$$

$$Ic = 26,54 \pm 196 * 7,97$$

$$IC = 26,54 \pm 15,61$$

$$IC = [10,93; 42,15]$$

## USO DEL CPU

### Media

$$\bar{X} = \frac{X_1 + X_2 + X_3 + \dots + X_R}{N}$$

$$\bar{X} = \frac{\sum_{i=1}^R X_i}{N}$$

$$X = \frac{29,421}{6} = 4,070$$

### Varianza

$$s^2 = \sum_{i=1}^n \frac{(xi)^2 - nx^2}{n}$$

$$s^2 = 8,342$$

### Desviación estándar

$$S = \sqrt{s}$$

$$S = 2,889$$

### Intervalo de confianza

$$IC = \bar{x} \pm z (S_x) \quad S_x = \frac{S}{\sqrt{n}}$$

$$Ic = 4,070 \pm 1,96 * 1,179$$

$$IC = 4,070 \pm 2,311$$

$$IC = [1,759; 6,381]$$

### USO DE MEMORIA RAM

Media

$$\bar{X} = \frac{X_1 + X_2 + X_3 + \dots + X_n}{N}$$

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{N}$$

$$X = \frac{524,536}{6} = 87,422$$

Varianza

$$s^2 = \sum_{i=1}^n \frac{(xi)^2 - nx^2}{n}$$

$$s^2 = 3221,378$$

**Desviación estándar**

$$s = \sqrt{S}$$

$$s = 56,757$$

**Intervalo de confianza**

$$IC = x \pm z (Sx) \quad Sx = \frac{s}{\sqrt{n}}$$

$$Ic = 87,422 \pm 1,96 * 23,17$$

$$IC = 87,422 \pm 45,41$$

$$IC = [42,012; 132,832]$$

**TIEMPO DE RESPUESTA**

Media

$$\bar{x} = \frac{x_1 + x_2 + x_3 + \dots + x_n}{N}$$

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{N}$$

$$X = \frac{14,2}{6} = 2,367$$

**Varianza**

$$s^2 = \sum_{i=1}^n \frac{(xi)^2 - nx^2}{n}$$

$$s^2 = 0,95$$

**Desviación estándar**

$$S = \sqrt{S}$$

$$S = 0,97$$

**Intervalo de confianza**

$$IC = x \pm z (Sx) \quad Sx = \frac{s}{\sqrt{n}}$$

$$Ic = 2,367 \pm 1,96 * 0,39$$

$$IC = 2,367 \pm 1,776$$

$$IC = [0,591; 4,143]$$

**TIEMPO DE MANTENIMIENTO**

Media

$$\bar{X} = \frac{X_1 + X_2 + X_3 + \dots + X_n}{N}$$

$$\bar{X} = \frac{\sum_{i=1}^n x_i}{N}$$

$$X = \frac{10,45}{6} = 1,741$$

### Varianza

$$s^2 = \frac{\sum_{i=1}^n (xi)^2 - nx^2}{n}$$

$$s^2 = 2,38$$

### Desviación estándar

$$S = \sqrt{s}$$

$$S = 1,54$$

### Intervalo de confianza

$$IC = x \pm z (Sx) \quad Sx = \frac{s}{\sqrt{n}}$$

$$Ic = 1,741 \pm 1,96 * 0,62$$

$$IC = 1,741 \pm 1,23$$

$$IC = [0,511; 2,971]$$

## APRENDIZAJE

### Disponibilidad de la Información Internet

TABLA III. IV: Disponibilidad de la información

FRAMEWORK	PAGINAS EN ESPAÑOL	LA WEB	PROMEDIO
PRISM	2,580,000 resultados (0,15 segundos)	3,410,000 resultados (0,17 segundos)	2,995,000
MVVM LIGHT	378,040 resultados (0,37 segundos)	150,100 resultados (0,15 segundos)	264070

A continuación se muestra la regla de tres para conseguir el resultado del framework, tomando en cuenta que 3259070 corresponde al máximo valor.

$$\text{Para Prism: } \frac{2995000 \times 100}{3259070} = 92\%$$

$$\text{Para Mvvm Light: } \frac{264070 \times 100}{3259070} = 8\%$$

### 3.3. Análisis Comparativo

En el desarrollo de los prototipos se utilizó los lenguajes de programación Visual Basic, C# lenguajes conocidos y fáciles de aprender; para la interfaz de usuario se la realizó con Windows Presentation Foundation (WPF) y el almacenamiento de datos con Microsoft SQL Server 2008 Express Edition.

Las características del computador Laptop SONY –VAIO en las que se desarrollan los prototipos y las pruebas correspondientes son:

- ✓ 4GB de Memoria RAM.

- ✓ 500 GB de disco duro.
- ✓ Procesador Intel Core i5

De los prototipos desarrollados se obtuvieron medidas para poder tener resultados usando una distribución Chi cuadrado con la cual se determinó las diferencias existentes entre las dos medias muestrales tomando como valor de desconfianza del 5%.

### **3.4. Puntajes Alcanzados**

Los valores de los parámetros se obtuvieron cuando se implementó los requerimientos específicos como muestra; para valorar la existencia de información y conocimiento se utilizó como medio al internet, las líneas de código escritas fueron contabilizadas en el momento de desarrollo de cada requisito del sistema.

Para medir el tiempo de desarrollo y mantenimiento se utilizó el tiempo que determina un cronometro y la herramienta propia del Visual Studio 2010

El uso del CPU y memoria RAM se utilizó la herramienta administrador de tareas de Windows 7 en la que se ejecutó los prototipos y se prueba cada uno de los requisitos.

Para el tiempo de respuesta se utilizó la herramienta de Visual Studio 2010.

Al finalizar el desarrollo de los dos prototipos del módulo de ventas del sistema se obtuvieron los datos anteriormente mencionados.

Se efectúa el cálculo de las medias de cada uno de los parámetros evaluados, así como también sus respectivos intervalos de confianza, la tabla III.V detalla los valores obtenidos

**TABLA III. V: Valores de las Medias de los prototipos**

<b>PROTOTIPO FRAMEWORK PRISM</b>						
	Líneas de código escritas	Tiempo de desarrollo	Uso CPU	Uso Memoria RAM	Tiempo de Respuesta	Tiempo Mantenimiento
<b>Media</b>	317,166	20,02	4,822	68,137	1,983	0,91
<b>Intervalo de Confianza</b>	175,356	9,89	3,336	58,991	1,474	0,419
	458,976	30,15	6,307	77,283	2,492	1,401
<b>PROTOTIPO FRAMEWORK MVVM LIGHT</b>						
	Líneas de código escritas	Tiempo de desarrollo	Uso CPU	Uso Memoria RAM	Tiempo de Respuesta	Tiempo Mantenimiento
<b>Media</b>	327,5	26,54	4,070	87,422	2,367	1,741
<b>Intervalo de Confianza</b>	161,19	10,93	1,759	42,012	0,591	0,511
	493,81	42,15	6,381	132,832	4,143	2,971

El cálculo de mayor relevancia es el Intervalo de Confianza con la que se procede a la comparación. El valor de la Media que sea menor y se encuentre dentro de este intervalo determinará si existe diferencia significativa entre los valores de cada parámetro.

### 3.5 Interpretación

**Internet:** La búsqueda de información disponible sobre conceptos de Prism y Mvvm Light se obtuvo mediante el buscador Web más utilizado; Google. Primeramente se buscó el concepto limitando las paginas solo en español y luego se realizó la búsqueda en la web; tomando como referencia la suma de todos los promedios para la obtención de los resultados.

El máximo valor es de 3259070 el cual corresponde al 100% de la información total de los frameworks, por lo tanto se realizó una regla de tres.

**TABLA III. VI: Análisis Parámetro Disponibilidad de la Información**

	<b>PRISM</b>	<b>MVVM LIGHT</b>
<b>Internet</b>	92%	8%

La mayoría de la información de los frameworks de estudio en este trabajo se encuentran en ingles siendo así una gran limitante para el desarrollador porque se debe dedicar más tiempo para aprender hasta traducir y entender.

Como se observa en la **Figura III. 5**, Prism supera a Mvvm Light en cuanto a la disponibilidad de la información en el Internet para el desarrollo del aprendizaje.

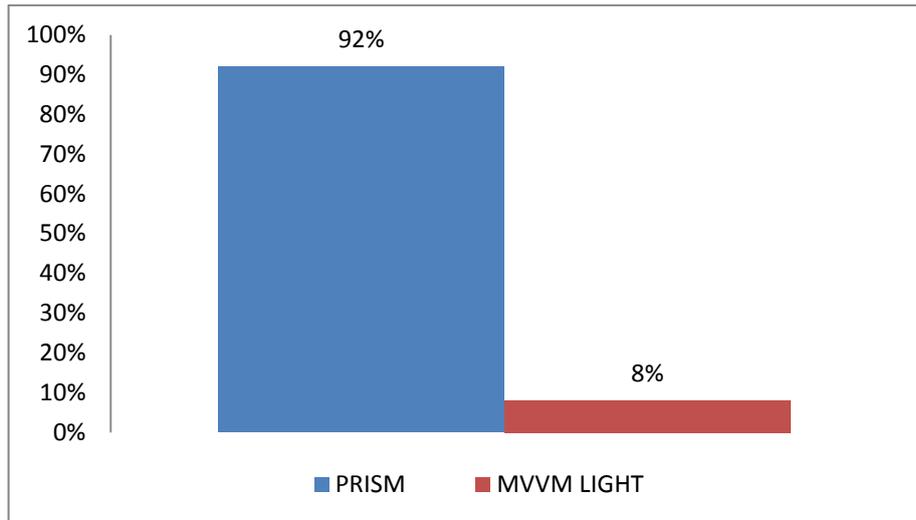


FIGURA III. 5: Análisis del Parámetro de Disponibilidad de Información

La Figura III.6 indica la comparación de los valores para el parámetro número de líneas de código escritas en los prototipos del módulo de ventas.

