

UNIVERSIDAD INTERNACIONAL SEK

FACULTAD DE ARQUITECTURA E INGENIERÍAS

Trabajo de fin de carrera titulado:

**“DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA
INFORMÁTICO WEB PARA LA GESTIÓN Y ADMINISTRACIÓN DE
PROCESOS EN LA EMPRESA ESPECIGEST”**

Realizado por:

BRYAN ALEXIS GUAMBA CHAMORRO

Director del proyecto:

MSC .DANIEL RIPALDA

Como requisito para la obtención del título de:

INGENIERO DE SISTEMAS EN DISEÑO Y MULTIMEDIA

Quito, Julio del 2015

DECLARACIÓN JURAMENTADA

Yo, BRYAN ALEXIS GUAMBA CHAMORRO, con cédula de identidad # 1715339444, declaro bajo juramento que el trabajo aquí desarrollado es de mi autoría, que no ha sido previamente presentado para ningún grado o calificación profesional; y, que ha consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración, cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la UNIVERSIDAD INTERNACIONAL SEK, según lo establecido por Ley de Propiedad Intelectual, por su reglamento y por la normativa institucional vigente.

Bryan Alexis Guamba Chamorro

C.C.: 1715339444

DECLARATORIA

El presente trabajo de investigación titulado:

**“DISEÑO, DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA
INFORMÁTICO WEB PARA LA GESTIÓN Y ADMINISTRACIÓN DE
PROCESOS EN LA EMPRESA ESPECIGEST”**

Realizado por

BRYAN ALEXIS GUAMBA CHAMORRO

Como requisito para la obtención del Título de:

INGENIERO DE SISTEMAS EN DISEÑO Y MULTIMEDIA

Ha sido dirigido por el profesor

MSC. DANIEL RIPALDA

Quien considera que constituye un trabajo original de su autor

Msc. Daniel Ripalda

DIRECTOR

PROFESOR INFORMANTE

ING. VERÓNICA RODRÍGUEZ

Después de revisar el trabajo presentado,
Lo ha calificado como apto para su defensa oral ante
El tribunal examinador

Ing. Verónica Rodríguez

Quito, 30 de Junio de 2015

DEDICATORIA

Dedico el presente trabajo de investigación a mis padres quienes supieron inculcarme valores y principios que han guiado mi vida. Gracias Padres por estar siempre, a mis profesores quienes con su apoyo incondicional han sabido darme el conocimiento necesario para alcanzar mis metas.

AGRADECIMIENTO

A mis padres, quienes en todo momento han estado a mi lado para apoyarme y brindarme todo su apoyo para seguir adelante.

Al profesor Daniel Ripalda por su acertada dirección del proyecto de investigación. Su profesionalismo y entrega fueron determinantes a la hora de conformar este documento.

A la Universidad Internacional SEK, por su esfuerzo de formar profesionales íntegros.

TABLA DE CONTENIDO

DECLARACIÓN JURAMENTADA.....	iii
DECLARATORIA	iv
PROFESOR INFORMANTE	v
DEDICATORIA.....	vi
AGRADECIMIENTO.....	vii
ÍNDICE DE TABLAS.....	xi
ÍNDICE DE FIGURAS	xi
CAPÍTULO I.....	12
INTRODUCCIÓN.....	12
1.1 El Problema de Investigación	12
1.1.1 Planteamiento del Problema	12
1.1.2 Objetivo General	13
1.1.3 Objetivos Específicos	13
1.1.4 Justificación.....	14
1.2 Marco Teórico	15
1.2.1 Ingeniería de software	15
1.2.2 Desarrollo de software.....	16
1.2.3 Metodologías ágiles para el desarrollo de software.....	16
1.2.4 Herramientas para el desarrollo de software.....	20
1.2.5 Base de datos.....	21
1.3 Marco conceptual.....	22
1.3.1 Sistema Informático	22
1.3.2 Aplicación Web.....	23

1.3.3	HTML	23
1.3.4	CSS	24
1.3.5	Bootstrap	24
1.3.6	PHP	24
1.3.7	Laravel	25
1.3.8	Javascript	25
1.3.9	Angularjs	26
1.3.10	MongoDB	26
CAPITULO II.....		27
MÉTODO		27
2.1	Análisis	27
2.1.1	Estudio preliminar	27
2.1.2	Descripción General del Sistema	31
2.2	Estudio de factibilidad	31
2.2.1	Factibilidad Operativa.....	31
2.2.2	Factibilidad Técnica	32
2.2.3	Factibilidad Económica.....	35
2.3	Diseño	35
2.3.1	Solución Técnica	35
2.3.2	Diagrama de Casos de Uso.....	36
2.3.3	Diagrama de arquitectura.....	37
2.3.4	Diagrama de clase	38
2.3.5	Diagrama de base de datos	39
2.4	Arquitectura del ambiente de desarrollo	42
2.4.1	Arquitectura del ambiente en producción	42
2.4.2	Plataforma	43

2.4.3	Esquema de distribución del sistema Web	44
2.4.4	Descripción de las interfaces.....	44
2.5	Administración y Seguridad.....	46
2.5.1	Acceso a niveles de usuario	46
2.5.2	Auditoría	47
CAPITULO III		48
RESULTADOS		48
3.1	CONSTRUCCIÓN	48
3.1.1	Generalidades	48
3.2	Diagramas Finales	49
3.2.1	Diagrama de paquetes.....	49
3.2.2	Diagrama de componentes.....	49
3.2.3	Descripción de componentes.....	50
3.3	IMPLEMENTACIÓN.....	51
3.3.4	Capacitación	56
3.3.5	Explotación y Beneficios.....	56
3.3.6	Mantenimiento	57
3.3.6.1	Preventivo.....	57
3.3.6.2	Correctivo	57
CAPÍTULO IV		58
DISCUSIÓN		58
4.1	CONCLUSIONES	58
4.2	RECOMENDACIONES	60
BIBLIOGRAFÍA		61

ÍNDICE DE TABLAS

Tabla #1 Características del equipo para el desarrollo del sistema	34
Tabla #2 Software necesario para el desarrollo del sistema	34
Tabla #3 Software necesario para el desarrollo del sistema	35
Tabla #4 Características del equipo para el ambiente de pruebas	42
Tabla #5 Plataforma para el sistema	43

ÍNDICE DE FIGURAS

Figura #1 Diagrama de Caso de Uso Root	36
Figura #2 Diagrama de Caso de Uso Administrador	36
Figura #3 Diagrama de Caso de Uso Ventas	37
Figura #4 Diagrama de Caso de Uso Compras	37
Figura #2 Diagrama de Arquitectura	37
Figura #3 Diagrama de clases de controladores	38
Figura #4 Diagrama de clases de modelos	39
Figura #5 Diagrama base físico de la base de datos	40
Figura #6 Diagrama de la estructura final de la base de datos	41
Figura #7 Inicio de sesión del sistema	45
Figura #8 Inicio de sesión fallido	45
Figura #9 Dashboard del sistema	46
Figura #10 Diagrama de paquetes	49
Figura #11 Diagrama de paquetes	49
Figura #12 Diagrama de despliegue	50
Figura #13 LINODE Manager	51

Figura #14 Configurar nuevo perfil LINODE.....	51
Figura #15 Dashboard LINODE	52
Figura #16 Acceso Remoto LINODE	53
Figura #17 Consola Acceso Remoto al servidor por SSH	53
Figura #18 Instalación base de datos MongoDB.....	54
Figura #19 Instalación del servidor web NGINX.....	54
Figura #20 Copiando archivos del sistema web al servidor en la nube.....	55
Figura #21 Corriendo estructura de base de datos.....	55

CAPÍTULO I

INTRODUCCIÓN

1.1 El Problema de Investigación

1.1.1 Planteamiento del Problema

La demanda de los servicios que presta Especigest ha crecido debido a que el gobierno ecuatoriano estableció en el Art. 326, numeral 5 de la Constitución del Ecuador y en el Decreto Ejecutivo 2393 que las empresas pequeñas, medianas y grandes deben velar por la salud y seguridad de sus empleados; además de estar sujetas a cumplir obligaciones, derechos y deberes en cuanto a la prevención de riesgos laborales.

