

# ESTUDIO COMPARATIVO DE LOS FRAMEWORKS RUBY ON RAILS Y DJANGO PARA LA IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO DE CONTROL Y ADMINISTRACIÓN DE NETWORK MARKETING.

*Autor-Rubén Guerrero<sup>1</sup>, Coautor-Ing. Omar Lara<sup>2</sup>*

<sup>1</sup> Facultad de Ing. En Ciencias Aplicadas, Univ. Técnica del Norte. Av. 17 de Julio 5-21, Ibarra, Imbabura

<sup>2</sup> Carrera de Ingeniería en Sistemas, Univ. Técnica del Norte. Av. 17 de Julio 5-21, Ibarra-Imbabura

rubenignacio1522@hotmail.com, olara@utn.edu.ec

**Resumen.** *El presente aplicativo consiste en implementar un sistema administrativo de network marketing.*

*Las secciones por las que está conformado el presente artículo se describen a continuación:*

*La sección I contiene los antecedentes y problemas que tiene la empresa o entidad en este momento y cuál será la solución ante este hecho.*

*La sección II está conformado por la investigación de cada una de las herramientas y sus características las cuales son estudiadas de acuerdo a lo que se crea conveniente, en esta sección abarcará las arquitecturas de las herramientas entre otros aspectos, como características o que temas son los precisos para realizar una posterior comparación entre los frameworks.*

*La sección III contiene la valoración de los parámetros con los cuales se realizará una comparación entre herramientas para determinar un ganador, de acuerdo a la escala Likert con la cual se trabajó.*

*La sección IV contiene la implementación del sistema dando solución a la administración de network marketing. Esta sección contiene el desarrollo de las fases de la metodología con la que se diseñó el aplicativo.*

*Por último la sección V contiene las conclusiones y recomendaciones también el análisis de impacto que se tiene con este trabajo realizado.*

## Palabras Claves

Network Marketing, Likert, Framework.

**Abstract.** *The present application is to implement an administrative system of network marketing.*

*Sections that this article is formed are described to continuation:br section 1 contains the background and problems that has the company or entity in this moment and what will be the solution to this fact.*

*Section II consists of the research of each of the tools and features which are studied according to creating convenient, this section will cover the architectures of the tools among other aspects, such as features or issues are the precise for further comparison between the frameworks.*

*Section III contains the evaluation of parameters which will include a comparison between tools to determine a winner, according to the Likert scale with which they worked.*

*Section IV contains the implementation of the system giving solution to the administration of network marketing. This section contains the development of phases of the methodology with which we designed the application.*

*Finally the section V contains the conclusions and recommendations also the analysis of impact that is has with this work carried out..*

## Keywords

Network Marketing, Likert, Framework.

## 1. Introducción

El presente capítulo detalla y a la vez informa el ámbito en el cual se va a desarrollar el aplicativo, el problema que existe actualmente, los antecedentes, objetivos y el alcance o solución que brindará dicho proyecto.

## PARA LA IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO DE CONTROL Y ADMINISTRACIÓN DE NETWORK MARKETING

## 1.1 Problema

### 1.1.1 Antecedente

La existencia de herramientas y plataformas para desarrollo de aplicaciones web han evolucionado tanto que los usuarios se han quedado en las más conocidas, sin darse cuenta de las ventajas que ofrecen las nuevas y sofisticadas plataformas de desarrollo web.

### 1.1.2 Situación actual

Actualmente la empresa lleva los trabajos de administración de cada empresario y cliente a papel, teniendo como debilidad y riesgo la manipulación de la información, esta situación se viene dando por la falta de un sistema web que permita administrar y organizar la información de manera correcta.

### 1.1.3 Planteamiento del problema

No existe un sistema web que permita administrar de manera organizada y controlada la información de empresarios y clientes en la empresa.

### 1.1.4 Prospectiva

El estado en el que se encuentra la unidad de administración de la empresa impide que se lleve un control ordenado de los empresarios y sus clientes, por lo tanto a futuro se solucionará este problema con el aplicativo a desarrollarse en el cual se visualizará los empresarios y los clientes de manera clara y ordenada.

## 1.2 Objetivos

### 1.2.1 Objetivo general

Establecer un estudio comparativo entre los frameworks Ruby On Rails y Django para implementar un sistema de control y administración de Network Marketing.

### 1.2.2 Objetivos específicos

- Realizar un análisis actual de la administración de la empresa
- Determinar la métricas de comparación
- Diagnosticar la herramienta que se utilizará para la implementación del sistema.
- Emplear la herramienta seleccionada para proceder a realizar el sistema.

### 1.3 Alcance

Se compararán las siguientes herramientas de desarrollo de software Ruby On Rails y Django en base a

parámetros y criterios de evaluación para así determinar la mejor herramienta que cubra todas las necesidades del usuario.

La implementación del Sistema beneficiará a la empresa Omnilife y a todos sus empresarios, ya que serán administradores de sus redes independientemente.

Los conocimientos adquiridos se aplicarán en el desarrollo de un sistema de administración y control orientado al Network Marketing (Robert Kiyosaki, 2012).

Este sistema tendrá la capacidad de:

- Tener un control sobre los clientes y valorar sus ganancias.

- Verificar las inscripciones que cada uno de los empresarios realice.

- Verificar el nivel en el cual se encuentran ubicados.

El sistema tendrá los siguientes módulos:

- Administración de usuarios: conllevará la creación, administración de cargos o roles y usuarios del sistema.

- Catálogo de proyectos y actividades: llevará un registro de las actividades y proyectos a ejecutarse en la empresa.