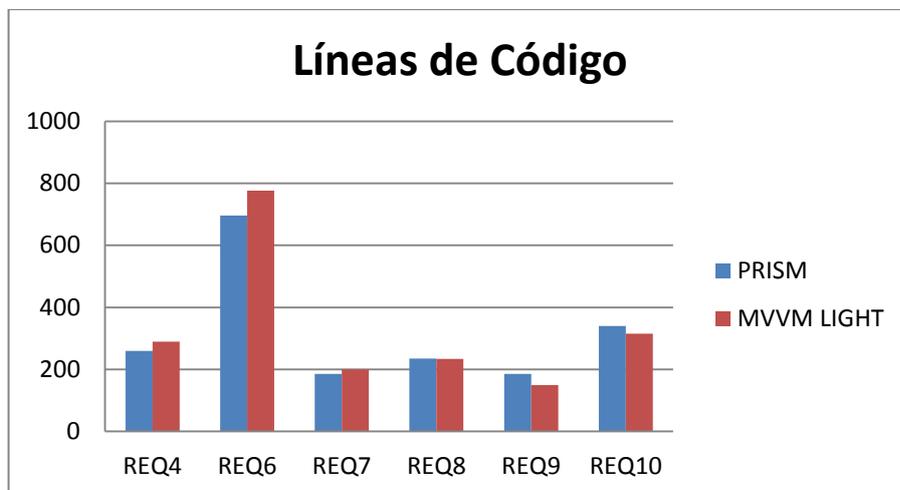
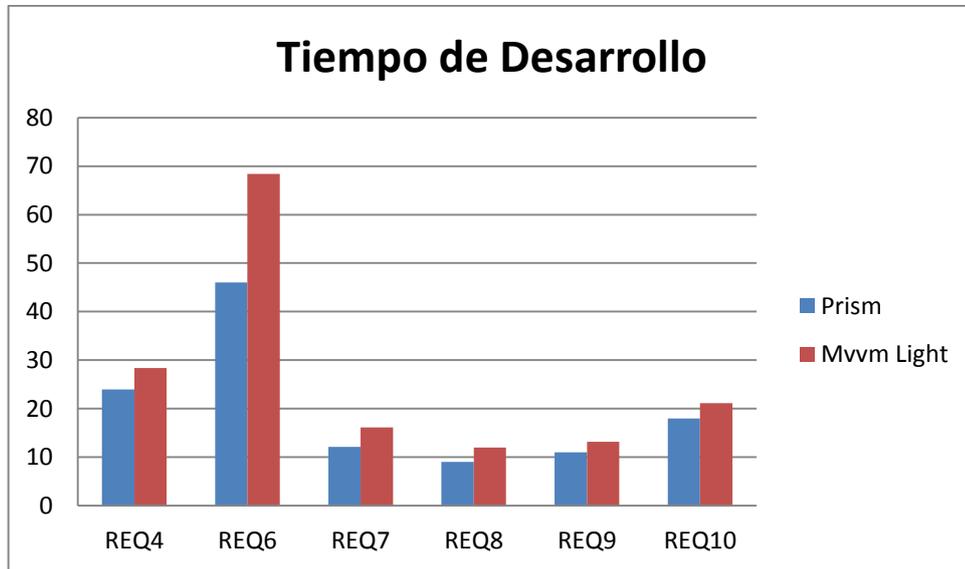


FIGURA III. 6: Comparación del parámetro Líneas de Código

Esta figura indica que no existe una diferencia significativa en el número de líneas de código entre los dos prototipos porque los valores hallados (medias de líneas de código) Prism 317,166 y Mvvm Light 327,5 se encuentran dentro del rango de intervalo de confianza [161,19 ; 493,81].

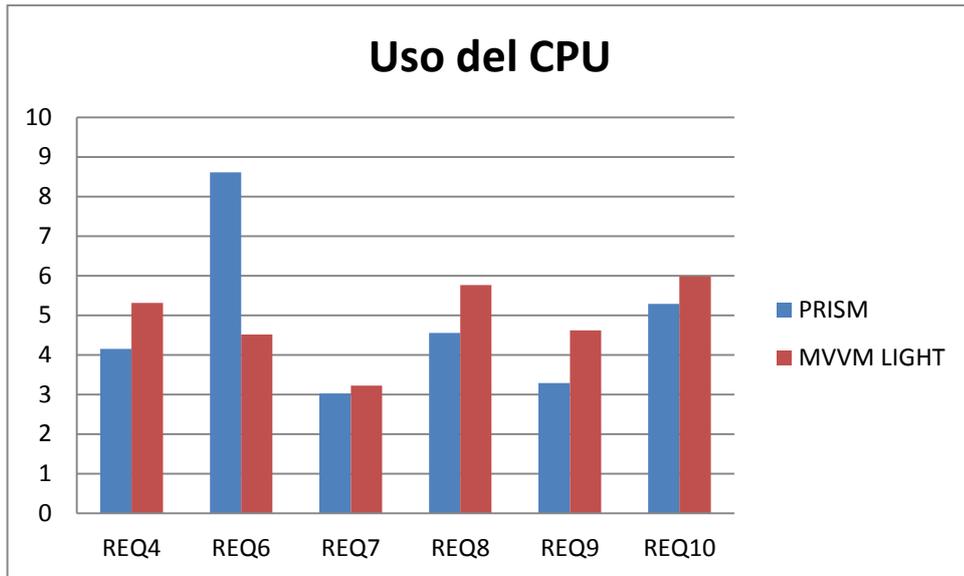
La Figura III.7 indica la comparación de los valores para el parámetro Tiempo de Desarrollo de los prototipos del módulo de ventas del sistema VAMEG SOFT.



**FIGURA III. 7: Comparación del parámetro Tiempo de Desarrollo**

Estadísticamente este Parámetro no tiene una diferencia significativa porque las medias para Prism: 20,02 y Mvvm Light: 26,54 se encuentran del intervalo de confianza hallados [9,89; 30,15].

La Figura III.8 muestra la comparación entre el uso del CPU de los prototipos del módulo de ventas del sistema VAMEG SOFT.



**FIGURA III. 8: Comparación del parámetro Uso del CPU**

Se puede observar que no hay diferencia significativa en el uso CPU al ejecutar los requerimientos sobre los prototipos del módulo de ventas del sistema VAMEGSOFT porque la medias obtenidas Prism 4,822 y Mvvm Light 4,070 se encuentran dentro del intervalo de confianza [1,759; 6,381].

La Figura III.9 muestra la comparación entre el uso de la Memoria RAM de los prototipos del módulo de ventas del sistema VAMEG SOFT.

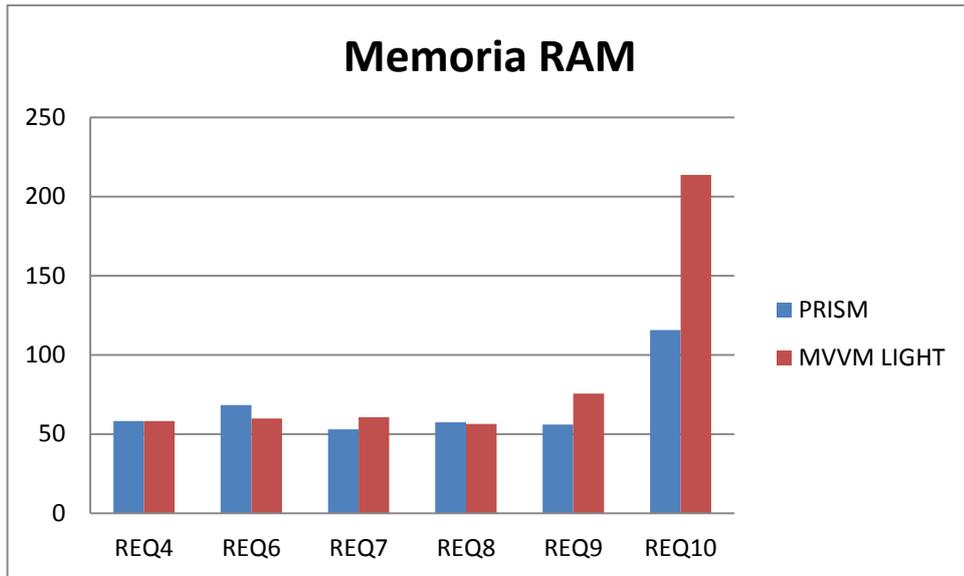


FIGURA III. 9: Comparación del parámetro Uso de la Memoria RAM

Estadísticamente no existe diferencia la media para Prism está en 68,137 y Mvvm Light 87,422 y se encuentra dentro del intervalo de confianza [42,012; 132,832].

La Figura III.10 muestra la comparación entre el Tiempo de Respuesta de los prototipos del módulo de ventas del sistema VAMEGSOFT.

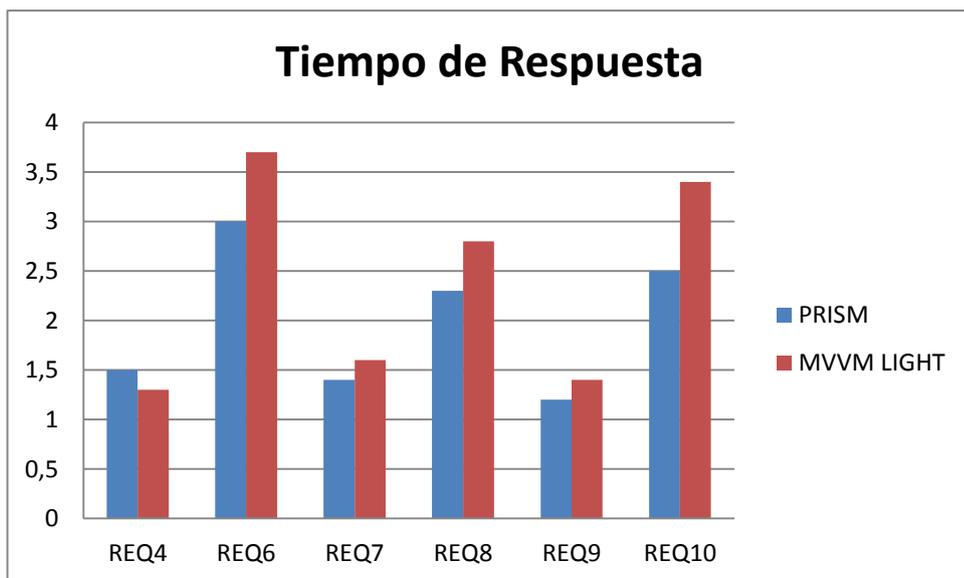


FIGURA III. 10: Comparación del parámetro Tiempo de Respuesta

No existe ninguna diferencia significativa porque para el prototipo Prism 1,983 y de Mvvm Light 2,367 se encuentran dentro del intervalo de confianza [1,474; 2,492]

La Figura III.11 muestra la comparación entre el Tiempo de Mantenimiento de los prototipos del módulo de ventas del sistema VAMEG SOFT.

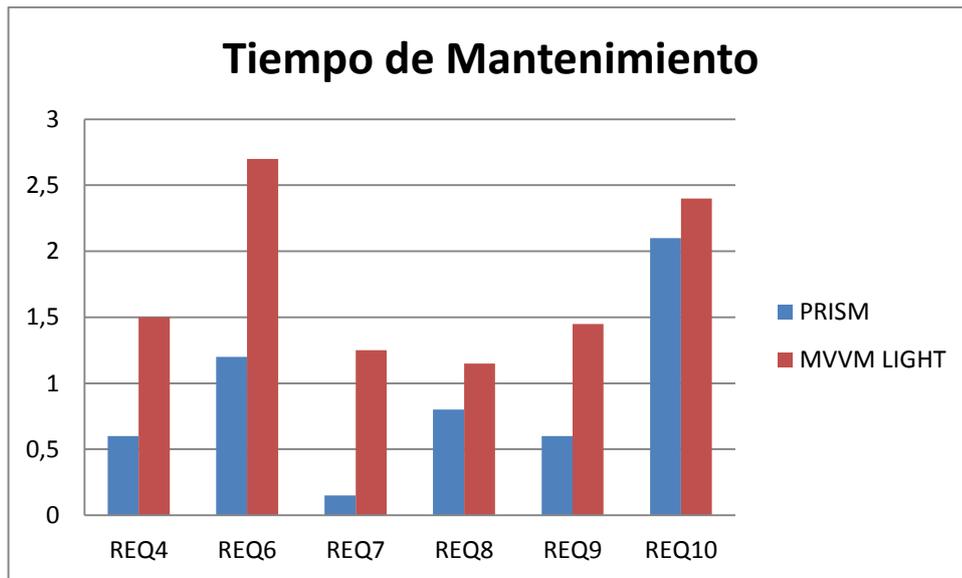


FIGURA III. 11: Comparación del parámetro Tiempo de Mantenimiento

En este parámetro existe una diferencia significativa porque los valores para Prism 0,91 y Mvvm Lght 1,741 que representan el tiempo que toma en implementar un nuevo requisito y adaptarlo al prototipo del módulo de ventas y no se encuentra dentro del intervalo de confianza [0,419; 1,401].