La normativa vigente ha provocado que el volumen de información que maneja la empresa Especigest en los procesos comerciales con sus clientes crezca rápidamente y su administración sea complicada y difícil de controlar.

La administración de este tipo de información considerada crítica y delicada, es un proceso que se debe llevar a cabo con mucha precisión y seguridad; las herramientas de ofimática actuales de trabajo (Microsoft Word, Excel, Power Point) no pueden garantizar la integridad de los datos.

1.1.2 Objetivo General

Diseñar, desarrollar e implementar un sistema informático web para la gestión y administración de la información en los procesos comerciales de la empresa Especigest.

1.1.3 Objetivos Específicos

- Realizar el levantamiento de requerimientos e identificar los puntos críticos en los cuales se basa el sistema.
- Diferenciar las funciones y actividades que realiza los miembros del equipo para establecer los perfiles adecuados de acceso.
- Estructurar la base de datos que permita un manejo dinámico y de alto crecimiento de la información.
- Diseñar y desarrollar el sistema informático web para la administración de información crítica con las normas y principios de calidad de software y código limpio.
- Implementar el servidor web con las características y configuraciones necesarias para la instalación del sistema.
- Realizar las pruebas necesarias para comprobar el funcionamiento y compatibilidad del sistema.

1.1.4 Justificación

Especigest presta un servicio importante en el mercado ecuatoriano; no solo asesora, sino que también provee de información importante sobre riesgos laborales a sus clientes, para que tomen medidas oportunas que permita cuidar la salud y seguridad de sus trabajadores.

Para cumplir este objetivo y en vista del crecimiento de información. Especigest tiene la necesidad de contar con un sistema informático que permita manejar los datos de forma óptima, segura y automatizada.

Con el desarrollo del sistema se consiguió optimizar la fuerza laboral pues ahora se dedican a tareas específicas, redujo el tiempo para cumplir una meta, aumentó su productividad y permitió atender a más clientes con un servicio eficiente, lo que generó mas ingresos.

La investigación contó con el apoyo del personal de los departamentos de sistemas, finanzas y ventas, que están conscientes que el sistema les ayuda a mejorar su rendimiento para cumplir con las exigencias de los clientes.

El sistema informático fue desarrollado por módulos, cuenta con una estructura que permite la generación de perfiles de usuarios para controlar los accesos y permisos asegurando la integridad de la información.

Los módulos del sistema son los siguientes:

- **Roles de Usuario:** Este módulo permite crear roles de acceso para los usuarios del sistema.

- **Clientes:** Este módulo permite ingresar, mantener y actualizar la información de todos los clientes.
- **Proveedores:** Este módulo permite ingresar y dar mantenimiento a la información de los proveedores.
- **Productos:** Permite el ingreso, mantenimiento y actualización de la información acerca de sus productos.
- **Inventarios:** Permite obtener un resumen de entradas y salidas que han tenido los productos.
- **Compras:** Este módulo está relacionado con el módulo de proveedores y productos, en el cual se registran las compras realizadas para aumentar el inventario y registrar gastos.
- **Ventas:** Este módulo está relacionado con el módulo de proveedores y productos, en el cual se registran las ventas realizadas.

1.2 Marco Teórico

1.2.1 Ingeniería de software

Barry Boehm (1976) señala que la ingeniería de software es “la aplicación práctica del conocimiento científico al diseño y construcción de programas de computadora a la documentación asociada requerida para desarrollar, operar y mantenerlos”.

Pete McBreen (2010) afirma que “el termino ingeniería implica nivel de rigor y de pruebas mucho mayores que lo habitual en los desarrollos actuales”.

Ingeniería del Software, es el término que utilizó Fritz Bauer en la primera conferencia sobre desarrollo de software patrocinada por el Comité de Ciencia de la OTAN celebrada en Garmisch (Alemania), en octubre de 1968.

Puede definirse según Alan Davis como “la aplicación inteligente de principios probados, técnicas, lenguajes y herramientas para la creación y mantenimiento, dentro de un coste razonable, de software que satisfaga las necesidades de los usuarios”.

1.2.2 Desarrollo de software

Desarrollar un software significa construirlo mediante su descripción. Está es una muy buena razón para considerar el desarrollo de software como una ingeniería. En un nivel más general, la relación existente entre un software y su entorno es clara ya que el software es introducido en el mundo de modo de provocar ciertos efectos en el mismo. (Yanvary, 2006).

1.2.3 Metodologías ágiles para el desarrollo de software

Para desarrollar sistemas informáticos es necesario seguir una metodología adecuada y moderna que permita ahorrar recursos y agilizar el tiempo de desarrollo.

Una metodología en el desarrollo de software busca realizar, gestionar y administrar un proyecto de tal forma que se tengan altas probabilidades de éxito. Lo que se busca al guiarse con una metodología es prolijidad, corrección, y control en cada etapa del desarrollo de un programa. (Rivas, 2012).

Las metodologías ágiles son sin duda uno de los temas recientes en ingeniería de software que están acaparando gran interés. En la comunidad de la ingeniería del software, se está viviendo con intensidad un debate abierto entre los partidarios de las metodologías tradicionales (referidas peyorativamente como "metodologías pesadas") y aquellos que apoyan las ideas emanadas del "Manifiesto Ágil". (Letelier, 2006)

Existen varias metodologías que se especializan en aspectos específicos y que tienen sus propias características. A continuación se resumen estas metodologías:

- **SCRUM:** Desarrollada por Ken Schwaber, Jeff Sutherland y Mike Beedle. Define una marco para la gestión de proyectos, que se ha utilizado con éxito durante los últimos 10 años. Está especialmente indicada para proyectos con un rápido cambio de requisitos. Sus principales características se pueden resumir en dos. El desarrollo de software se realiza mediante iteraciones, denominadas sprints, con una duración de 30 días. El resultado de cada sprint es un incremento ejecutable que se muestra al cliente. La segunda característica importante son las reuniones a lo largo del proyecto. Éstas son las verdaderas protagonistas, especialmente la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración. (Letelier, 2006)
- **Crystal Methodologies:** Se trata de un conjunto de metodologías para el desarrollo de software caracterizadas por estar centradas en las personas que componen el equipo (de ellas depende el éxito del proyecto) y la reducción al máximo del número de artefactos producidos. Han sido desarrolladas por Alistair Cockburn. El desarrollo de software se considera un juego cooperativo de invención y comunicación, limitado por los recursos a utilizar. El equipo de desarrollo es un factor clave, por lo que se deben invertir esfuerzos en mejorar

sus habilidades y destrezas, así como tener políticas de trabajo en equipo definidas. Estas políticas dependerán del tamaño del equipo, estableciéndose una clasificación por colores, por ejemplo Crystal Clear (3 a 8 miembros) y Crystal Orange (25 a 50 miembros). (Letelier, 2006)