- Bitácora: se registrará los clientes que cada usuario inscriba bajo su código o su membresía.

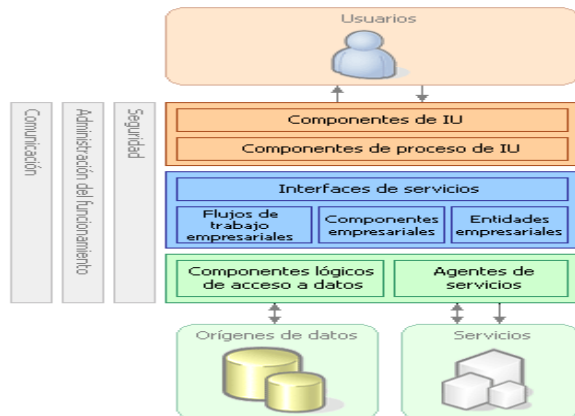
- Reportes

Para la realización del sistema informático se utilizará la metodología en cascada que está basada en procesos secuenciales o lineales, esencialmente como un modelo lineal de desarrollo de software. Este modelo fluye secuencialmente desde el punto inicial al punto final. (Aristega, 2010) (Shari Lawrence Pfleeger, 2002)



**Gráfico 1:** Arquitectura del sistema organizacional de la Empresa

**Fuente:** Propia



**Gráfico 2:** Arquitectura Técnica del Sistema.

**Fuente:** Modelo n-capas, recuperado de <http://wiki-aplidaniel.wikispaces.com/file/view/n-capas.png/251847972/n-capas.png>, 2015

### 1.4 Justificación

El avance tecnológico hoy en día ha hecho que las instituciones o empresas busquen la manera de automatizar sus campos o procesos para facilitar su desempeño. Con el estudio comparativo de las herramientas se busca facilitar la decisión de usar una herramienta para crear un sistema que permitirá agilizar los procesos de control y administración dentro del campo de Network Marketing (Cabrera, 2005).

El análisis de herramientas como proyecto de tesis, permitirá obtener una gran documentación misma que servirá como guía para docentes y estudiantes, además de profesionales en el ámbito de la informática, garantizando así un desarrollo de aplicaciones de excelente calidad.

Será un incentivo en los estudiantes a seguir realizando las debidas investigaciones de nuevas metodologías y herramientas tales como los frameworks para realizar de mejor manera el desarrollo de software.

## 2. Marco Teórico

El presente capítulo contiene temas relacionados a la empresa Omnilife de la cual se dará un breve concepto de lo que es y qué negocio maneja o cuál es su actividad y a la vez contenidos de cada una de las herramientas o frameworks que se utilizará en los próximos capítulos para desarrollo del sistema aplicativo.

### 2.1 Empresa Omnilife

La empresa Omnilife o Grupo Omnilife como se le conoce, es una empresa que se encarga de realizar negocios de multinivel, relacionado esto con el network marketing o negocio del siglo XXI.



**Gráfico 1:** Logotipo de la Empresa Omnilife

**Fuente:** [https://www.omnilife.com/boletin/images/22JUL15/boletin\\_220715\\_01\\_c.jpg](https://www.omnilife.com/boletin/images/22JUL15/boletin_220715_01_c.jpg)

Esta empresa se encarga de elaborar productos multivitamínicos que son suplementos alimenticios que ayudan a mejorar las funciones del cuerpo humano, además de esto también produce productos de belleza. Su sede principal está ubicada en Guadalajara-México. Tiene más de 6000 empleados y más de 5.5 millones de empresarios independientes también conocidos como distribuidores que operan en más de 19 países

### 2.2 Ruby On Rails

Ruby es un lenguaje que creó el japonés Yukihiro Matz Matsumoto el 24 de febrero del año de 1993 y lo presentó públicamente en el año 1995.

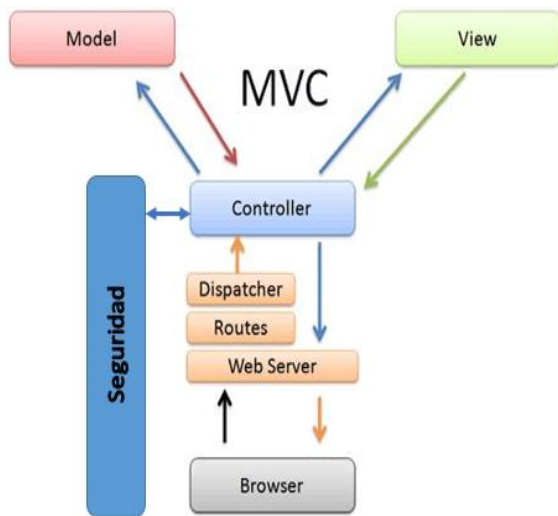


Su nombre Ruby se le dió en son de broma aludiendo al lenguaje que en ese entonces era llamado Perl. Ruby On Rails es un lenguaje de programación creado por David Heinemier y lanzado al público en el año de 2004. Rails es un lenguaje se podría decir que es obstinado en decir que existen mejores cosas y poder lograrlas en cuanto a la programación se refiere, se ha puesto en las manos de programadores para facilitarle el trabajo ya que evita el tener que repetir código, ya que ésta es la filosofía de RoR (Ruby On Rails), Don't repeat Yourself.

Rails es fundamentalmente una implementación de un patrón de diseño de software conocido como MVC.

Este divide un sistema en tres componentes:

- El modelo de datos.
- La vista o interfaz gráfica.
- El controlador que contiene la lógica del negocio.