### 3.6 Resultados del Análisis

El análisis finaliza usando estadística descriptiva e inferencial para escoger el mejor framework para desarrollar la aplicación final del Sistema VAMEG SOFT; en el caso de disponibilidad de información, líneas de código, tiempo de desarrollo, uso de CPU, uso

de memoria RAM y tiempo de respuesta determino que no existió una diferencia significativa por el contrario para el tiempo de mantenimiento demuestra que el framework Prism es el más adecuado para la realización del sistema VAMEG SOFT.

**TABLA III. VII: Valores para la comprobación análisis**

	<b>Prototipo Prism</b>		<b>Prototipo Mvvm Light</b>	
	<b>Media</b>	<b>Peso</b>	<b>Media</b>	<b>Peso</b>
<b>Disponibilidad de Información</b>		0,100		0,008
<b>Líneas de Código</b>	317,166	0,080	327,5	0,077
<b>Tiempo de Desarrollo</b>	20,02	0,120	26,54	0,086
<b>Uso CPU</b>	4,822	0,086	4,070	0,100
<b>Uso RAM</b>	68,137	0,100	87,422	0,089
<b>Tiempo de Respuesta</b>	1,983	0,100	2,367	0,28
<b>Tiempo de Mantenimiento</b>	0,91	0,400	1,741	0,055
<b>Totales</b>		0,98		0,69
<b>TOTAL (%)</b>		<b>98</b>		<b>69</b>

La figura III.12 muestra la comparación entre los puntajes alcanzados por los prototipos Prism y Mvvm Light.

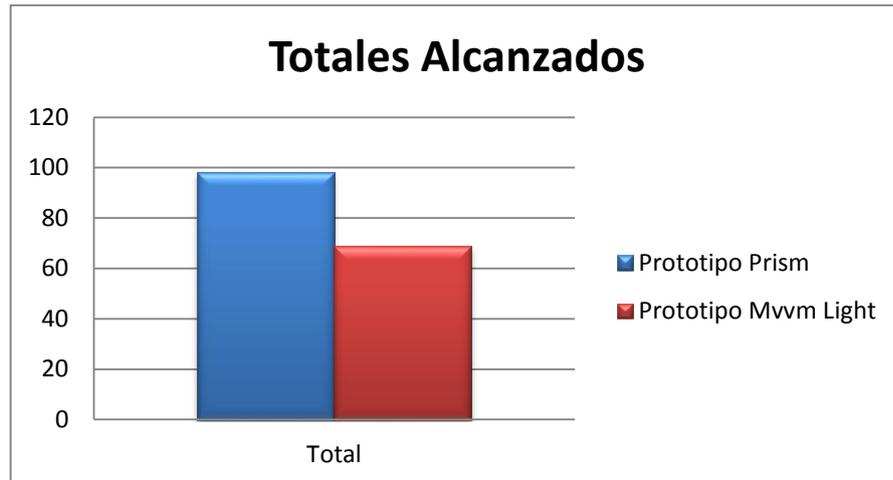


FIGURA III. 12: Totales Alcanzados en la Comparación

En conclusión se tiene como resultado que el mejor framework en cuanto a rendimiento para el desarrollo de la aplicación para la empresa Cybertronic, es el Framework Prism con un noventa y ocho por ciento sobre el sesenta y nueve por ciento del framework Mvvm Light.

### 3.7. Comprobación de la Hipótesis

Para poder determinar cuál es el mejor framework para el desarrollo de la aplicación se aplicó un modelo estadístico Chi Cuadrado, la tabla III.VIII, muestra los datos Observados

Se toma de referencia las medias obtenidas para cada parámetro.

#### ✓ PLATEAMIENTO DE LA HIPOTESIS

**H<sub>0</sub>**: La implementación de una aplicación con Prism no es más eficiente en Rendimiento, que una aplicación desarrollada en MVVM Light.

**H<sub>1</sub>**: La implementación de una aplicación con Prism es el más eficiente en Rendimiento, que una aplicación desarrollada en MVVM Light

✓ **NIVEL DE SIGNIFICANCIA**

Se Debe determinar el nivel de significancia, para el caso del presente análisis se utilizará un nivel de significación estadística de  $\alpha = 0,05$ .

$$1 - 0,05 = 0,95$$

✓ **DESCRIPCIÓN DE LA MUESTRA**

La muestra se ha tomado de dos frameworks Prism Y Mvvm Light, porque tienen las mismas características para ser comparadas en su rendimiento; además porque son frameworks nuevos que facilitan a los desarrolladores al momento de implementar una aplicación.

✓ **ESTADÍSTICO UTILIZADO**

Se ha utilizado para la comprobación de este trabajo de investigación el modelo Chi cuadrado porque es uno de los primeros que se los realiza y no se tiene investigaciones previas para su comparación; se utilizó la fórmula a una sola cola extendiéndose a la derecha, con los grados de libertad y su nivel de significancia.

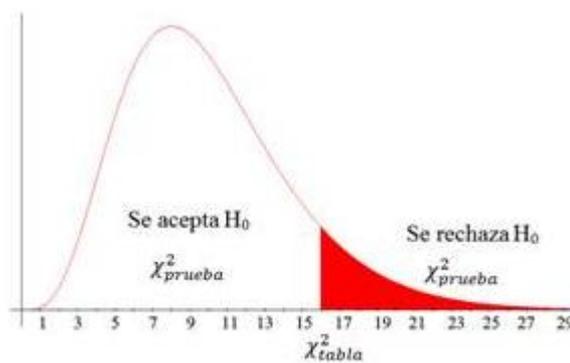


FIGURA III. 13: Chi Cuadrado

✓ CÁLCULOS

TABLA III. VIII: Datos Observados Chi Cuadrado

	Prototipo PRISM	Prototipo Mvvm Light	Total
Tiempo de Desarrollo	20.02	26.54	46.56
Uso CPU	5	4	8.892
Uso RAM	68	87	155.559
Tiempo de Respuesta	2	2	4.35
Tiempo de Mantenimiento	0.91	2	2.651
<b>Total</b>	<b>95.872</b>	<b>122.14</b>	<b>218.012</b>

TABLA III. IX: Valores esperados

r \ k	Prototipo PRISM	Prototipo Mvvm Light	Total
Tiempo de Desarrollo	20.47502119	26.08497881	46.56
Uso CPU	4	5	8.892
Uso RAM	68	87	155.559
Tiempo de Respuesta	2	2	4.35
Tiempo de Mantenimiento	1.165792122	1	2.651
<b>Total</b>	<b>95.872</b>	<b>122.14</b>	<b>218.012</b>

TABLA III. X: Valores de contingencia

Valores Observados	Valores Esperados	Vo - Ve	(Vo - Ve) <sup>2</sup>	(Vo - Ve) <sup>2</sup> /Ve
20.02	20.47502119	-0.45502119	0.207044283	0.010112042
5	4	0.822	0.675684	0.168921
68	68	0.137	0.018769	0.000276015
2	2	-0.017	0.000289	0.0001445
0.91	1.165792122	-0.255792122	0.06542961	0.056124594
26.54	26.08497881	0.45502119	0.207044283	0.007937299
4	5	-0.93	0.8649	0.17298
87	87	0.422	0.178084	0.002046943
2	2	0.367	0.134689	0.0673445
2	1	0.741	0.549081	0.549081
<b>TOTAL</b>				<b>1.034967893</b>

### Grados de libertad

La determinación de los grados de libertad (**gl**), está en función del número de filas (**r**) y el número de columnas (**k**), con la siguiente expresión:

$$gl = (r - 1) * (k - 1)$$

Entonces:

$$gl = (5 - 1) * (2 - 1)$$

$$gl = 4 * 1$$

$$gl = 4$$

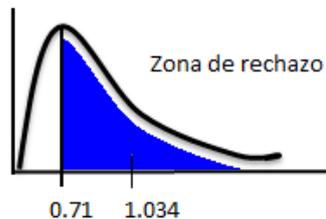


FIGURA III. 14: Comprobación de Hipótesis

✓ **CONCLUSIÓN**

Entonces se tiene  $x^2$  calculado  $>$   $x^2$  tabla, lo que significa que  $x^2$  calculado esta en la zona de rechazo de la  $H_0$  por tanto se rechaza la hipótesis nula y se acepta la de la investigación, siendo la siguiente:

**“La implementación de una aplicación con Prism es el más eficiente en Rendimiento, que una aplicación desarrollada en MVVM Light”**

## **CAPÍTULO IV**

### **DESARROLLO DEL SISTEMA VAMEG SOFT PARA LA EMPRESA CYBERTRONIC**

#### **4.1. Gestión del Proyecto**

En este capítulo se describe el desarrollo de la metodología XP (Programación Extrema), la misma que se aplicó de acuerdo a los requerimientos necesarios para la implementación del sistema VAMEG SOFT, esta metodología contiene cuatro fases, cada una de ellas con sus respectivas actividades que se detallan a continuación.

La metodología XP contiene cuatro fases:

Fase I: Planificación

Fase II: Diseño

Fase III: Codificación

Fase IV: Pruebas

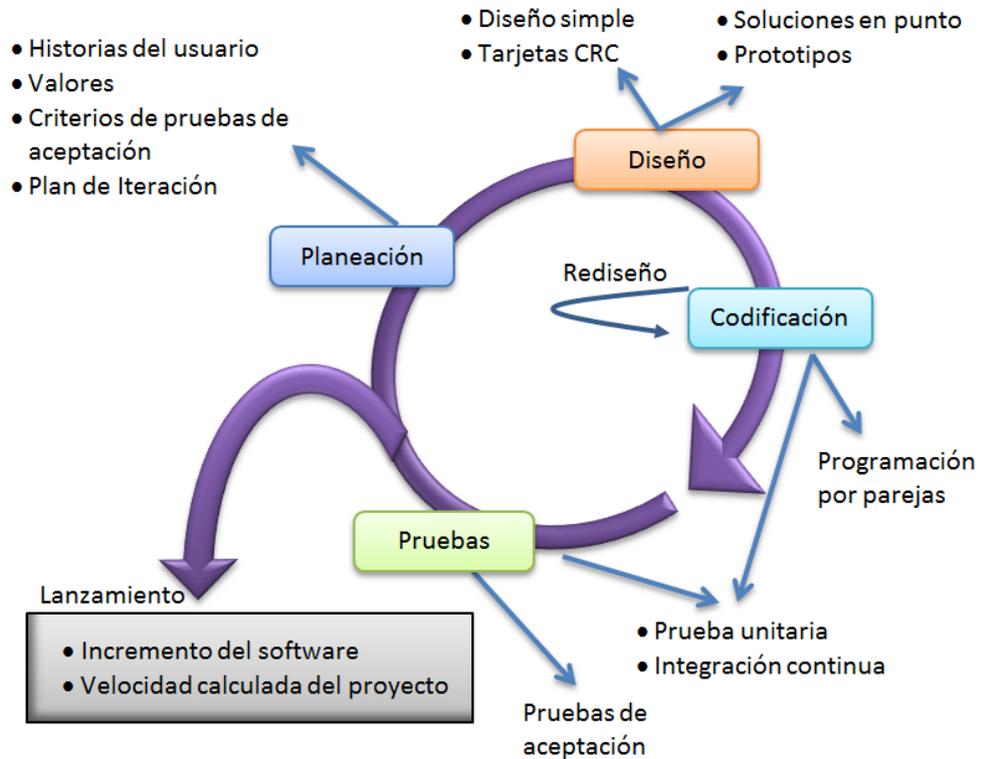


FIGURA IV. 15: Fases de la Metodología XP

## 4.2. Desarrollo de la Metodología XP

### 4.2.1. Fase I: Planificación

En esta fase se desarrollaran todas las actividades que permitirán conocer, comprender y entender el proceso de compras, ventas de la empresa Cybertronic, así como también los requerimientos e historias de usuario del mismo, la planificación inicial e iteraciones necesarias para cumplir con los objetivos planteados.