- **Dynamic Systems Development Method⁶ (DSDM):** Define el marco para desarrollar un proceso de producción de software. Nace en 1994 con el objetivo de crear una metodología RAD unificada. Sus principales características son: es un proceso iterativo e incremental y el equipo de desarrollo y el usuario trabajan juntos. Propone cinco fases: estudio viabilidad, estudio del negocio, modelado funcional, diseño y construcción, y finalmente implementación. Las tres últimas son iterativas, además de existir realimentación a todas las fases. (Letelier, 2006)
- **Adaptive Software Development (ASD):** Su impulsor es Jim Highsmith. Sus principales características son: iterativo, orientado a los componentes software más que a las tareas y tolerante a los cambios. El ciclo de vida que propone tiene tres fases esenciales: especulación, colaboración y aprendizaje. En la primera de ellas se inicia el proyecto y se planifican las características del software; en la segunda desarrollan las características y finalmente en la tercera se revisa su calidad, y se entrega al cliente. La revisión de los componentes sirve para aprender de los errores y volver a iniciar el ciclo de desarrollo. (Letelier, 2006)

- **Feature-Driven Development⁸ (FDD):** Define un proceso iterativo que consta de 5 pasos. Las iteraciones son cortas (hasta 2 semanas). Se centra en las fases de diseño e implementación del sistema partiendo de una lista de características que debe reunir el software. Sus impulsores son Jeff De Luca y Peter Coad. (Letelier, 2006)
- **Lean Development (LD):** Definida por Bob Charette's a partir de su experiencia en proyectos con la industria japonesa del automóvil en los años 80 y utilizada en numerosos proyectos de telecomunicaciones en Europa. En LD, los cambios se consideran riesgos, pero si se manejan adecuadamente se pueden convertir en oportunidades que mejoren la productividad del cliente. Su principal característica es introducir un mecanismo para implementar dichos cambios. (Letelier, 2006)
- **Programación Extrema (XP):** Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. (Letelier, 2006)

1.2.4 Herramientas para el desarrollo de software

El desarrollo de Software es un proceso complejo y a menudo difícil que requiere la síntesis de muchos sistemas. Desde el modelado y diseño hasta el código, administración del proyecto, pruebas, despliegue, administración de cambios y más. Para simplificar y ayudar en el desarrollo de software existen herramientas para cada necesidad:

- **Herramientas CASE:** Son herramientas que ayudan en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, cálculo de costos, implementación, compilación automática, documentación, o detección de errores entre otras. (Gómez, 2003)
- **Herramientas de modelado:** El labor del análisis involucra el modelado del sistema que desea el usuario. Los modelos de análisis de sistema son representaciones abstractas de lo que al final será una combinación de hardware y software. (Slideshare, 2014)
- **Editores/IDEs:** Un IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación, o sea, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Los IDEs pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes. (EcuRed, 2012)

1.2.5 Base de datos

“Desde el punto de vista informático, la base de datos es un sistema formado por un conjunto de datos almacenados en discos que permiten el acceso directo a ellos y un conjunto de programas que manipulen ese conjunto de datos”. (Valdés, 2007)

Cada base de datos se compone de una o más tablas que guarda un conjunto de datos. Cada tabla tiene una o más columnas y filas. Las columnas guardan una parte de la información sobre cada elemento en la tabla, cada fila de la tabla conforma un registro. (Valdés, 2007)

“Una base de datos proporciona a los usuarios el acceso a la información, que pueden visualizar, ingresar o actualizar, en concordancia con los derechos de acceso que se les haya otorgado”. (Kioskea, 2015)

“Una base de datos es un “almacén” que nos permite guardar grandes cantidades de información de forma organizada para que luego podamos encontrar y utilizar fácilmente” (Pérez, 2007)

A continuación se resumen los diferentes tipos de bases de datos que existen:

Base de datos Relacional

Una base de datos relacional es la que cumple con las normal de un modelo relacional que está basado en diagramas o modelos entidad-relación (denominado por su siglas, ERD “Diagram Entity relationship”) es una herramienta para el modelado de datos de un sistema de información. Estos modelos expresan entidades relevantes para un sistema de información, sus inter-relaciones y propiedades. (Pérez, 2007)

Base de datos No Relacionales

“Es un término utilizado para describir un subconjunto de bases de datos que difiere en varios modos de bases de datos tradicionales (RDBMS). No tienen schemas, no permiten JOINS, no intentan garantizar ACID y escalan horizontalmente. El término fue acuñado en 1998 por Carlo Strozzi y resucitado en 2009 por Eric Evans, el propio Evans sugiere mejor referirse a esta familia de BBDD de nueva generación como Big Data”. (López, 2012)

El modelo relacional, con más de 40 años sigue vigente en la actualidad y es la base de la mayoría de aplicaciones que acceden a una base de datos. Sin embargo, no es el único modelo existente, ni el único en uso. En los últimos tiempos han cobrado importancia las bases de datos que no cumplen con los principios expresados por Codd en 1970. (Mesas, 2014)

1.3 Marco conceptual

1.3.1 Sistema Informático

Un Sistema Informático (SI) se puede definir como todo el conjunto de partes que funcionan relacionándose entre sí, para conseguir un objetivo preciso.

Las partes de un sistema informático son:

- **Hardware:** está formado por los dispositivos electrónicos y mecánicos que realizan los cálculos y el manejo de la información. Todo lo que se puede tocar físicamente. (Alegsa, 2009)

- **Software:** se trata de un conjunto de programas, instrucciones y reglas informáticas que permiten ejecutar distintas tareas en un computador. (Alegsa, 2009)
- **Personal:** está compuesto tanto por los usuarios que interactúan con los equipos hardware, como por aquellos que desarrollan y mantienen el software (para que esa interacción sea posible). (Alegsa, 2009)
- **Información descriptiva:** es el conjunto de manuales técnicos o de usuario, formularios, documentación de procedimientos o cualquier soporte que dé instrucciones, sobre el uso del sistema informático. (Alegsa, 2009)

1.3.2 Aplicación Web

“Una aplicación web es cualquier aplicación que es accedida vía web por una red como internet o una intranet”. (Alegsa, 2009)

“En General el término también se utiliza para designar aquellos programas informáticos que son ejecutados en el entorno del navegador o codificado con algún lenguaje soportado por el navegador”. (Alegsa, 2009)

1.3.3 HTML

“HTML es el lenguaje con el que se definen las páginas web. Básicamente se trata de un conjunto de etiquetas que sirven para definir el texto y otros elementos que compondrán una página web”. (Álvarez, 2001)