**Gráfico 4:** Arquitectura MVC de Ruby On Rails

Fuente: propia

### 2.3 Django



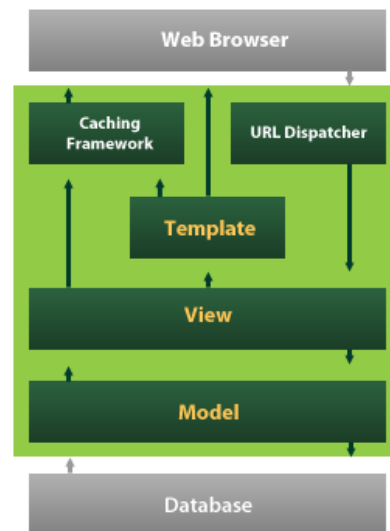
Django es un framework que tiene por característica principal facilitar la tarea del programador o desarrollador de software. Este framework fue trabajado en la administración de páginas de noticias, tal es el caso que su diseño se hace evidente debido a que proporciona una serie de características que facilitan la creación o desarrollo ágil de páginas orientadas a contenidos.

Algunos autores mencionan que Django crea una página administrativa o aplicación incorporada para la administración de contenidos y páginas; ésta página permite crear, actualizar y eliminar objetos del contenido, pero sin duda lleva un registro de cada acción que realice, proporciona de igual forma una interfaz para administrar los usuarios y grupos de usuarios teniendo en cuenta los permisos que se puede asignar.

Django también cuenta con herramientas que permiten proporcionar un sistema de comentarios, herramientas de contenido vía RSS y Atom, cuenta también con páginas planas que sirven para gestionar páginas de contenido sin necesidad de realizar controladores o vistas para dichas páginas, y por último pero no menos importante cuenta con un sistema de redirección de URLs (Uniform Resource Locator o Localizador Uniforme de Recursos).

Django sigue la arquitectura Modelo Vista Controlador (MVC), aunque los que desarrollan sus proyectos con este framework hacen énfasis en que prefieren no basarse a esto

sino que en Django lo que se llama controlador en un verdadero framework MVC aquí se lo denomina vista y la vista en cambio toma el nombre de plantilla. Debido a las capas que tiene el framework Django que son mediator y foundation, permite que los programadores se dediquen a construir únicamente los objetos Entity y la lógica para mostrar o presentar su trabajo y también desde luego el control para ellos.



**Gráfico 5:** Arquitectura MVC en Django

Fuente: [http://blog.chattyhive.com/wp-content/uploads/2014/01/Django\\_mvc.png](http://blog.chattyhive.com/wp-content/uploads/2014/01/Django_mvc.png)

- **Lógica de presentación**

Aquí se maneja la interacción del programador o usuario y su equipo de computación. (PC, Laptop). En El framework Django, ésta tarea es designada al *template* y al *template loader* que recopilan la información y la presentan al usuario.

Dentro de esta capa de presentación también se encuentra el sistema de configuración de URLs.

- **Control**

En la capa de control se encuentra la lógica de aplicación o el programa en sí. En el framework Django son presentados por los manipulators y las views, la capa de lógica de presentación depende de ésta capa y ésta depende de igual forma de la capa de dominio.

- **Mediator**

Es el principal encargado de manejar la interacción entre los subsistemas Foundation y Entity. En esta parte se

realiza el mapeo objeto-relacional que está a cargo del motor del framework Django.

- **Entity**

Entity o subsistema Entity es el encargado de manejar los objetos de negocio, además tiene una cierta ventaja ya que por el mapeo objeto-relacional de Django permite escribir los objetos que son de tipo Entity de forma fácil y estándar.

- **Foundation**

Foundation o subsistema foundation tiene la principal tarea de manejar el trabajo con las bases de datos a bajo nivel. Existe el soporte a nivel de foundation para algunas BDD<sup>1</sup> pero para otras aún está en prueba, para saber todo acerca del soporte.

## 2.4 Postgresql

PostgreSQL es un potente sistema de base de datos, de código abierto objeto-relacional. Cuenta con más de 15 años de desarrollo activo y una arquitectura probada que se ha ganado una sólida reputación por su fiabilidad, integridad de datos y corrección.

Se ejecuta en todos los sistemas operativos, incluyendo Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64), y Windows. Es totalmente compatible con ACID, tiene soporte completo para claves foráneas, combinaciones, vistas, triggers y procedimientos almacenados (en varios idiomas). Incluye más de SQL: 2008 tipos de datos, incluyendo INTEGER, numéricos, booleanos, CHAR, VARCHAR, DATE, INTERVALO y TIMESTAMP. También es compatible con el almacenamiento de grandes objetos binarios, incluyendo imágenes, sonidos o vídeos. Cuenta con interfaces de programación nativo de C / C ++, Java, .Net, Perl, Python, Ruby, Tcl, ODBC, entre otros, y la documentación excepcional.

Una base de datos de clase empresarial, PostgreSQL cuenta con características sofisticadas como Multi-Version Concurrency Control (MVCC), punto en el tiempo de recuperación, tablespaces, replicación asincrónica, transacciones anidadas (puntos de retorno), en línea copias de seguridad / calor, un sofisticado planificador de consulta

/ optimizador, y escribir por delante el registro para la tolerancia a fallos.

## 3. Análisis antes y después de implementar el sistema.

En este tiempo que la empresa ha ido generando herramientas, no lo ha hecho para uso de cada uno de los usuarios, sino de manera global.

Lo que se pretende hacer hoy, es el análisis de una herramienta que sea capaz de dar información al usuario de manera sencilla y dinámica. Para esto la idea que se tiene y se la va a plasmar es que cada uno de los usuarios tenga un software que les permita llevar un control de publicidad y de ingresos al Network marketing en esta empresa.