### ✓ **Descripción del Sistema**

El sistema VAMEG SOFT permite las compras y ventas que posee la empresa Cybertronic, es decir ingresa clientes y proveedores de los productos que expende la empresa generando reportes de las cuentas por pagar y cobrar.

El sistema se desarrolló en Visual Studio 2010 con el framework Prism, para el almacenamiento de la información se utilizó el gestor de base de datos SQL Server.

### ✓ **Definición del flujo del proceso del sistema**

Para conocer a fondo el sistema es necesario definir con el cliente el proceso del negocio, para obtener los requerimientos en este caso se tuvo una entrevista con el gerente general de la empresa Cybertronic quien conoce muy bien lo que desea del sistema.

A continuación se presenta el flujo del proceso desde que ingresa un usuario al sistema la manera de facturar compras, ventas y el inventario de los productos que se ofertan.

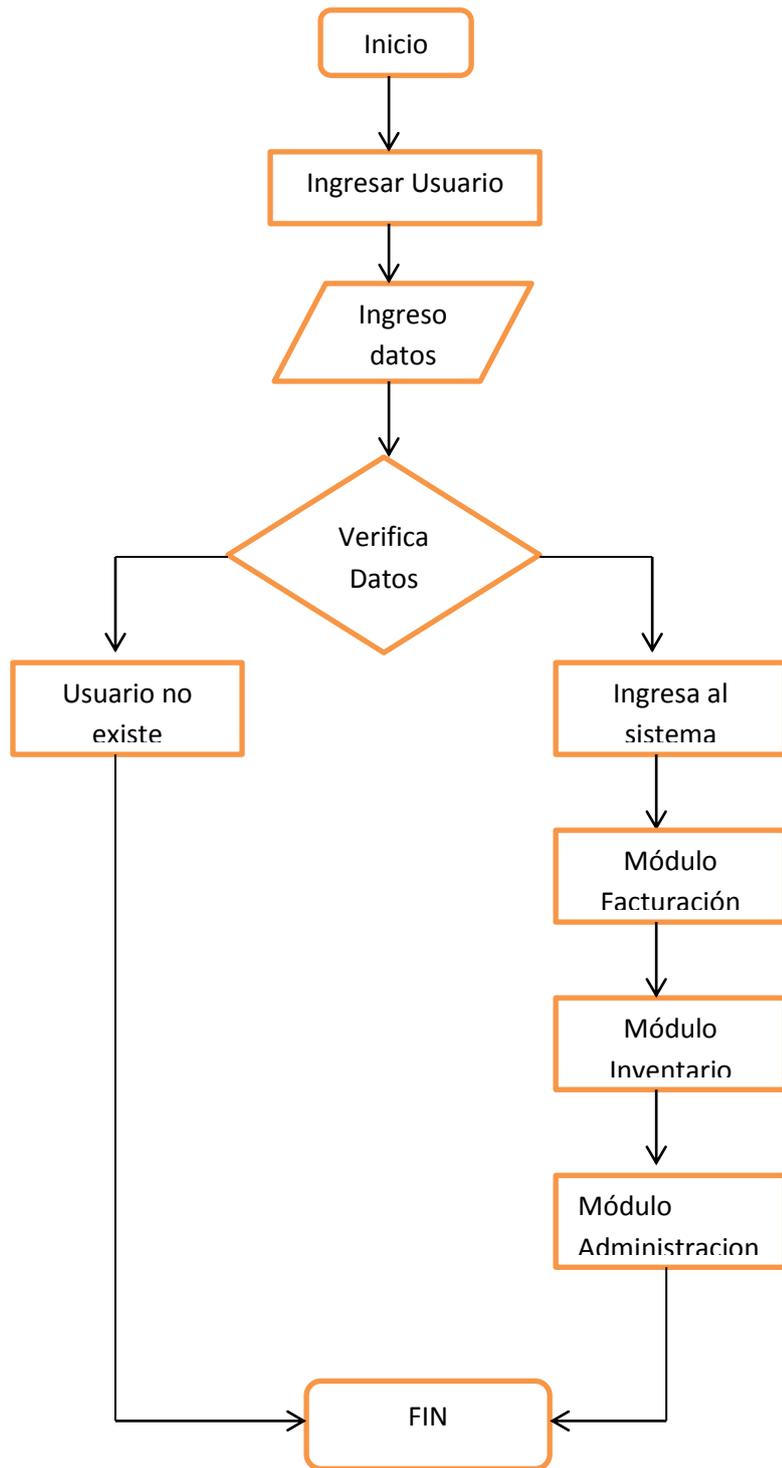


FIGURA IV. 16: Flujo de Proceso del Sistema

✓ **Especificación de Requerimientos**

Comprende la descripción completa del sistema VAMEG SOFT, como la perspectiva del producto, requerimientos no funcionales, incluye un conjunto de historias de usuario que describen cada uno de los requerimientos funcionales, además de un plan de iteraciones a seguir para el diseño e implementación del sistema.

**Perspectiva del Producto**

El sistema VAMEG SOFT será utilizado para gestionar las compras y ventas de la empresa Cybertronic.

**Requerimientos funcionales**

1. El sistema debe permitir ingresar al módulo Administración
2. El sistema debe permitir ingresar el módulo de Facturación(Compras, Ventas)
3. El sistema debe permitir ingresar el Módulo de Inventario.
4. El sistema debe permitir ingresar, actualizar y eliminar clientes.
5. El sistema debe permitir ingresar, actualizar y eliminar proveedores.
6. El sistema debe permitir ingresar, actualizar y eliminar productos.
7. El sistema debe permitir ingresar, actualizar y eliminar marca.
8. El sistema debe permitir ingresar, actualizar y eliminar categoría.
9. El sistema debe permitir ingresar, actualizar y eliminar subcategoría.
10. El sistema debe permitir realizar las búsquedas de clientes, proveedores, productos.
11. El sistema debe permitir obtener reportes de las compras, ventas, productos y proveedores.

12. El sistema debe permitir obtener información sobre las cuentas por cobrar y pagar.

### **Requerimientos No funcionales**

✓ **Disponibilidad**

Este sistema estará disponible las 24 horas para cualquier acción que desee realizar el usuario.

✓ **Fiabilidad**

El sistema es confiable porque ha sido sometido a continuas pruebas y validaciones para medir su grado de eficiencia.

✓ **Mantenibilidad**

Mantenimiento de la base de datos.

#### **Definición de usuarios**

Tomando en cuenta cada requerimiento, se definen los usuarios que interactuarán directamente con el sistema VAMEG SOFT, así como los roles de cada uno de ellos.



FIGURA IV. 17: Usuarios del Sistema VAMEG SOFT

### Definición de roles

Teniendo en cuenta las necesidades de cada uno de los usuarios antes mencionados se definió y asignó los roles como se muestra a continuación:

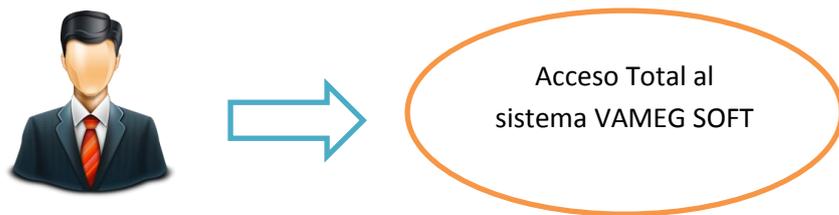


FIGURA IV. 18: Rol de Usuario Administrador

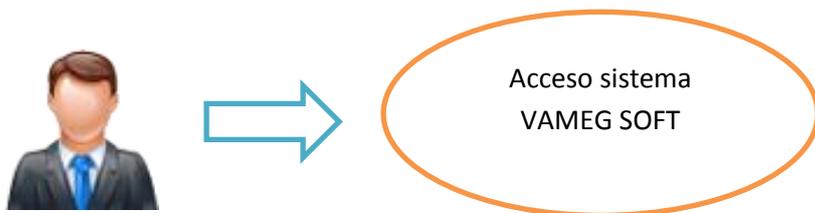
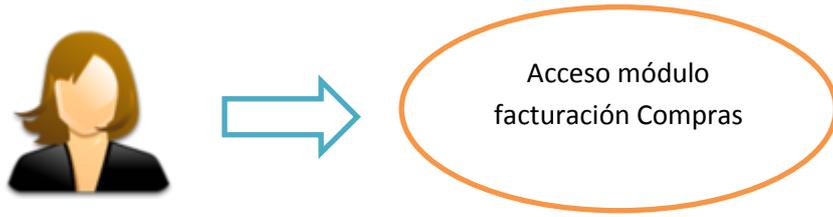
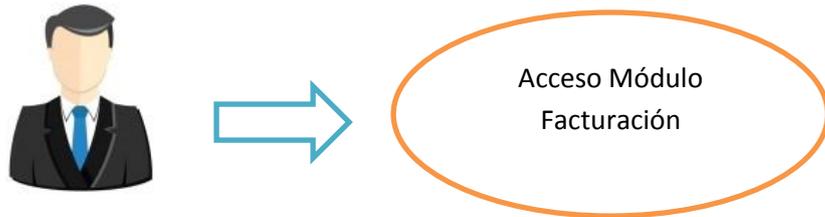


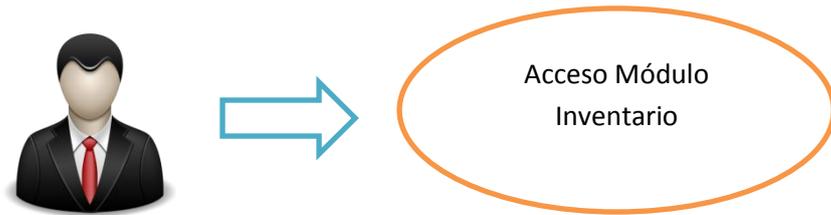
FIGURA IV. 19: Rol de Usuario Gerente



**FIGURA IV. 20: Rol Usuario Secretaria**



**FIGURA IV. 21: Rol de Usuario Contador**



**FIGURA IV. 22: Rol de Usuario Bodeguero**

## **Historias de Usuario**

Una historia de usuario es una representación de un requerimiento de software escrito en una o dos fases utilizando el lenguaje común del usuario, a continuación se detalla cada una de estas.