1.3.4 CSS

“El CSS no es un lenguaje de programación como JavaScript ni tampoco es un lenguaje de etiquetas como HTML; de hecho, no hay nada con lo que se le pueda comparar. Las tecnologías que definían las interfaces antes del desarrollo de la web mezclaban siempre la presentación y la estructura. Sin embargo, en un entorno que cambia tan a menudo como la web, ésta no es una manera muy inteligente de hacer las cosas, y por ello se inventó el CSS”. (Heilmann, 2008)

Mientras que el HTML estructura el documento e indica a los navegadores cuál es la función de un elemento concreto, el CSS da instrucciones al navegador sobre cómo debe mostrar un elemento concreto: estilo, espaciado y posición. Si el HTML son los puntales y los ladrillos que forman la estructura de una casa, el CSS es el yeso y la pintura que la decoran. (Heilmann, 2008)

1.3.5 Bootstrap

Bootstrap, es un framework originalmente creado por Twitter, que permite crear interfaces web con CSS y JavaScript, cuya particularidad es la de adaptar la interfaz del sitio web al tamaño del dispositivo en que se visualice. Es decir, el sitio web se adapta automáticamente al tamaño de una PC, una Tablet u otro dispositivo. Esta técnica de diseño y desarrollo se conoce como “responsive design” o diseño adaptativo. (Solis, 2014)

1.3.6 PHP

“PHP (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML”. (The PHP Group, 2015)

1.3.7 Laravel

“Laravel es un nuevo y poderoso Framework PHP desarrollado por Taylor Otwell, que promete llevar al lenguaje PHP a un nuevo nivel”. (Patricio, 2013)

Laravel, propone una forma de desarrollar aplicaciones web de un modo mucho más ágil. Por ejemplo, en Laravel podemos usar el patrón de diseño MVC (Modelo-Vista-Controlador), donde al igual que otros frameworks PHP, el controlador es programado como una clase. (Patricio, 2013)

Por lo tanto, un Controlador es una clase PHP que dispone de métodos públicos que son el punto de entrada final de una petición HTTP (Request PHP) a nuestra aplicación. Pero, Laravel propone además una forma distinta y más directa de responder a la solicitud HTTP, que veremos enseguida. (Patricio, 2013)

1.3.8 Javascript

“Javascript es un lenguaje con muchas posibilidades, utilizado para crear pequeños programas que luego son insertados en una página web y en programas más grandes, orientados a objetos mucho más complejos. Con Javascript podemos crear diferentes efectos e interactuar con nuestros usuarios.” (Valdés, 2007)

Este lenguaje posee varias características, entre ellas se puede mencionar que es un lenguaje basado en acciones que posee menos restricciones. Gran parte de la programación en este lenguaje está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas, utilización de teclas, cargas de páginas entre otros. (Valdés, 2007)

“Es necesario resaltar que hay dos tipos de JavaScript: por un lado está el que se ejecuta en el cliente, este es el Javascript propiamente dicho, aunque técnicamente

se denomina Navigator JavaScript. Pero también existe un Javascript que se ejecuta en el servidor, es más reciente y se denomina LiveWire Javascript”. (Valdés, 2007)

1.3.9 Angularjs

“AngularJS es un framework MVC de JavaScript para el Desarrollo Web Front End que permite crear aplicaciones SPA (Single-Page Applications). Entra dentro de la familia de frameworks como BackboneJS o EmberJS”. (Azaustre, 2013)

1.3.10 MongoDB

“MongoDB es una base de datos orientada a documentos. Esto quiere decir que en lugar de guardar los datos en registros, guarda los datos en documentos. Estos documentos son almacenados en BSON, que es una representación binaria de JSON”. (Fernández, 2014)

“Una de las diferencias más importantes con respecto a las bases de datos relacionales, es que no es necesario seguir un esquema. Los documentos de una misma colección concepto similar a una tabla de una base de datos relacional, pueden tener esquemas diferentes” (Fernández, 2014)

“MongoDB está escrito en C++, aunque las consultas se hacen pasando objetos JSON como parámetro. Es algo bastante lógico, dado que los propios documentos se almacenan en BSON”. (Fernández, 2014)

MongoDB viene de serie con una consola desde la que podemos ejecutar los distintos comandos. Esta consola está construida sobre JavaScript, por lo que las consultas se realizan utilizando ese lenguaje. Además de las funciones de MongoDB, se puede utilizar muchas de las funciones propias de JavaScript. En la consola también podemos definir variables, funciones o utilizar bucles. (Fernández, 2014)

CAPITULO II

MÉTODO

2.1 Análisis

2.1.1 Estudio preliminar

Para el levantamiento de requerimientos se llevó acabo reuniones con el personal de los departamentos de sistemas, ventas y de finanzas que laboran en la empresa con el fin de determinar las características y funcionalidad con las que cuenta el sistema.

Los temas principales de las reuniones realizadas fueron en base a los módulos que conforman el sistema siguiendo el lineamiento propuesto por la metodología ágil Scrum. Estas reuniones fueron las siguientes:

Reunión #1 Módulo de usuarios

En esta reunión se hablo con el personal de ventas y finanzas, para determinar que permisos se necesitaba para cada uno de los usuarios del sistema. Se pudo identificar 4 tipos de usuarios:

- Root: Acceso a todo el sistema.
- Admin: Acceso a todo el sistema con excepción de la configuración

- Ventas: Acceso solo al módulo de ventas y de inventarios
- Compras: Acceso solo al módulo de compras y de inventarios
- Bodegas: Acceso al modulo de Inventarios.

Reunión #2 Módulo de clientes

En esta reunión se trato con el personal de ventas para identificar la información que Especigest requiere de sus clientes. Se pudo identificar los siguientes datos:

- Cédula / Ruc
- Nombres
- Apellidos
- Teléfono
- Celular
- Correo
- Dirección

Además se pudo conocer que Especigest contaba con una base de datos de sus clientes en formato Excel con lo que nació la necesidad de desarrollar una subida masiva de datos.

Reunión #3 Módulo de proveedores

En esta reunión se trato con el personal de compras para conocer la información que necesitan de sus proveedores. Se pudo identificar los siguientes datos:

- Cédula / Ruc

- Nombres
- Apellidos
- Razón Social
- Fax
- Teléfonos
- Celulares
- Correos
- Dirección

Reunión #4 Módulo de productos

En esta reunión se trato con el personal de ventas, compras y bodega para determinar que datos se deben guardar de los productos y servicios que ofrece Especigest. Se pudo identificar los siguientes datos:

- Tipo
- Categoría
- Código
- Marca
- Fecha de expiración
- Unidad de medida
- Stock
- Descripción
- Precio de venta
- Costo unitario
- IVA

- Descuento
- Si es gasto

Además, se pudo identificar la necesidad de contar con la creación de categorías en forma jerárquica para organizar sus productos, y el desarrollo de un reporte para mostrar las entradas y salidas de inventarios .