Con esta idea se hará que cada usuario cree su empresa y lleve el control de las personas que van a trabajar en su organización, a la vez es una herramienta publicitaria o de información, porque contendrá información de los servicios que brinda y de la forma de realizar el negocio.

### 3.1 Estudio comparativo de herramientas

Una correcta comparación respecto a las características más importantes en cuanto a los Frameworks Ruby On Rails y Django, es el eje fundamental para seleccionar cuál de las herramientas será la mejor para el desarrollo del aplicativo, y la que genere una mejor solución al problema planteado. Estos puntos deben ser analizados de manera conveniente para obtener resultados positivos.

Los puntos que se toman en cuenta deberán ser valorados por una escala la cual deberá asumir los parámetros que son tomados de acuerdo a la necesidad que se ha planteado y para satisfacción de un mejor entorno de desarrollo en cuanto a la programación se refiere.

#### 3.1.1 Escalas de valoración

Una escala de valoración es la que permite obtener resultados de un determinado tema a tratar. Con esto se podrá sacar conclusiones y determinar resultados sobre el tema que se elija. Una escala siempre contiene parámetros los cuales son las bases sobre las que se va a trabajar o

---

<sup>1</sup> Base de Datos

PARA LA IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO DE CONTROL Y ADMINISTRACIÓN DE NETWORK MARKETING

también conocidos como condicionantes, variables, etc. Existe un sinnúmero de escalas las que permiten trabajar de manera efectiva para un análisis o investigación por el momento se usará la técnica o escala e Likert.

Bien para realizar la comparación se utilizará la escala de LIKERT como se ha dicho antes, la cual mide o valora las opciones o ítems que se propone para realizar un trabajo, en dicha escala se mide el criterio de las personas o de la información que se obtiene en una investigación, la cual puede manejar 4 o 5 o 7 parámetros todo esto en función de lo que estime conveniente el usuario.

Para tener una idea de la escala de Likert se cita un ejemplo, el cual indicará la manera que tiene para trabajar dicha escala:

Cuadro 2. Escala de evaluaciones

Valor	Significado
1	El atributo no es importante en la evaluación del desempeño del encuestado.
2	El atributo es ligeramente importante en la evaluación del desempeño del encuestado.
3	El atributo es regularmente importante en la evaluación del desempeño del encuestado.
4	El atributo es importante en la evaluación del desempeño del encuestado.
5	El atributo es muy importante en la evaluación del desempeño del encuestado.

Gráfico 2: Ejemplo de uso de Likert

Fuente:

<http://www.scielo.org.mx/img/revistas/peredu/v34n137/a9c2.jpg>

### 3.1.2 Parámetros de evaluación

Los parámetros que se describen a continuación serán los pilares fundamentales para llegar a un resultado el cual se está buscando. Los parámetros son los siguientes:

- **Aprendizaje:** para ser una herramienta eficiente debe ser fácil su uso y su comprensión, para así de este modo interactuar de una manera clara o precisa ahorrando tiempo y evitando tener problemas al momento de realizar alguna aplicación o software.

- **Portabilidad:** este aspecto es un importante punto debido a que necesitamos que nuestra aplicación se puede acoplar a cualquier sistema o sea compatible con cualquier infraestructura.
- **Documentación:** en este punto hace referencia a la facilidad de encontrar documentación (libros), páginas web, sitios o blogs que sean de mucha ayuda con su información de las herramientas que se investigan en esta parte.
- **Soporte:** toda herramienta o a su vez toda aplicación debe tener soporte, esto quiere decir que debe existir una documentación en cuanto a su manejo, a sus ventajas, desventajas, blogs o libros en los cuales se hable del mantenimiento o uso correcto de la herramienta en este caso de los frameworks que se va a investigar.
- **Plantillas:** son las herramientas que tienen por defecto cada uno de los frameworks para facilitar al usuario la creación de modelos de páginas web o blogs.
- **Vistas Dinámicas:** este parámetro es una de las opciones que más llama la atención ya que son útiles para realizar actualizaciones inmediatas en una página web.
- **Seguridad:** el internet como todos sabemos es un peligro en cuanto a seguridad, pero los frameworks han trabajado mucho en eso para brindar una mejor protección de los datos e información.
- **Sesiones:** un sitio seguro debe tener una pantalla para realizar un inicio de sesión de usuario para protección y además según esto pueda realizar acciones según su categoría.

Cada uno de estos parámetros, son importantes para obtener un resultado de un software que ayude a solucionar el problema pero que también tenga las garantías necesarias para que la información sea segura y evite tener conflictos en esta parte.

En la siguiente tabla se explica la valoración y cómo se va a trabajar para sacar el resultado final y la herramienta ganadora.

Se tendrá el nombre de la herramienta, su indicador o parámetro, su valoración la cual estará entre el rango de 1 a 5 y ésta significará lo siguiente.

**Tabla 1:** Puntaje de Indicadores Establecidos

Nombre del framework		
Características	Puntos	Porcentaje
Deficiente	1	20%
Poco eficiente	2	40%
Limitado	3	60%
Eficiente	4	80%
Muy Eficiente	5	100%

**Fuente:** Propia

De esta manera se obtiene los indicadores que determinan cuál será la herramienta ganadora para dar solución al problema que se tiene actualmente. Estos parámetros serán esenciales al momento de calificar a los frameworks seleccionados para su estudio porque son las características de más importancia en los dos, según las investigaciones que se ha hecho.