TABLA IV. XI: Historia de Usuario Ingreso al módulo Administración

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 1	<b>Usuario:</b> Administrador
<b>Nombre Historia:</b> El sistema permitirá el ingreso al módulo Administración	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto
<b>Programador Responsable:</b> Verónica Arguello	
<b>Descripción:</b> El usuario ingresara al Sistema VAMEG SOFT y tendrá control total del mismo.	
<b>Observaciones:</b>	

TABLA IV. XII: Historia de Usuario Ingreso al módulo Facturación

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 2	<b>Usuario:</b> Contador
<b>Nombre Historia:</b> El sistema debe permitir ingresar el módulo de Facturación(Compras, Ventas)	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto
<b>Programador Responsable:</b> María Fernanda Girón	
<b>Descripción:</b> El usuario ingresara al módulo de Facturación	
<b>Observaciones:</b>	

TABLA IV. XIII: Historia de Usuario Ingreso al módulo de Inventario

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 3	<b>Usuario:</b> Bodeguero
<b>Nombre Historia:</b> El sistema debe permitir ingresar al módulo de Inventario	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto
<b>Programador Responsable:</b> Verónica Arguello	
<b>Descripción:</b> El usuario ingresará al módulo de inventario	
<b>Observaciones:</b>	

TABLA IV. XIV: Historia de Usuario Ingresar, actualizar y eliminar clientes

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 4	<b>Usuario:</b> Secretaria
<b>Nombre Historia:</b> El sistema debe permitir ingresar, actualizar y eliminar clientes	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto
<b>Programador Responsable:</b> María Fernanda Girón	
<b>Descripción:</b> El usuario ingresara, modificara y eliminara los datos de un cliente	
<b>Observaciones:</b>	

TABLA IV. XV: Historia de Usuario Ingresar, actualizar y eliminar proveedores

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 5	<b>Usuario:</b> Secretaria
<b>Nombre Historia:</b> El sistema debe permitir ingresar, actualizar y eliminar proveedores	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto
<b>Programador Responsable:</b> Verónica Arguello	
<b>Descripción:</b> El usuario ingresara, modificara y eliminara los datos de un proveedor	
<b>Observaciones:</b>	

TABLA IV. XVI: Historia de Usuario Ingresar, actualizar y eliminar productos

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 6	<b>Usuario:</b> Secretaria, Bodeguero
<b>Nombre Historia:</b> El sistema debe permitir ingresar, actualizar y eliminar productos	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto
<b>Programador Responsable:</b> María Fernanda Girón	
<b>Descripción:</b> El usuario ingresara, modificara y eliminara los datos de un producto	
<b>Observaciones:</b>	

TABLA IV. XVII: Historia de Usuario Ingresar, actualizar y eliminar marcas

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 7	<b>Usuario:</b> Secretaria, Bodeguero
<b>Nombre Historia:</b> El sistema debe permitir ingresar, actualizar y eliminar marcas	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto
<b>Programador Responsable:</b> Verónica Arguello	
<b>Descripción:</b> El usuario ingresara, modificara y eliminara los datos de las marcas	
<b>Observaciones:</b>	

TABLA IV. XVIII: Historia de Usuario Ingresar, actualizar y eliminar categorías

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 8	<b>Usuario:</b> Secretaria, Bodeguero
<b>Nombre Historia:</b> El sistema debe permitir ingresar, actualizar y eliminar categorías	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto
<b>Programador Responsable:</b> María Fernanda Girón	
<b>Descripción:</b> El usuario ingresara, modificara y eliminara los datos de una categoría	
<b>Observaciones:</b>	

TABLA IV. XIX: Historia de Usuario Ingresar, actualizar y eliminar subcategorías

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 9	<b>Usuario:</b> Secretaria, Bodeguero
<b>Nombre Historia:</b> El sistema debe permitir ingresar, actualizar y eliminar subcategorías	<b>Iteración Asignada:</b> 1
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto
<b>Programador Responsable:</b> Verónica Arguello	
<b>Descripción:</b> El usuario ingresara, modificara y eliminara los datos de subcategorías	
<b>Observaciones:</b>	

TABLA IV. XX: Historia de Usuario realizar búsquedas de clientes, proveedores, productos

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 10	<b>Usuario:</b> Gerente
<b>Nombre Historia:</b> El sistema debe permitir realizar búsquedas de clientes, proveedores, productos.	<b>Iteración Asignada:</b> 2
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto
<b>Programador Responsable:</b> María Fernanda Girón	
<b>Descripción:</b> El usuario realizará búsquedas de clientes, proveedores, productos	
<b>Observaciones:</b>	

TABLA IV. XXI: Historia de Usuario Obtener reportes de las compras, ventas, clientes, proveedores

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 11	<b>Usuario:</b> Gerente, Contador
<b>Nombre Historia:</b> El sistema debe permitir obtener reportes de las compras, ventas, clientes, proveedores.	<b>Iteración Asignada:</b> 3
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto
<b>Programador Responsable:</b> Verónica Arguello	
<b>Descripción:</b> El usuario obtendrá reportes de compras, ventas, clientes, proveedores, productos.	
<b>Observaciones:</b>	

TABLA IV. XXII: Historia de Usuario Obtener reportes de las cuentas por cobrar y pagar

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 12	<b>Usuario:</b> Contador
<b>Nombre Historia:</b> El sistema debe permitir obtener reportes de las cuentas por cobrar y pagar	<b>Iteración Asignada:</b> 3
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto
<b>Programador Responsable:</b> María Fernanda Girón	
<b>Descripción:</b> El usuario obtendrá reportes de las cuentas por pagar y cobrar	
<b>Observaciones:</b>	

### **Planificación Inicial**

Se desarrolló un plan de actividades previo al desarrollo de la aplicación, en el mismo se detallan las actividades, el responsable y el tiempo en que se desarrolla cada una de estas.

### **Plan de Iteraciones**

El desarrollo se divide en tres iteraciones, en cada una de ellas se eligen las historias de usuarios a desarrollar, con un tiempo de implementación de 40 puntos estimados. Un punto tiene la duración de 4 horas.

Todas las iteraciones corresponden a un módulo funcional y entregable de la aplicación, es así que aproximadamente cada cuatro semanas la aplicación de estudio constará con una funcionalidad adicional.

✓ **Iteración 1**

En esta iteración se presenta el ingreso, actualización y eliminación de un cliente, proveedor, producto, marca, categoría, subcategoría.

✓ **Iteración 2**

En esta iteración se presenta las búsquedas de clientes, proveedores, productos.

✓ **Iteración 3**

En esta iteración se presenta reportes de compras, ventas, clientes, proveedores, productos.

## 4.2.2. Fase II: Diseño del software

### Diseño de Base de datos

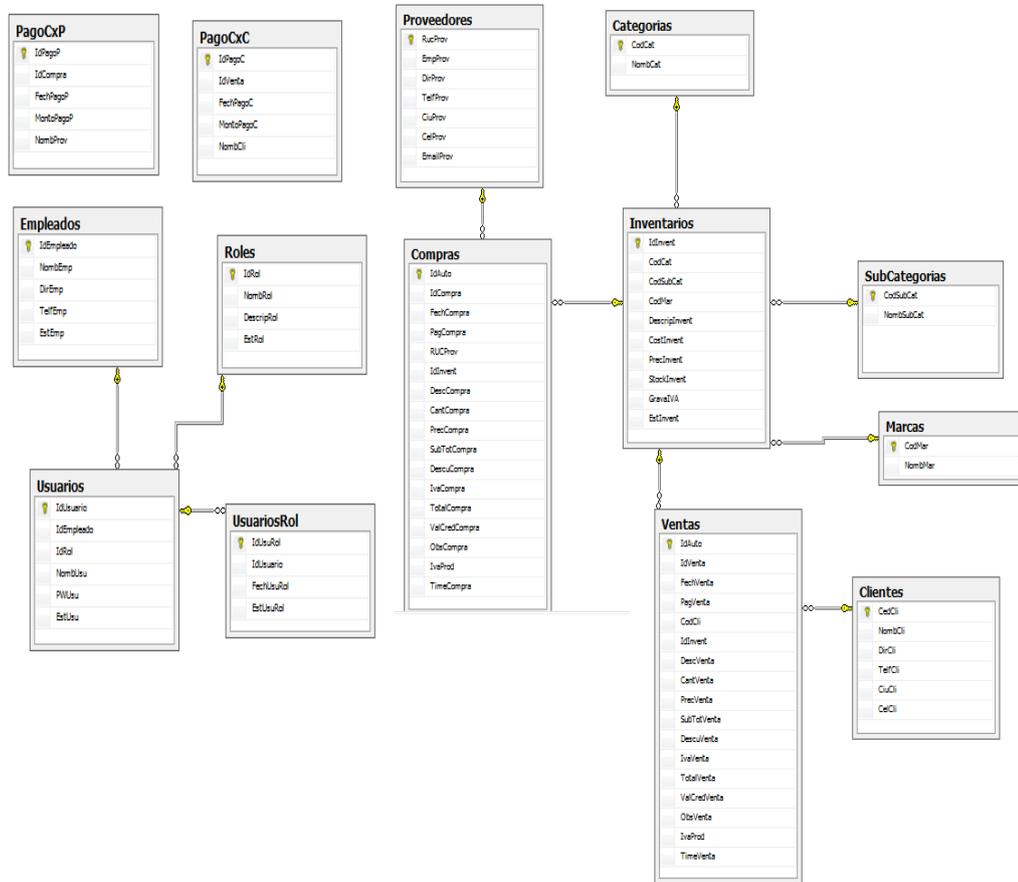


FIGURA IV. 23: Diseño de Base de Datos

## Diseño de interfaces

La metodología XP sugiere que hay que conseguir diseños simples y sencillos, hay que procurar hacerlo todo lo menos complicado posible para conseguir un diseño fácil de entender que conste de menos tiempo y esfuerzo en desarrollo, para lo cual se propone los siguientes prototipos.

INGRESO AL SISTEMA

Usuario:

Contraseña:

Aceptar

Este prototipo muestra una interfaz de usuario para el ingreso al sistema. Tiene un encabezado azul con el título "INGRESO AL SISTEMA". Debajo, hay dos campos de entrada: uno para el "Usuario" y otro para la "Contraseña". Debajo de los campos, hay un botón rectangular con el texto "Aceptar".

FIGURA IV. 24: Ingreso al Sistema

BIENVENIDOS AL SISTEMA

Ventas

Compras

Inventario

Administración

Gestionar Reportes

Clientes

Facturación

Pagos Cuentas Por

Este prototipo muestra una interfaz de usuario para el menú del sistema. Tiene un encabezado azul con el título "BIENVENIDOS AL SISTEMA". A la izquierda, hay una lista vertical de botones: "Ventas", "Compras", "Inventario" y "Administración". A la derecha, hay un panel con dos pestañas: "Gestionar" y "Reportes". Debajo de las pestañas, hay un menú desplegable que muestra tres opciones: "Clientes", "Facturación" y "Pagos Cuentas Por".