Reunión #4 Módulo de ventas

En esta reunión se trato con el personal de ventas para determinar cual es el flujo de trabajo que tienen con sus clientes. Se pudo identificar los siguientes procesos:

- Oferta de venta
- Pedido de cliente
- Entrega de mercadería
- Factura Cliente
- Resumen de Ventas

Reunión #5 Módulo de compras

En esta reunión se trato con el personal de compras para determinar cual es el flujo de trabajo que tienen con sus proveedores. Se pudo identificar los siguientes procesos:

- Solicitud de compra
- Aprobación de solicitud
- Pedido
- Ingreso de mercaderías

- Factura proveedor
- Resumen de compras

2.1.2 Descripción General del Sistema

El sistema web para la gestión y administración de procesos en la empresa Especigest ayuda a controlar la lista de clientes y de proveedores con los que la empresa cuenta. Permite gestionar de forma automática el inventario de sus productos, los ingresos ocasionados por la prestación de servicios y los gastos que la empresa produce con sus actividades comerciales.

Además el sistema permite controlar los permisos de los usuarios para garantizar la organización del personal y la integridad de los datos.

2.2 Estudio de factibilidad

2.2.1 Factibilidad Operativa

Todos los departamentos de la empresa Especigest cuentan con la capacitación adecuada para poder operar el sistema.

El departamento de sistemas puede: monitorear la actividad que tiene el servidor en el que esta operando el sistema, consultar el archivo de registro de errores del sistema y de la base de datos, respaldar la información para impedir la pérdida de datos.

El departamento de ventas tiene el conocimiento para manipular el módulo de ventas, ingresar facturas y consultar el inventario de productos.

El departamento administrativo puede realizar la configuración de los parámetros generales del sistema, además de la administración de los perfiles de usuario y la consulta de reportes generales.

Toda la información de la empresa está almacenada en los servidores de LINODE; servicio en la nube que se contrato para garantizar la integridad y seguridad de los datos.

2.2.2 Factibilidad Técnica

El desarrollo del sistema web esta basado en las siguientes plataformas: el lenguaje de programación para el Back-End es PHP, para el Front-End Html y Javascript, para la gestión de base de datos MONGODB; todas las tecnologías utilizadas son libres y cuentan con una gran comunidad internacional de apoyo en donde se puede encontrar la ayuda y el soporte técnico necesario para su correcto funcionamiento.

Especigest contaba con la infraestructura necesaria para la implementación del sistema, ya que el departamento de sistema estaba preparado técnicamente para solventar las necesidades del personal.

A continuación se describen las características de la infraestructura con la que cuenta:

Servidor

Servicio en la nube contratado con LINODE, que es lugar donde está almacenado el sistema y la base de datos sus características son

- Procesador 1 Core
- Ram 1 GB
- Almacenamiento 24GB SSD
- Transferencia 2TB
- Capacidad de entrada de Red 40Gbps
- Capacidad de salida de Red 125 Mbps

Estaciones de Trabajo:

Especigest cuenta con 10 computadores personales, los mismos que están totalmente satisfactorios para la ejecución del sistema web, sus características son las siguientes:

- Procesador Intel Core i3
- RAM 4gb
- Disco duro 500GB
- Tarjeta de red base 10/10/1000

Infraestructura de Red

La red en la que operan esta basada en tecnología Ethernet, funciona bajo el estándar base 100 Mbps, y cuenta con un internet CLARO de 10 Mbps.

Equipo para el desarrollo del sistema

Una de las estaciones de trabajo fue utilizada para el desarrollo del sistema, cuenta con las siguientes características :

Tabla #1 Características del equipo para el desarrollo del sistema

Elaborado por: Bryan Guamba

Nombre	Características
CPU	Intel Core i3
RAM	4GB
DISCO DURO	500gb
TARJETA DE VIDEO	1 GB

El software necesario para el desarrollo del sistema web es el siguiente:

Tabla #2 Software necesario para el desarrollo del sistema

Elaborado por: Bryan Guamba

Nombre	Características
Sublime	IDE de programación versión FREE
Photoshop	Software de edición de imágenes
Robomongo	Software libre para la gestión de base de datos MONGODB.
Linux Debian	Sistema operativo libre para el desarrollo de aplicaciones web.

2.2.3 Factibilidad Económica

A continuación se detalla la inversión para la implementación, y desarrollo del sistema y que Especigest los asumió en su totalidad:

Tabla #3 Software necesario para el desarrollo del sistema

Elaborado por: Bryan Guamba

Descripción	Tiempo mínimo	Costo
Servidor en la nube LINODE	1 año	\$120.00
Internet Claro	1 mes	\$75.00
Licencia Adobe Fotografía	1 año	\$120.00
Licencia Sublime	Indefinido	\$ 70.00
Laptop para desarrollo		\$1,000.00
TOTAL		\$1,385.00

2.3 Diseño

2.3.1 Solución Técnica

El sistema web cuenta con funciones específicas para cada flujo de compra y venta de los productos y servicios que ofrece Especigest; para garantizar que estos procesos se cumplan cuenta con un administrador de perfiles que permite controlar que los usuarios solo puedan hacer las actividades que están autorizados.