### 3.1.3 Determinación del software ganador

Realizando un análisis de todos los indicadores para determinar el software ganador, se obtiene una tabla general de los puntajes que contiene cada uno de los ítems tomados como base con su respectiva calificación en porcentaje, para luego encontrar al mejor framework o herramienta para implementar el sistema, de acuerdo a la necesidad que actualmente se tiene, con esto se logrará dar una solución eficiente y de manera dinámica al problema que se ha suscitado.

**Tabla 2:** Resultados porcentuales de frameworks según sus indicadores

Parámetros	Ruby On Rails	Django
Aprendizaje	80%	100%
Portabilidad	80%	100%
Documentación	80%	100%
Soporte	100%	100%
Plantillas	80%	100%
Vistas Dinámicas	100%	100%
Seguridad	100%	100%
Sesiones	100%	100%
Total:	720%=90%	800%=100%

**Fuente:** Autor

Para una mayor apreciación se ha tomado el número de valoración que se le ha dado a cada parámetro para que sea

de mayor comprensión para el lector en el cual se escribe el valor según lo estimado en la escala de 1 a 5 en la escala de Likert para valorar los parámetros que el autor lo requiera y de acuerdo a las necesidades que se haya tenido al momento de querer dar solución al problema encontrado actualmente en la empresa, o a la vez en la necesidad que tenga el usuario.

Para ello a continuación se presenta la tabla que contiene dichos ítems ya con su calificación dada según la investigación que se ha realizado para obtener la mejor herramienta de las dos que se ha tomado en cuenta para su investigación, en esta se muestra la calificación en valores o números según la escala de Likert:

**Tabla 3:** Tabla de Resultados según la escala de Likert

Parámetros	Ruby On Rails	Django
Aprendizaje	4	5
Portabilidad	5	5
Documentación	4	5
Soporte	5	5
Plantillas	4	5
Vistas Dinámicas	5	5
Seguridad	5	5
Sesiones	5	5
Total:	37=90%	40=100%

**Fuente:** Propia

Así obtenemos el resultado de cada uno de los parámetros y por ende se logra la obtención de una herramienta ganadora en este caso como se puede observar el ganador el software de desarrollo Django, el cual obtiene un porcentaje de 100% de acuerdo a los ítems que se le ha planteado, según las necesidades que hacen referencia a lo que se necesita para el mejor desarrollo de un sistema de Network marketing

Los resultados deben poder ser vistos y entendidos de forma rápida y clara. Es por ello por lo que la construcción de esta sección debe comenzar por la elaboración de las tablas y figuras, y sólo posteriormente redactar el texto pertinente en función de ellas. El primer párrafo de este texto debe ser utilizado para resumir en una frase concisa, clara y directa, el hallazgo principal del estudio. Esta sección debe ser escrita utilizando los verbos en pasado.

Aunque se utiliza mucho la voz pasiva o el impersonal (“se ha encontrado que...”) como prefieren algunos editores. En el texto se deben citar todas las tablas y figuras, en caso de ser tomadas de otros estudios deberán constar las referencias bibliográficas. Todas las tablas y figuras deben tener su respectiva leyenda.

En cuanto al formato, esta sección se puede organizar en subtítulos, y cada uno de estos no puede subdividirse nuevamente. Para un ejemplo referirse a los anexos.

## 4. Implementación del sistema

Una vez realizado al análisis correspondiente de las herramientas y tomando en cuenta cada ítem que se le ha dado para saber su resultado frente a otra herramienta hemos obtenido un ganador y ahora se procede con la realización de la aplicación.

Para realizar la implementación del sistema se va usar la metodología Web llamada OOHDM (Método de Diseño de Hipermedia Orientado a Objetos).

Esta metodología está orientada a lo que es el diseño de aplicaciones hipermedia y para la Web, está convirtiéndose en la metodología más usada para el diseño de aplicaciones de diferentes tipos tanto como son las galerías que son interactivas, los famosos sitios web y las presentaciones multimedia, que se encuentran en auge total en las redes de internet.

La metodología OOHDM es inspirada en el modelo conocido como HDM (Modelo de Diseño de Hipermedia), el cual separa los aspectos globales y estructurales de una aplicación.

Pero se distingue por su orientación a objetos, además ésta metodología tiene 4 fases o etapas las cuales son:

- **Diseño Conceptual**
- **Diseño Navegacional**
- **Diseño de Interfaces Abstractas**
- **Implementación**

Cada una de las etapas o fases definen un cierto modelo o esquema específico en el cual se introducen las clases o los nuevos elementos.

**Diseño Conceptual.-** en esta etapa se realiza una construcción de un esquema conceptual el cual es representado por las clases de dominio llamados objetos y también las relaciones entre dichas clases.

En esta etapa lo más usado es un modelo de entidades y relaciones llamado modelo de datos semántico. El modelo de metodología OOHDM (Método de Diseño de Hipermedia

Orientado a Objetos), propone un esquema basado en clases, relaciones y subsistemas, llamado también esquema conceptual.

**Diseño Navegacional.-** en esta etapa se define todas las clases navegacionales así como son los nodos y los enlaces. Por otro lado también se tiene las estructuras índices y visitas las cuales son guiadas.

Los enlaces no son sino los que se derivan de las relaciones que existen y los nodos son la representación de cada una de las ventanas lógicas o llamadas views. El diseñador realiza una descripción de la estructura navegacional de manera clara y con términos llamados de contexto navegacionales. OOHDM (Método de Diseño de Hipermedia Orientado a Objetos), propone un diseño o modelo muy enriquecido para el dominio de lo que es la aplicación.