FIGURA IV. 25: Menú del Sistema

**MOVIMIENTO DATOS CLIENTE**

Primero      Anterior      Siguiete      Ultimo

Cédula:

Nombres:

Dirección:

Ciudad:

Teléfono:

**FIGURA IV. 26: Movimiento de Datos del Cliente**

**BUSCAR CLIENTES**

Cédula

Nombres

Cédula	Nombres

**FIGURA IV. 27: Buscar Datos del Cliente**





**MOVIMIENTO DATOS CATEGORÍA**

Primer	Anterior	Siguiete	Ultimo	
Código:	<input type="text"/>	Buscar		
Categoría:	<input type="text"/>			
Limpiar/Nuev	Modificar	Eliminar	Guarda	Salir

**FIGURA IV. 32: Movimiento Datos Categoría**

**MOVIMIENTO DATOS SUBCATEGORÍA**

Primer	Anterior	Siguiete	Ultimo	
Código:	<input type="text"/>	Buscar Subcategoría		
Subcategoría:	<input type="text"/>			
Limpiar/Nuev	Modificar	Eliminar	Guarda	Salir

**FIGURA IV. 33: Movimiento Datos Subcategoría**

**MOVIMIENTO DATOS MARCA**

Primer	Anterior	Siguiete	Ultimo	
Código:	<input type="text"/>	Buscar Marca		
Marca:	<input type="text"/>			
Limpiar/Nuev	Modificar	Eliminar	Guarda	Salir

**FIGURA IV. 34: Movimiento Datos Marca**

MOVIMIENTO DATOS PRODUCTO

Primero Anterior Siguiete Ultimo

Código Producto:  Buscar Producto

Categoría:  Asignar Categoría

Subcategoría:  Asignar Subcategoría

Marca:  Asignar Marca

Descripción:

Precio Costo:   Graba IVA

PVP:   Activo

Stock:

Limpiar/Nuevo Modificar Eliminar Guardar Salir

FIGURA IV. 35: Movimiento Datos Producto

MOVIMIENTO DATOS USUARIOS

Primero Anterior Siguiete Ultimo

Usuario:  Buscar Usuario

Contraseña:

Estado:  Activo

Limpiar/Nuev Modificar Eliminar Guardar Salir

FIGURA IV. 36: Movimiento Datos Usuario

## Diccionario de Datos

Una buena práctica de ingeniería es generar un diccionario de datos ya que es fundamental a la hora de conocer su estructura por técnicos ajenos al proyecto.

A continuación se muestra el diccionario de datos de todas las tablas creadas para el desarrollo del sistema VAMEG SOFT.

**TABLA IV. XXIII: Diccionario de Datos**

TABLA	COLUMNA	TIPO DE DATO	PERMITE NULLS	ES AUTONUMERICO	CLAVES FORANEAS
Categorias	CodCat	Nchar	NO	NO	NULL
Categorias	NombCat	Text	NO	NO	NULL
Cientes	CedCli	Nchar	NO	NO	NULL
Cientes	NombCli	Text	NO	NO	NULL
Cientes	DirCli	Text	NO	NO	NULL
Cientes	TelfCli	Text	NO	NO	NULL
Cientes	CiuCli	Text	NO	NO	NULL
Cientes	CelCli	Text	NO	NO	NULL
Compras	IdAuto	Int	NO	SI	NULL
Compras	IdCompra	Nchar	NO	NO	NULL
Compras	FechCompra	Date	NO	NO	NULL
Compras	PagCompra	Nchar	NO	NO	NULL
Compras	RUCProv	Char	NO	NO	FK_Compras_Proveedores
Compras	IdInvent	Nchar	NO	NO	FK_Compras_Inventarios
Compras	DescCompra	Text	NO	NO	NULL
Compras	CantCompra	Int	NO	NO	NULL
Compras	PrecCompra	Decimal	NO	NO	NULL
Compras	SubTotCompra	Decimal	NO	NO	NULL
Compras	DescuCompra	Decimal	NO	NO	NULL
Compras	IvaCompra	Decimal	NO	NO	NULL
Compras	TotalCompra	Decimal	NO	NO	NULL
Compras	ValCredCompra	Decimal	SI	NO	NULL
Compras	ObsCompra	Text	SI	NO	NULL
Compras	IvaProd	Int	NO	NO	NULL
Compras	TimeCompra	Int	NO	NO	NULL
Empleados	IdEmpleado	Nchar	NO	NO	NULL
Empleados	NombEmp	Text	NO	NO	NULL
Empleados	DirEmp	Text	NO	NO	NULL
Empleados	TelfEmp	Text	NO	NO	NULL
Empleados	EstEmp	Int	NO	NO	NULL
Inventarios	IdInvent	Nchar	NO	NO	NULL
Inventarios	CodCat	Nchar	NO	NO	FK_Inventarios_Categorias
Inventarios	CodSubCat	Nchar	NO	NO	FK_Inventarios_SubCategorias
Inventarios	CodMar	Nchar	NO	NO	FK_Inventarios_Marcas
Inventarios	DescripInvent	Text	NO	NO	NULL
Inventarios	CostInvent	Decimal	NO	NO	NULL
Inventarios	PreInvent	Decimal	NO	NO	NULL
Inventarios	StockInvent	Int	NO	NO	NULL
Inventarios	GravaIVA	Int	NO	NO	NULL
Inventarios	EstInvent	Int	NO	NO	NULL
Marcas	CodMar	Nchar	NO	NO	NULL

Marcas	NombMar	Text	NO	NO	NULL
PagoCxC	IdPagoC	Int	NO	SI	NULL
PagoCxC	IdVenta	Nchar	NO	NO	NULL
PagoCxC	FechPagoC	Date	NO	NO	NULL
PagoCxC	MontoPagoC	Decimal	NO	NO	NULL
PagoCxC	NombCli	Text	SI	NO	NULL
PagoCxP	IdPagoP	Int	NO	SI	NULL
PagoCxP	IdCompra	Nchar	NO	NO	NULL
PagoCxP	FechPagoP	Date	NO	NO	NULL
PagoCxP	MontoPagoP	Decimal	NO	NO	NULL
PagoCxP	NombProv	Text	SI	NO	NULL
Proveedores	RucProv	Char	NO	NO	NULL
Proveedores	EmpProv	Text	NO	NO	NULL
Proveedores	DirProv	Text	NO	NO	NULL
Proveedores	TelfProv	Text	NO	NO	NULL
Proveedores	CiuProv	Text	NO	NO	NULL
Proveedores	CelProv	Text	NO	NO	NULL
Proveedores	EmailProv	Text	NO	NO	NULL
Roles	IdRol	Int	NO	SI	NULL
Roles	NombRol	Text	NO	NO	NULL
Roles	DescripRol	Text	NO	NO	NULL
Roles	EstRol	Int	SI	NO	NULL
SubCategorias	CodSubCat	Nchar	NO	NO	NULL
SubCategorias	NombSubCat	Text	NO	NO	NULL
Usuarios	IdUsuario	Int	NO	SI	NULL
Usuarios	IdEmpleado	Nchar	NO	NO	FK_Usuarios_Empleados
Usuarios	IdRol	Int	NO	NO	FK_Usuarios_Roles
Usuarios	NombUsu	Text	NO	NO	NULL
Usuarios	PWUsu	Text	NO	NO	NULL
Usuarios	EstUsu	Int	NO	NO	NULL
UsuariosRol	IdUsuRol	Int	NO	SI	NULL
UsuariosRol	IdUsuario	Int	NO	NO	FK_UsuariosRol_Usuarios
UsuariosRol	FechUsuRol	Date	NO	NO	NULL
UsuariosRol	EstUsuRol	Int	NO	NO	NULL
Ventas	IdAuto	Int	NO	SI	NULL
Ventas	IdVenta	Nchar	NO	NO	NULL
Ventas	FechVenta	Date	NO	NO	NULL
Ventas	PagVenta	Nchar	NO	NO	NULL
Ventas	CodCli	Nchar	NO	NO	FK_Ventas_Clientes
Ventas	IdInvent	Nchar	NO	NO	FK_Ventas_Inventarios
Ventas	DescVenta	Text	NO	NO	NULL
Ventas	CantVenta	Int	NO	NO	NULL
Ventas	PrecVenta	Decimal	NO	NO	NULL
Ventas	SubTotVenta	Decimal	NO	NO	NULL
Ventas	DescuVenta	Decimal	NO	NO	NULL

Ventas	IvaVenta	Decimal	NO	NO	NULL
Ventas	TotalVenta	Decimal	NO	NO	NULL
Ventas	ValCredVenta	Decimal	SI	NO	NULL
Ventas	ObsVenta	Text	SI	NO	NULL
Ventas	IvaProd	Int	NO	NO	NULL
Ventas	TimeVenta	Int	NO	NO	NULL

### 4.2.3. Fase III: Codificación

Para empezar con el desarrollo de la aplicación, es necesario tener una idea clara de la estructura y cómo van a interactuar los usuarios con la base de datos, por lo cual se ha diseñado el siguiente diagrama permitiendo de esta manera enfocarse en la mejor solución.

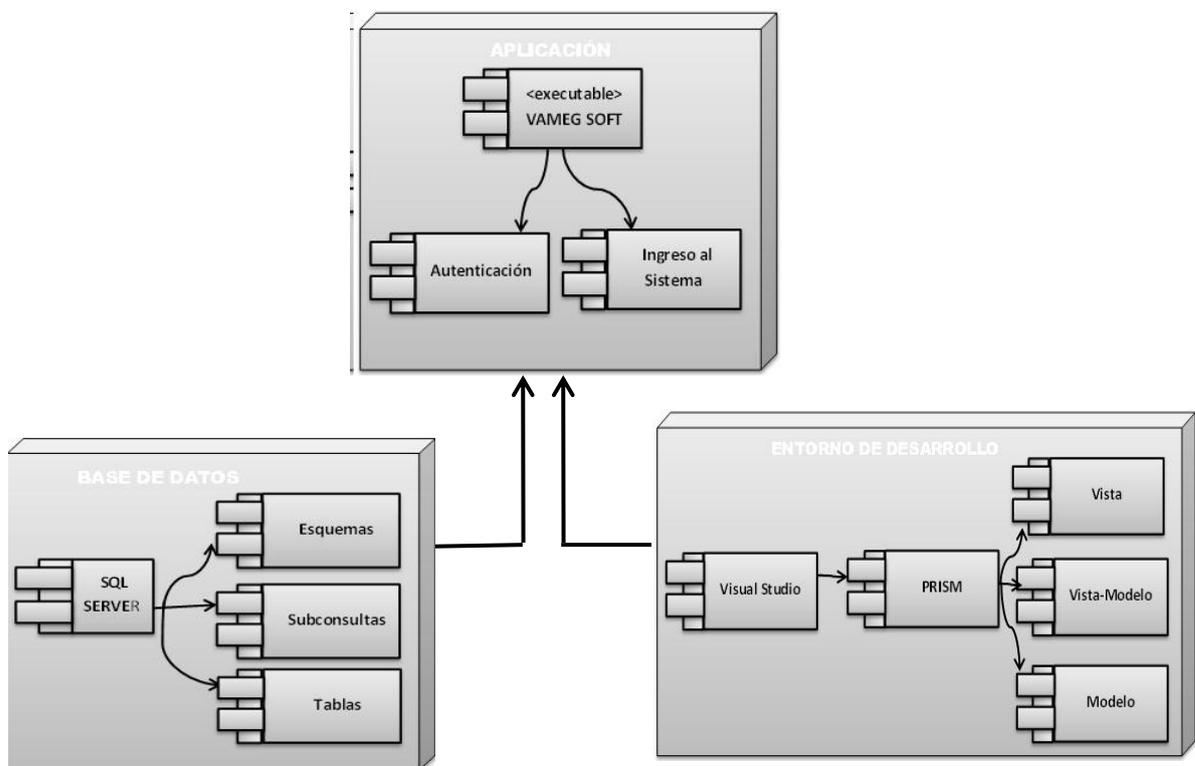


FIGURA IV. 37: Diagrama de Bloques y Despliegue del Sistema

El sistema VAMEG SOFT se desarrolla en Microsoft Visual Studio, que es un entorno de desarrollo integrado para sistemas operativos Windows con lenguaje Visual Basic que utiliza el framework Prism y cuyas interfaces se realizaron en WPF, pues cuenta con una gran gama de componentes que hacen que la experiencia del usuario al utilizar la aplicación sea agradable, fácil de entender. Además se basa en el patrón MVVM

(Modelo Vista –Modelo Vista) que es una arquitectura flexible y que permite independencia.