2.3.2 Diagrama de Casos de Uso

Figura #1 Diagrama de Caso de Uso Root

Elaborado por: Bryan Guamba

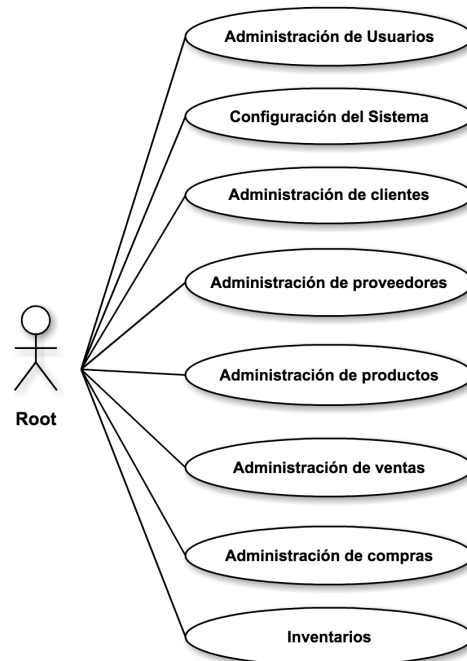


Figura #2 Diagrama de Caso de Uso Administrador

Elaborado por: Bryan Guamba

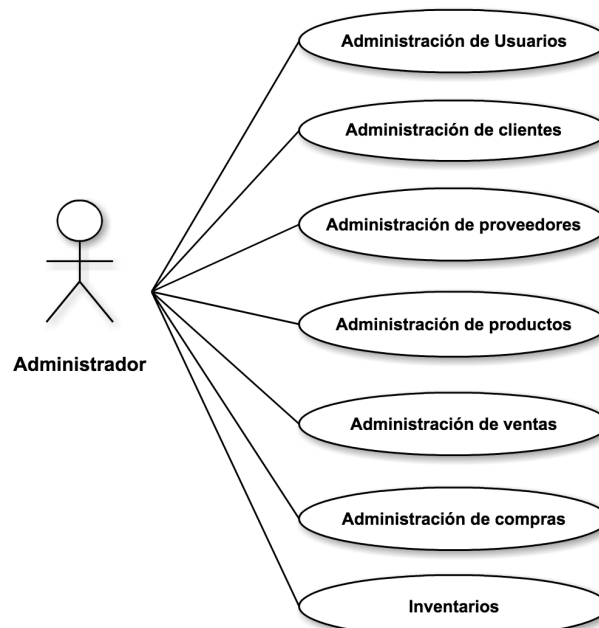


Figura #3 Diagrama de Caso de Uso Ventas

Elaborado por: Bryan Guamba

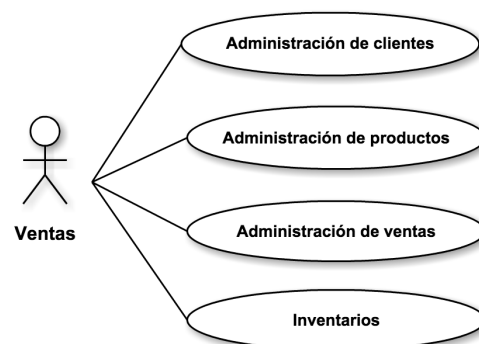
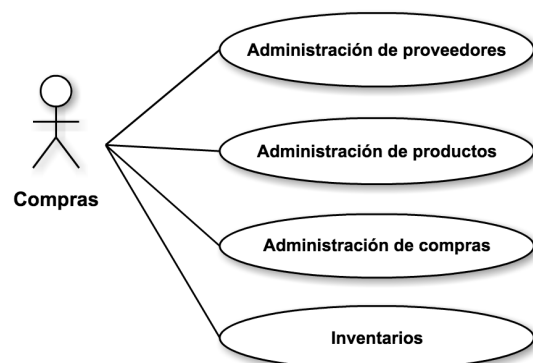