El modelo de hipermedia sin embargo se define en un nivel llamado nivel de abstracción y consta de dos partes que son: las clases navegacionales y los contextos navegacionales.

**Diseño de Interfaces Abstractas.-** como su nombre lo indica esta interfaz está dedicada a la especificación de la interfaz abstracta.

Esta se define de una manera en la cual deberán aparecer los llamados contextos navegacionales. En esta etapa se realiza un tipo de descripción de los elementos u objetos que tiene la interfaz o view, y se los asociará con los objetos de navegación desde luego que deberían ir con cada una de las funcionalidades que tenga cada objeto.

Si se separa el diseño navegacional y lo que sería el diseño de interfaz abstracta permitirá construir dos interfaces diferentes para el modelo navegacional.

**Implementación.-** esta cuarta etapa es la dedicada a poner en práctica todos los objetos de interfaz con los que cuenta la aplicación.

Estos serían, sus relaciones entre esquemas siendo estos los conceptuales, navegacionales y los objetos que son de interfaz en el modelo OOHDM (Método de Diseño de Hipermedia Orientado a Objetos), que propone un diseño o modelo muy enriquecido para el dominio de lo que es la aplicación.



Este método es un modelo abierto porque los modelos de dominio no vienen impuestos y los de soporte permiten la especialización de las clases y los contextos navegacionales.

### 4.1 Diseño Conceptual

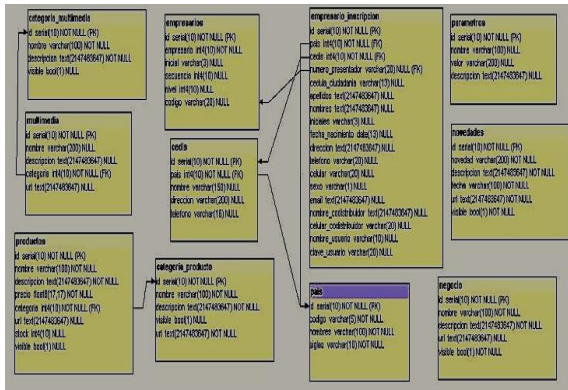


Gráfico 3: Diseño Conceptual de la Aplicación

Fuente: Propia

En el modelo de la base de datos se refleja cada una de las clases conocidas también como objetos las cuales se relacionan entre sí para poder ayudar a la realización del software o aplicación de administración de Network Marketing.

### 4.2 Diseño Navegacional

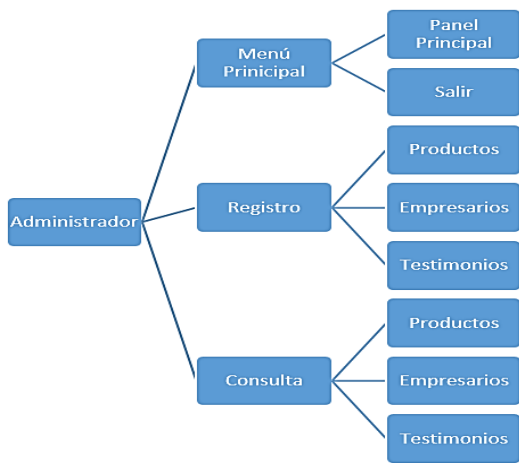


Gráfico 8: Modelo Navegacional del Administrador

Fuente: Propia

El modelo navegacional no es más que el diseño gráfico de las funciones que puede realizar el administrador con respecto a la aplicación. Su administración, su

funcionamiento, su modificación e incluso eliminación de la misma. Esto para una mayor comprensión del usuario o lector en este caso, para que posteriormente comprenda la nueva metodología Web.

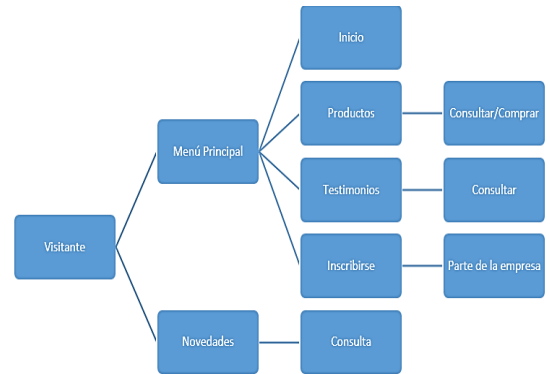


Gráfico 4: Modelo Navegacional del Cliente

Fuente: Propia

El modelo navegacional del usuario o visitante explica las opciones que puede revisar, podrá hacerlo desde cualquier lugar, además de esto existe la opción de poder formar parte de la empresa, esto lo realiza en la opción de inscribirse la cual pedirá todos sus datos y explicará en que consiste dicha opción, así como también los beneficios de pertenecer a la empresa.

### 4.3 Diseño de Interfaz Abstracta

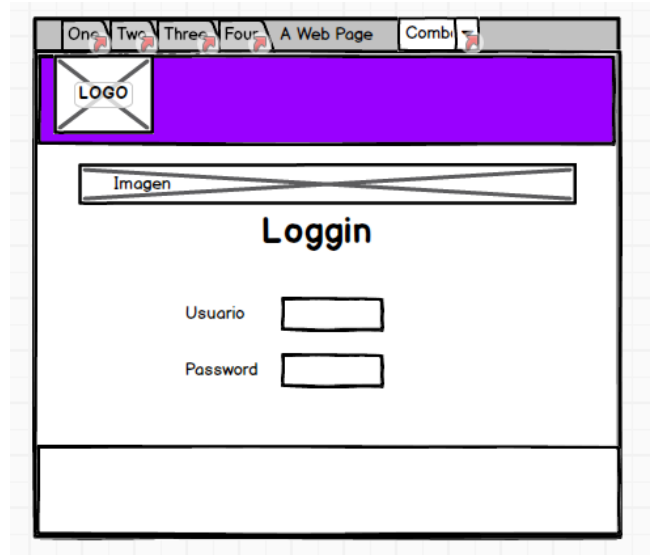


Gráfico 5: Fase de interfaz abstracta Pantalla de Ingreso de Usuarios.