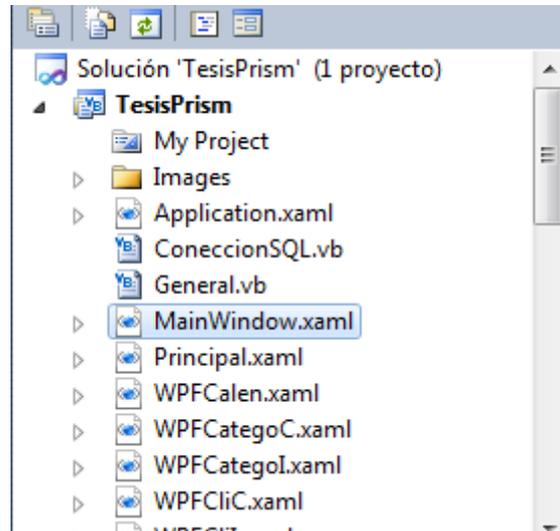


FIGURA IV. 38: Codificación

#### 4.2.4. Fase IV: Pruebas de Funcionamiento

Previa a la entrega del sistema VAMEG SOFT se realizó las pruebas para verificar el correcto funcionamiento del mismo, ingresando los datos necesarios para su buen funcionamiento, y lo más importante que cumpla con los requerimientos establecidos al inicio del desarrollo de la aplicación.

A continuación se muestran las pruebas realizadas para cada requerimiento específico.

TABLA IV. XXIV: Ingreso, modificación y eliminación de clientes

<b>1. Ingreso, Modificación y Eliminación de clientes.</b>	
Descripción	El administrador y gerente una vez que ingresa al sistema podrá ingresar modificar y eliminar un cliente.
Condición de Ejecución	El usuario debe estar registrado en el sistema
Entrada	<ul style="list-style-type: none"><li>- El usuario debe ingresar el nombre de usuario y contraseña.</li><li>- En el menú principal debe elegir a que módulo va a ingresar.</li><li>- Presionar el botón Limpiar/Nuevo para ingresar un nuevo Cliente.</li><li>- Presionar el botón Modificar para poder modificar los datos del cliente.</li><li>- Presionar el botón eliminar si desea eliminar por completo el cliente de la base de datos.</li></ul>
Resultado Esperado	Luego de haber ingresado, modificado y eliminado los datos de clientes y a la vez mostrar los mensajes que todas las acciones se realizaron con éxito.
Evaluación de la prueba	<input type="checkbox"/> Mala <input type="checkbox"/> Aceptable <input checked="" type="checkbox"/> Satisfactoria

TABLA IV. XXV: Ingreso, modificación y eliminación de proveedores

<b>2. Ingreso, Modificación y Eliminación de Proveedores</b>	
Descripción	El administrador y gerente una vez que ingresa al sistema podrá ingresar modificar y eliminar proveedores.
Condición de Ejecución	El usuario debe estar registrado en el sistema
Entrada	<ul style="list-style-type: none"><li>- El usuario debe ingresar el nombre de usuario y contraseña.</li><li>- En el menú principal debe elegir a que módulo va a ingresar.</li><li>- Presionar el botón Limpiar/Nuevo para ingresar un nuevo proveedor.</li><li>- Presionar el botón Modificar para poder modificar los datos del proveedor.</li><li>- Presionar el botón eliminar si desea eliminar por completo el proveedor de la base de datos.</li></ul>
Resultado Esperado	Luego de haber ingresado, modificado y eliminado los datos de proveedores y a la vez mostrar los mensajes que todas las acciones se realizaron con éxito.
Evaluación de la prueba	<input type="checkbox"/> Mala <input type="checkbox"/> Aceptable <input checked="" type="checkbox"/> Satisfactoria

TABLA IV. XXVI: Ingreso, modificación y eliminación de productos

<b>3. Ingreso, Modificación y Eliminación de Productos.</b>	
Descripción	El administrador, gerente y bodeguero una vez que ingresa al sistema podrá ingresar modificar y eliminar productos.
Condición de Ejecución	El usuario debe estar registrado en el sistema
Entrada	<ul style="list-style-type: none"><li>- El usuario debe ingresar el nombre de usuario y contraseña.</li><li>- En el menú principal debe elegir a que módulo va a ingresar.</li><li>- Presionar el botón Limpiar/Nuevo para ingresar un nuevo producto.</li><li>- Presionar el botón Modificar para poder modificar los datos del producto.</li><li>- Presionar el botón eliminar si desea eliminar por completo el producto de la base de datos.</li></ul>
Resultado Esperado	Luego de haber ingresado, modificado y eliminado los datos de productos y a la vez mostrar los mensajes que todas las acciones se realizaron con éxito.
Evaluación de la prueba	<input type="checkbox"/> Mala <input type="checkbox"/> Aceptable <input checked="" type="checkbox"/> Satisfactoria

TABLA IV. XXVII: Ingreso, modificación y eliminación de categoría

<b>4. Ingreso, Modificación y Eliminación de Categoría</b>	
Descripción	El administrador y gerente una vez que ingresa al sistema podrá ingresar modificar y eliminar la categoría a la que pertenece el producto.
Condición de Ejecución	El usuario debe estar registrado en el sistema
Entrada	<ul style="list-style-type: none"><li>- El usuario debe ingresar el nombre de usuario y contraseña.</li><li>- En el menú principal debe elegir a que módulo va a ingresar.</li><li>- Presionar el botón Limpiar/Nuevo para ingresar una nueva categoría para el producto.</li><li>- Presionar el botón Modificar para poder modificar los datos de la categoría.</li><li>- Presionar el botón eliminar si desea eliminar por completo la categoría de la base de datos.</li></ul>
Resultado Esperado	Luego de haber ingresado, modificado y eliminado los datos de la categoría y a la vez mostrar los mensajes que todas las acciones se realizaron con éxito.
Evaluación de la prueba	<input type="checkbox"/> Mala <input type="checkbox"/> Aceptable <input checked="" type="checkbox"/> Satisfactoria

TABLA IV. XXVIII: Ingreso, modificación y eliminación de subcategoría

<b>5. Ingreso, Modificación y Eliminación de Subcategoría</b>	
Descripción	El administrador y gerente una vez que ingresa al sistema podrá ingresar modificar y eliminar Subcategoría.
Condición de Ejecución	El usuario debe estar registrado en el sistema
Entrada	<ul style="list-style-type: none"><li>- El usuario debe ingresar el nombre de usuario y contraseña.</li><li>- En el menú principal debe elegir a que módulo va a ingresar.</li><li>- Presionar el botón Limpiar/Nuevo para ingresar una nueva subcategoría.</li><li>- Presionar el botón Modificar para poder modificar los datos de la subcategoría.</li><li>- Presionar el botón eliminar si desea eliminar por completo la subcategoría de la base de datos.</li></ul>
Resultado Esperado	Luego de haber ingresado, modificado y eliminado los datos de la subcategoría y a la vez mostrar los mensajes que todas las acciones se realizaron con éxito.
Evaluación de la prueba	<input type="checkbox"/> Mala <input type="checkbox"/> Aceptable <input checked="" type="checkbox"/> Satisfactoria

TABLA IV. XXIX: Ingreso, modificación y eliminación de marcas

<b>6. Ingreso, Modificación y Eliminación de Marca</b>	
Descripción	El administrador y gerente una vez que ingresa al sistema podrá ingresar modificar y eliminar marca.
Condición de Ejecución	El usuario debe estar registrado en el sistema
Entrada	<ul style="list-style-type: none"><li>- El usuario debe ingresar el nombre de usuario y contraseña.</li><li>- En el menú principal debe elegir a que módulo va a ingresar.</li><li>- Presionar el botón Limpiar/Nuevo para ingresar una nueva marca.</li><li>- Presionar el botón Modificar para poder modificar los datos de la marca.</li><li>- Presionar el botón eliminar si desea eliminar por completo la marca de la base de datos.</li></ul>
Resultado Esperado	Luego de haber ingresado, modificado y eliminado los datos de la marca y a la vez mostrar los mensajes que todas las acciones se realizaron con éxito.
Evaluación de la prueba	<input type="checkbox"/> Mala <input type="checkbox"/> Aceptable <input checked="" type="checkbox"/> Satisfactoria

TABLA IV. XXX: Búsqueda de clientes, proveedores, productos

<b>7. búsqueda de clientes, proveedores, productos</b>	
Descripción	<p>El administrador, gerente una vez que ingresa al sistema podrá realizar las búsquedas de clientes proveedores y productos.</p> <p>La secretaria y contadora una vez que ingresa al sistema podrá realizar las búsquedas de clientes proveedores.</p> <p>El bodeguero una vez que ingresa al sistema podrá realizar las búsquedas de productos.</p>
Condición de Ejecución	El usuario debe estar registrado en el sistema
Entrada	<ul style="list-style-type: none"><li>- El usuario debe ingresar el nombre de usuario y contraseña.</li><li>- En el menú principal debe elegir a que módulo va a ingresar.</li><li>- Presionar el botón buscar ya sea para clientes, proveedores o productos (buscara por cedula y nombre).</li></ul>
Resultado Esperado	Luego de haber realizados las respectivas búsqueda y a la vez mostrar los mensajes que todas las acciones se realizaron con éxito.
Evaluación de la prueba	<input type="checkbox"/> Mala <input type="checkbox"/> Aceptable <input checked="" type="checkbox"/> Satisfactoria

TABLA IV. XXXI: Reportes de las compras, ventas, productos, clientes y proveedores

<b>8. Reportes de las compras, ventas, productos, clientes y proveedores</b>	
Descripción	<p>El administrador, gerente una vez que ingresa al sistema podrá obtener resultados de las compras, ventas, productos, clientes y proveedores.</p> <p>La secretaria y contadora una vez que ingresa al sistema podrá obtener resultados de las ventas, compras, clientes productos y proveedores.</p> <p>El bodeguero una vez que ingresa al sistema podrá obtener resultados de los productos.</p>
Condición de Ejecución	El usuario debe estar registrado en el sistema
Entrada	<ul style="list-style-type: none"> <li>- El usuario debe ingresar el nombre de usuario y contraseña.</li> <li>- En el menú principal debe elegir a que módulo va a ingresar.</li> <li>- Presionar el botón Reportes y seleccionar que reporte desea realizar</li> </ul>
Resultado Esperado	Luego de haber obtenidos los resultados y a la vez mostrar los mensajes que todas las acciones se realizaron con éxito.
Evaluación de la prueba	<input type="checkbox"/> Mala <input type="checkbox"/> Aceptable <input checked="" type="checkbox"/> Satisfactoria