Fuente: Propia

En la fase de implementación abstracta se crea un bosquejo general de las pantallas que se va a usar en el

---

 PARA LA IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO DE CONTROL Y ADMINISTRACIÓN DE NETWORK MARKETING
 

---

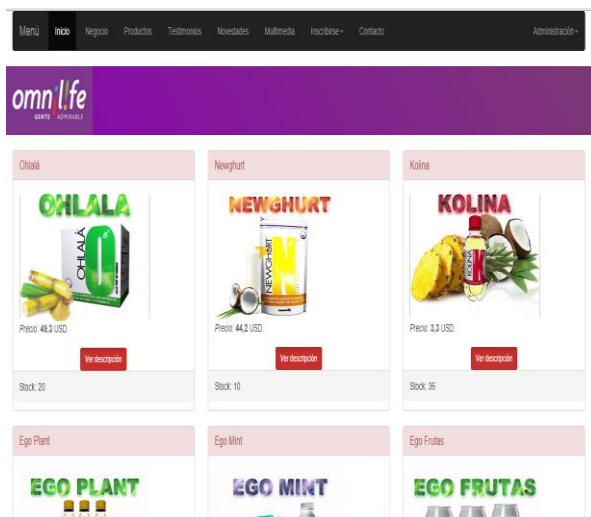
sistema web, esto es una idea de cómo se puede presentar la vista para el usuario.

Este servirá de ayuda debido a que se debe tener una idea clara antes de realizar la implementación, encontrando la mejor manera de poder llegar al usuario y que sea de su completo agrado a la hora de visitar el sitio web.

Para ello pues se realiza una presentación en cuanto a los elementos se refiere, como se lo puede apreciar, así pues constará de una cabecera, de un contexto, imágenes, que servirán de información y comunicación con el usuario.

Todo esto de manera dinámica y entretenida dando una mejor presentación de la empresa a los usuarios o clientes.

#### 4.4 Implementación



**Gráfico 6:** Vista de productos

**Fuente:** Propia

La implementación es la fase donde se da estilo y color a cada una de las pantallas realizadas como bosquejo siendo estas un reflejo del trabajo realizado anteriormente y de toda la investigación que se logró en el trabajo.

Tenemos como ejemplo la vista de productos, es una de las tantas vistas que tendrá el cliente o usuario.

La vista de productos contendrá información clara de cada elemento para mejor información del usuario, en esta ventana existe también la opción de que una vez que se haya informado de lo que realiza cada producto lo pueda adquirir.

### 5. Conclusiones

- La realización e implementación del sistema de control de Network Marketing, permite de manera eficiente tener una exactitud en cuanto a ganancias se obtiene, para mayor superación de la empresa y además mejorará de forma notable la

comunicación con los empresarios que estén dentro de la red.

- Una vez realizado este sistema y estar en funcionamiento en la Web, se obtendrá una ganancia en comunicación con el público debido a que estará disponible para todos, en cualquier momento para que se puedan informar acerca de los beneficios.
- La aplicación de una nueva metodología aplicada a la Web permite tener otro punto de vista en cuanto a tecnologías y metodologías se refiere, ya que se sale de los esquemas a los que se está acostumbrados. Esto ayudará a tener una mayor facilidad de comprensión de las aplicaciones Web y sus procedimientos.
- Concluida la aplicación resulta notorio el ahorro de ciertos recursos tales como: tiempo, dinero, papel y el trabajo de llegar a muchas personas. Esto es una manera de mejorar tanto la comunicación, los pedidos, las ventas y de esta manera se beneficiará la empresa ya que tiene la opción de realizar nuevas tareas de mejor manera. En la conclusión se debe dar respuesta al problema planteado en la introducción, dar a conocer cuál fue la contribución real de la investigación, saber a qué conclusiones se arribó y a las implicaciones teórico-prácticas que se pueden inferir.

### Agradecimientos

Mi agradecimiento eterno a mi esposa e hijas que supieron apoyarme en todo momento, para poder realizar este trabajo.

A mi familia, mis padres y hermanas, quienes supieron darme apoyo cuando tenía momentos de flaqueza en el transcurso del periodo de la elaboración de mi trabajo de tesis.

Al Ing. Omar Lara quien se dió la tarea de revisarnos los documentos y corregir cada una de las debilidades que se mostraban en ellos.

### Referencias Bibliográficas

#### Libros

- [1] Arias Ángel. Aprende a Programar con Ruby on Rails. (2014). RC Libros.
- [2] Hinojosa Gutierrez Angel Pablo. Phyton paso a paso. (2016). RA-MA.
- [3] Rerez Castaño Arnaldo. Phyton Fácil. (2016). Marcombo, S.A.
- [4] Knowlton Jim. Phyton: Crear, Modoficar, Reutilizar. (2009). Anaya Multimedia.