TABLA IV. XXXII: Información sobre las cuentas por cobrar y pagar

<b>9. Información sobre las cuentas por cobrar y pagar</b>	
Descripción	<p>El administrador, gerente una vez que ingresa al sistema podrá obtener información de las cuentas por cobrar y pagar</p> <p>La secretaria y contadora una vez que ingresa al sistema podrá obtener información de las cuentas por cobrar y pagar</p>
Condición de Ejecución	El usuario debe estar registrado en el sistema
Entrada	<ul style="list-style-type: none"><li>- El usuario debe ingresar el nombre de usuario y contraseña.</li><li>- En el menú principal debe elegir a que módulo va a ingresar.</li><li>- Presionar el botón pagos ya sea por cobrar o pagar</li></ul>
Resultado Esperado	Luego de haber obtenido la información y los resultados requeridos y a la vez mostrar los mensajes que todas las acciones se realizaron con éxito.
Evaluación de la prueba	<input type="checkbox"/> Mala <input type="checkbox"/> Aceptable <input checked="" type="checkbox"/> Satisfactoria

## CONCLUSIONES

1. Se ha estudiado sobre los frameworks Prism y Mvvm Light, sus ventajas y componentes.
2. Realizando un análisis se pudo determinar cómo parámetros: la disponibilidad de información, líneas de código escritas, tiempo de respuesta, uso del CPU, uso de la memoria RAM y tiempo de mantenimiento, los mismos que contribuyeron grandemente para determinar el framework más eficiente.
3. Se desarrolló prototipos, en el cual se escogió del módulo de facturación el proceso de ventas para realizar el análisis de rendimiento.
4. En el estudio que se realizó entre los frameworks Prism y Mvvm light no existió diferencias significativas en la mayoría de los parámetros tales como: número de líneas de código escritas, tiempo de desarrollo, uso de CPU, uso de memoria RAM, tiempo de respuesta; aunque se obtuvo una diferencia en el tiempo de mantenimiento que ayuda a determinar que el framework Prism es el más eficiente desarrollando el sistema VAMEG SOFT.
5. El desarrollo de un framework PRISM brinda múltiples ventajas como modularidad, calidad, extensibilidad además de que hacen un sistema flexible, crean un punto de partida para la reorganización del código y brindan un vocabulario de diseño de alto nivel con fácil manejo.
6. Se demostró la hipótesis propuesta indicando que Prism es más eficiente que Mvvm Light.

## RECOMENDACIONES

1. Se recomienda hacer uso del framework Prism para el desarrollo de aplicaciones que vayan a modificar o incrementar sus funcionalidades en el tiempo.
2. Se recomienda determinar los parámetros de comparación en base a las características determinantes y al contexto en el que será aplicado.
3. Realizar un caso de prueba por cada requisito para verificar que se cumpla con lo establecido.
4. Se recomienda que el personal de Cybertronic utilice la documentación del sistema como referencia para el uso y mantenimiento del sistema.
5. Para iniciar un análisis de rendimiento se debe seleccionar cuidadosamente los frameworks involucrados en el mismo, porque deben tener características semejantes para compararlas sin ningún inconveniente.

## RESÚMEN

Se analizó el rendimiento entre los frameworks Prism y Mvvm Light en la empresa Cybertronic (Quito); con los resultados se desarrolló el sistema VAMEG SOFT.

Los métodos que se utilizaron fueron: el científico para el planteamiento del problema, la recopilación de la información, la descripción de la hipótesis comprobándose la misma con el método experimental; para el rendimiento se implementaron prototipos con los frameworks Prism y Mvvm Light.

Con estadística descriptiva Chi Cuadrado se determinaron varios parámetros, teniendo una diferencia significativa en el tiempo de mantenimiento pues se observó que en Prism alcanzo un 98 % y en Mvvm Light un 69% , concluyéndose que el framework Prism es el más eficiente, fácil de configurar, extensible y existe mayor cantidad de información en la web, por lo que se desarrolló el sistema VAMEG SOFT en la plataforma Visual Studio, con el lenguaje de programación Visual Basic, utilizando una metodología ágil XP quedando el sistema funcionando correctamente.

## SUMMARY

Performance between Prism and MVVM light frameworks was analyzed at Cybertronic Company( Quito); with the results the VAMEG SOFT system was developed.

The methods used were: the scientific method for the problem statement, compilation of information, the description of the hypothesis which was proved with the experimental method; for its performance prototypes with Prism and MVM Light frameworks were implemented.

With descriptive statistics Chi Cuadrado several parameters were determined to have a significant difference in maintenance time because it was observed that Prism reached 98% and MVM Light 69%, concluding that the Prism framework is the most efficient, easy to configure, extensible and there is a lot of information on the web, thus the VAMEG SOFT system was developed in Visual Studio platform, the Visual Basic programming language, using an quick XP methodology leaving the system running properly.

**ANEXOS**

### TABLA CHI CUADRADO

Nivel de Significancia

Grados de Libertad

v/p	0,55	0,6	0,65	0,7	0,75	0,8	0,85	0,9	0,95	0,975	0,99	0,995
1	0,3573	0,2750	0,2059	0,1485	0,1015	0,0642	0,0358	0,0158	0,0039	0,0010	0,0002	0,0000
2	1,1957	1,0217	0,8616	0,7133	0,5754	0,4463	0,3250	0,2107	0,1026	0,0506	0,0201	0,0100
3	2,1095	1,8692	1,6416	1,4237	1,2125	1,0052	0,7978	0,5844	0,3518	0,2158	0,1148	0,0717
4	3,0469	2,7528	2,4701	2,1947	1,9226	1,6488	1,3665	1,0636	0,7107	0,4844	0,2971	0,2070
5	3,9959	3,6555	3,3251	2,9999	2,6746	2,3425	1,9938	1,6103	1,1455	0,8312	0,5543	0,4118
6	4,9519	4,5702	4,1973	3,8276	3,4546	3,0701	2,6613	2,2041	1,6354	1,2373	0,8721	0,6757
7	5,9125	5,4932	5,0816	4,6713	4,2549	3,8223	3,3583	2,8331	2,1673	1,6899	1,2390	0,9893
8	6,8766	6,4226	5,9753	5,5274	5,0706	4,5936	4,0782	3,4895	2,7326	2,1797	1,6465	1,3444
9	7,8434	7,3570	6,8763	6,3933	5,8988	5,3801	4,8165	4,1682	3,3251	2,7004	2,0879	1,7349
10	8,8124	8,2955	7,7832	7,2672	6,7372	6,1791	5,5701	4,8652	3,9403	3,2470	2,5582	2,1558
11	9,7831	9,2373	8,6952	8,1479	7,5841	6,9887	6,3364	5,5778	4,5748	3,8157	3,0535	2,6032
12	10,7553	10,1820	9,6115	9,0343	8,4384	7,8073	7,1138	6,3038	5,2260	4,4038	3,5706	3,0738
13	11,7288	11,1291	10,5315	9,9257	9,2991	8,6339	7,9008	7,0415	5,8919	5,0087	4,1069	3,5650
14	12,7034	12,0785	11,4548	10,8215	10,1653	9,4673	8,6963	7,7895	6,5706	5,6287	4,6604	4,0747
15	13,6790	13,0298	12,3809	11,7212	11,0365	10,3070	9,4993	8,5468	7,2609	6,2621	5,2294	4,6009
16	14,6555	13,9827	13,3096	12,6243	11,9122	11,1521	10,3090	9,3122	7,9616	6,9077	5,8122	5,1422
17	15,6328	14,9373	14,2406	13,5307	12,7919	12,0023	11,1249	10,0852	8,6718	7,5642	6,4077	5,6973

## **BIBLIOGRAFÍA**

**[1] JORDISAN, PROFESIONAL PROJECT, FRAMEWORKS**

<http://jordisan.net/blog/2006/que-es-un-framework>

[Consulta 2013 09 01]

**[2] MAESTROS DEL WEB, QUE SON LAS BASE DE DATOS**

<http://www.maestrosdelweb.com/principiantes/%C2%BFque-son-las-bases-dedatos/>

[Consulta 2013 09 01]

**[3] VÍCTOR, DEFINICIÓN DE SQL SERVER**

<http://242725.blogspot.com/2007/10/definicion-de-sql-server.html>

[Consulta 2013 09 01]

**[4] OSKARIN HURTATIZ, HISTORIA Y EVOLUCIÓN DE VISUAL STUDIO**

<http://oskarito-hurtatiz-32.blogspot.com/2009/08/historia-y-evolucion-de-visual-studio.html>

[Consulta 2013 09 01]

**[5] MONOGRAFÍAS.COM, VISUAL BASIC**

<http://www.monografias.com/trabajos33/visual-basic/visual-basic.shtml>

[Consulta 2013 09 01]

**[6] MICROSOFT, VISUAL C#**

<http://msdn.microsoft.com/es-es/library/kx37x362.aspx>

[Consulta 2013 09 01]

**[7] MICROSOFT APPLICATION ARCHITECTURE GUIE, 2ND EDITION, ARQUITECTURA DE SOFTWARE**

<http://msdn.microsoft.com/en-us/library/ff650706.aspx>

[Consulta 2013 09 01]

**[8] WINDOWS, PATTERNS & PRACTICES, PATRÓN DE DISEÑO MVVM**

<http://msdn.microsoft.com/es-es/library/windows/apps/jj883732.aspx>

[Consulta 2013 09 01]

**[9] ALEGSA, WPF**

<http://www.alegsa.com.ar/Dic/windows%20presentation%20foundation.php>

[Consulta 2011 06 02]

**[10] CALIDAD Y SOFTWARE, PRISM UN FRAMEWORK PARA WPF**

<http://calidadyssoftware.blogspot.com/2012/03/prism-un-framework-para-wpf.html>

[Consulta 2011 06 02]

**[11] MICROSOFT, DEVELOPER NETWORK, INTRODUCCIÓN A PRISM**

[http://msdn.microsoft.com/en-us/library/ff921153\(v=pandp.40\).aspx](http://msdn.microsoft.com/en-us/library/ff921153(v=pandp.40).aspx)

[Consulta 2011 06 02]

**[12] CODE PROJECT, DISEÑO DE APLICACIÓN DE PRISM**

<http://www.codeproject.com/Articles/507077/PRISM-application-from-the-beginning-part-I>

[Consulta 2013 10 14]

**[13] TEHDENCIAS, MVVM LIGHT**

<http://blogtechdencias.azurewebsites.net/blog/2013/03/12/capitulo-3-el-paquete-mvvm-light/>

[Consulta 2013 10 14]

**[14] MVVM LIGHT TOOLKIT, INSTALACIÓN MVVM LIGHT**

<http://www.galasoft.ch/mvvm/installing/>

[Consulta 2014 02 14]

**[15] ESTADÍSTICA DESCRIPTIVA E INFERENCIAL**

Probabilidad y Estadística Aplicadas a la Ingeniería (Montgomery - Runger) - 2ª Edición

[Cap 1 - 8]

[Consulta 2014 02 16]