- [5] Buttu Marco. El Gran Libro de Phyton. (2016). Marcombo S.A.
- [6] Summerfield Mark. Phyton 3. (2009) Anaya Multimedia.
- [7] González R Patricia. Programacion en Ruby on Rails. (2013). (L. RC, Ed.)
- [8] Realpe Rosero Christian Fernando. Análisis y estudio de Tecnología Ruby on Rails con bases de datos Postgres para aplicaciones Web 2.0. Aplicativo: Implementación del Portal Web 2.0 para la Mancomunidad de la Cuenca del Río Mira. (2013). Ibarra: [TESIS] UTN.
- [9] Kiyosaki Robert, D. T. El toque de midas. (2012). Sin Editorial.
- [10] Sam, Thomas, & Dave. Desarrollo Web con Rails. (2009). Anaya Multimedia.
- [11] Ponce Moreno Santiago. Ruby on Rails. Desarrollo práctico de aplicaciones web. (2013). RC Libros.
- [12] Ponce Moreno, Santiago. Ruby on Rails: Desarrollo práctico de aplicaciones web. (2013). RC Libros.
- [13] Ponce Moreno Santiago. Ruby on Rails: Desarrollo práctico de aplicaicones web. (2013). RC libros.
- [14] Chazallet Sebastien. Phyton3: Fundamentos de Lenguaje. (2015). ENI.
- [15] Lawrence Pflieger Shari, E. Q. Ingeniería de software: teoría y práctica. (2012). Pearson Education.
- [16] Sprenger, S., & Kieran Hayes. Ruby on Rails 3.1 Profundidad. (2012). Dpunkt. Verlag.
- [17] Stefan Tennigkeit Michael Voigt. Ruby on Rails 3. (2010). Entwickler.Press.
- [18] Tate, B. A., & Curt Hibbys. Ruby on Rails. (2007). Anaya Multimedia.
- [19] Deepak Vohra. Ruby on Rails para PHP y Desarrolladores Java. (2012). Springer.
- [20] Shaw Zed A. Aprenda a Programar con Phyton. (2014). Anaya Multimedia.
- WEB**
- [21] Apress. (2016). La Guía Definitiva de Django. Consultado el 15 de Agosto de `2015, de La Guía Definitiva de Django: [http://www.academia.edu/7698037/Django\\_La\\_guia\\_Definitiva\\_Django\\_Software\\_Foundation](http://www.academia.edu/7698037/Django_La_guia_Definitiva_Django_Software_Foundation)
- [22] Cristalab (2012). Python en la web con Django (VI): web de administración. Consultado el 20 de Octubre de 2015, de Python en la web con Django (VI): web de administración: <http://www.cristalab.com/tutoriales/python-en-la-web-con-django-vi-web-de-administracion-c1042861/>
- [23] Alvarez Miguel Angel (2014). Qué es MVC. Consultado el 6 de Agosto de 2015, de Qué es MVC: <http://www.desarrolloweb.com/articulos/que-es-mvc.html>
- [24] Gaitán, F. (2013). Ruby on Rails, parte 2: Modelo Vista Controlador. Consultado el 29 de Octubre de 2015, de Ruby on Rails, parte 2: Modelo Vista Controlador: <http://fernandogaitan.com.ar/ruby-on-rails-parte-2-modelo-vista-controlador/>
- [25] Guerrero, M. B. (2013). Ruby on Rails desde cero: Primeros pasos. Consultado el 20 de Mayo de 2015, de Ruby on Rails desde cero: Primeros pasos: <http://html5facil.com/tutoriales/ruby-on-rails-desde-cero-primeros-pasos/>
- [26] Lapuente, M. J. (2013). Modelo OOHDM. Consultado el 5 de Noviembre de 2015, de Modelo OOHDM: <http://www.hipertexto.info/documentos/oohdm.htm>
- [27] LibrosWeb. (2006-2016). El libro de Django 1.0. Consultado el 4 de Marzo de 2016, de El libro de Django 1.0: [http://librosweb.es/libro/django\\_1\\_0/](http://librosweb.es/libro/django_1_0/)
- [28] LibrosWeb. (2006-2016). Hola Rails. Consultado el 20 de Septiembre de 2015, de Hola Rails: [http://librosweb.es/libro/introduccion\\_rails/capitulo\\_4.html](http://librosweb.es/libro/introduccion_rails/capitulo_4.html)
- [29] LibrosWeb. (2006-2016). Introducción a Ruby on Rails. Consultado el 5 de enero de 2016, de Introducción a Ruby on Rails: [http://librosweb.es/libro/introduccion\\_rails/](http://librosweb.es/libro/introduccion_rails/)
- [30] Picca Carlos. (2013). Django desde Cero: Vistas Dinámicas. Consultado el 15 de Diciembre de 2015, de Django desde Cero: Vistas Dinámicas: <http://codehero.co/django-desde-cero-vistas-dinamicas/>
- [31] Sin Autor. (2012). Deshacer cambios en git. Consultado el 17 de Mayo de 2015, de Deshacer cambios en git: <https://mxrubyonrails.wordpress.com/page/2/>
- [32] Llauradó Oriol. (2014). La Escala de Likert: Qué es y cómo utilizarla. Consultado el 25 de Diciembre de 2015, de La Escala de Likert: Qué es y cómo utilizarla: <http://www.netquest.com/blog/es/la-escala-de-likert-que-es-y-como-utilizarla/>

## Sobre los Autores...

**Autor:** Rubén Ignacio Guerrero Benalcázar Estudiante de Ingeniería en Sistemas Computacionales en la Universidad Técnica del Norte de la ciudad de Ibarra-Ecuador.

**Coautor:** Ing. Omar Lara. Docente de Ingeniería en Sistemas Computacionales en la Universidad Técnica del Norte de la ciudad de Ibarra-Ecuador.