Figura #4 Diagrama de Caso de Uso Compras

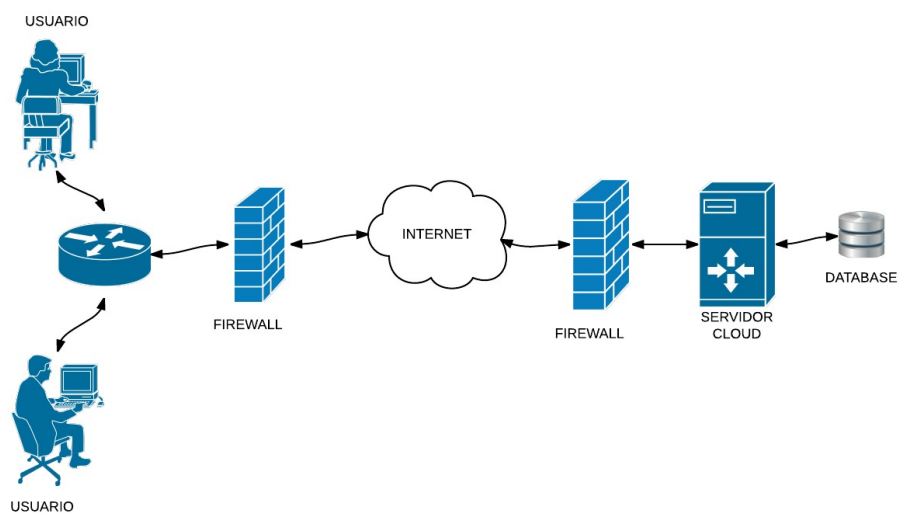
Elaborado por: Bryan Guamba



2.3.3 Diagrama de arquitectura

Figura #5 Diagrama de Arquitectura

Elaborado por: Bryan Guamba



2.3.4 Diagrama de clase

Figura #6 Diagrama de clases de controladores

Elaborado por: Bryan Guamba

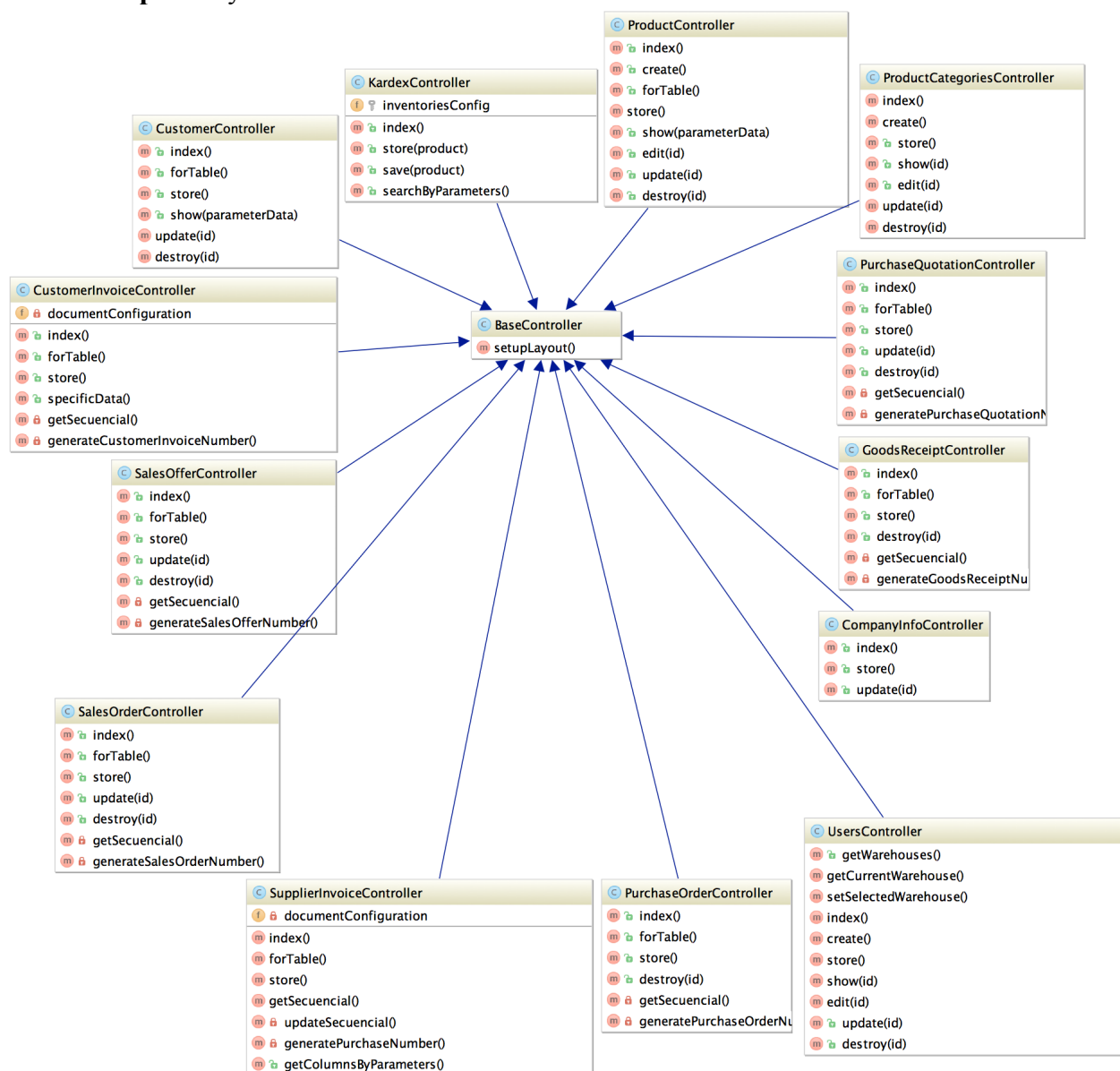
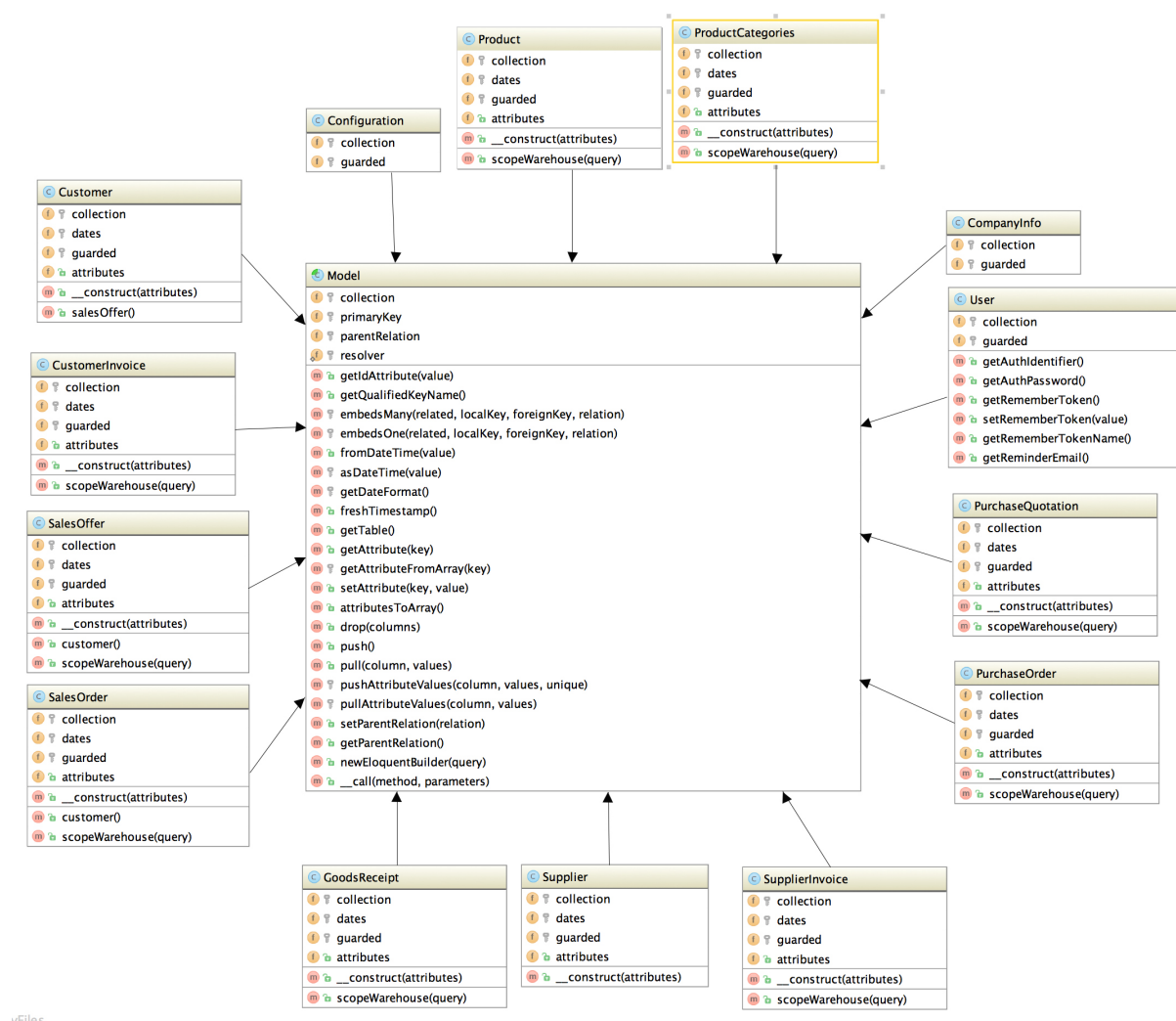


Figura #7 Diagrama de clases de modelos
Elaborado por: Bryan Guamba



2.3.5 Diagrama de base de datos

Para la administración de datos se utilizó MONGODB, que es una base de datos No SQL basada en documentos.

Las principales características de esta base de datos es la eliminación de relaciones en todas las entidades mientras sea posible, además de ser libre de esquemas es decir que no se rige en un modelo inicial sino que se construye su estructura de acuerdo al desarrollo y evolución del sistema, el esquema final de la base de datos es el siguiente:

Figura #8 Diagrama base físico de la base de datos

Elaborado por: Bryan Guamba

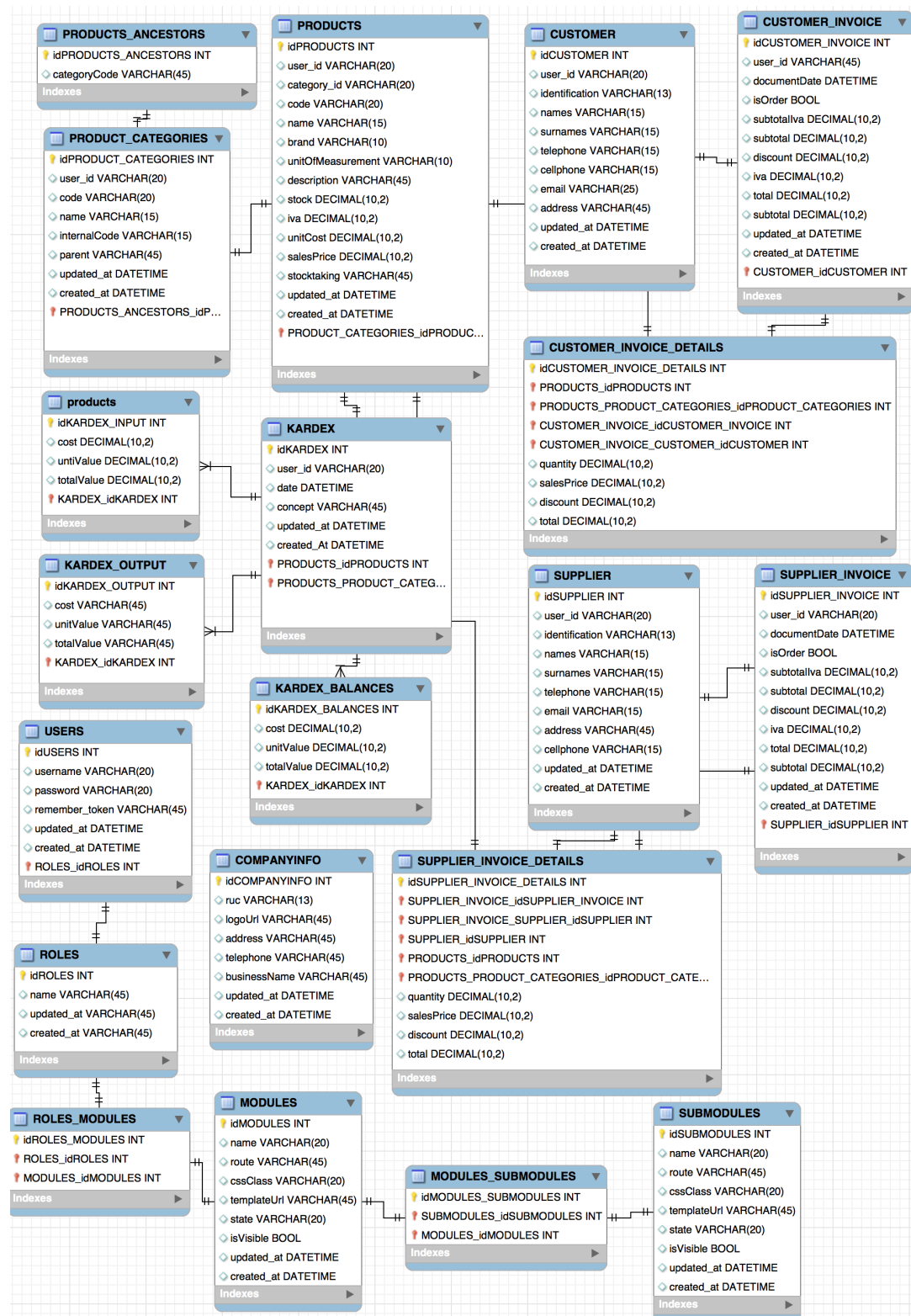
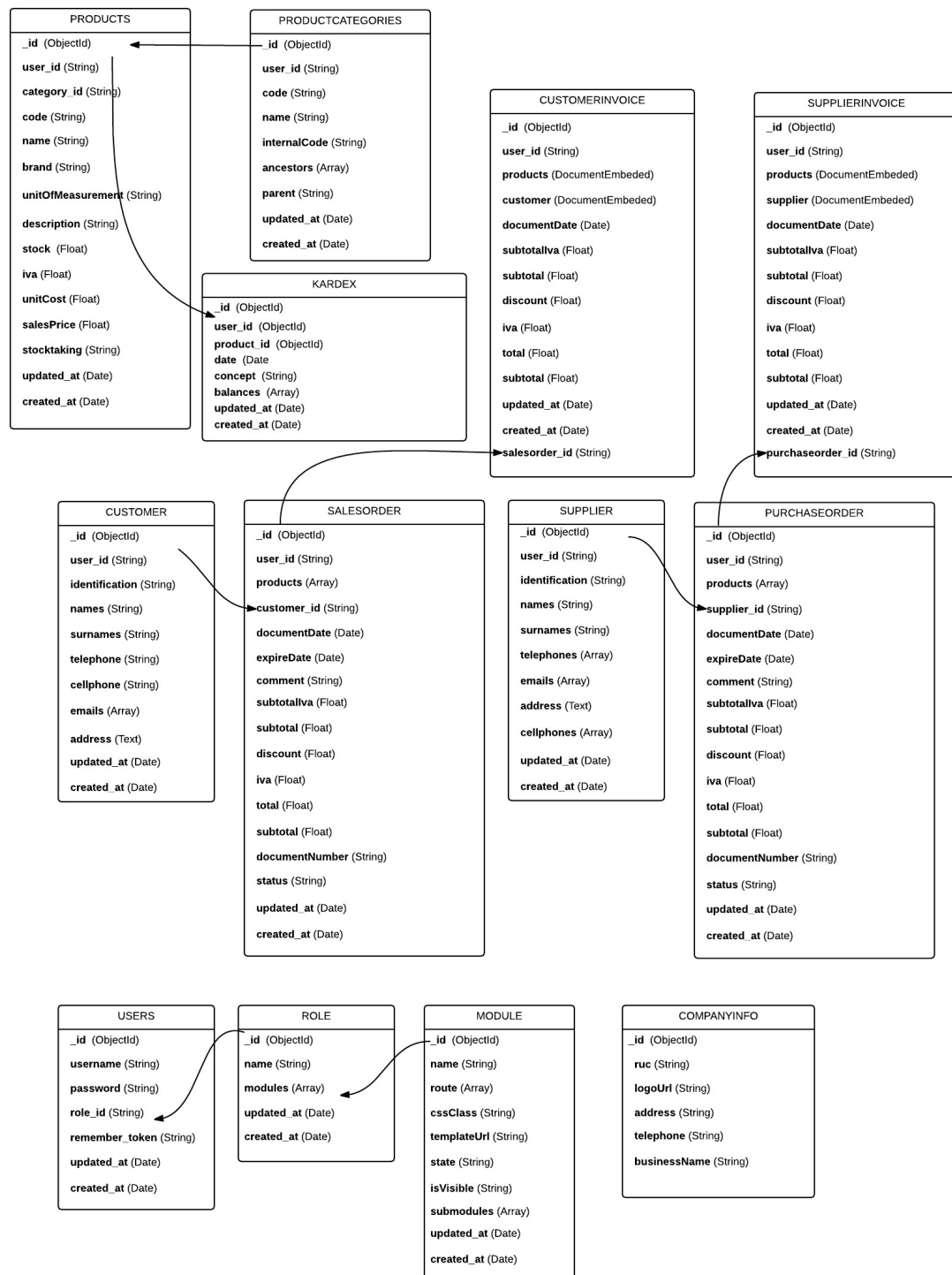


Figura #9 Diagrama de la estructura final de la base de datos
Elaborado por: Bryan Guamba



2.4 Arquitectura del ambiente de desarrollo

El ambiente de pruebas se ejecutó en un computador con las siguientes características

Tabla #4 Características del equipo para el ambiente de pruebas

Elaborado por: Bryan Guamba

Nombre	Características
CPU	Intel Core i3
RAM	4GB
DISCO DURO	500gb
TARJETA DE VIDEO	1 GB
SISTEMA OPERATIVO	Linux Debian

Además, para montar el servidor de desarrollo se utilizó la herramienta de código ARITSAN que forma parte del framework LARAVEL, que permite ejecutar un ambiente web de pruebas con el comando “*php artisan serve*”

2.4.1 Arquitectura del ambiente en producción

Para el ambiente de producción se optó por contratar un servicio en la nube con LINODE, por las siguientes razones:

- Total control sobre el servidor
- Soporte 24/7
- Programación de Backups automáticos
- Almacenamiento SSD
- Imágenes de sistemas operativos listas al click
- Soporte para SSL
- Infraestructura Antihackeo

- Geo redundancia (7 Datacenters en 3 Continentes)
- 40Gbit en red con múltiples niveles y redundancia
- Procesadores Intel E5
- Panel de Control
- Escalabilidad automática
- Posibilidad de creación de snapshots para recuperación

2.4.2 Plataforma

La plataforma en la que el sistema funciona es un sistema operativo Linux, Robomongo como administrador de base de datos, y NGINX como servidor web.

Tabla #5 Plataforma para el sistema

Elaborado por: Bryan Guamba

Nombre	Características
SISTEMA OPERATIVO	Linux Debian
SERVIDOR WEB	NGINX
LENGUAJES DE PRGROMACIÓN	PHP v5.5
	HTML5
	ANGULAR JS 1.3.X
BASE DE DATOS	MONGODB 2.6

2.4.3 Esquema de distribución del sistema Web

1	2
3	4

Descripción de la distribución

1. Logotipo
2. Información de la empresa
3. Menú Principal
4. Contenido

2.4.4 Descripción de las interfaces

El sistema se desarrolló con estándares de diseño y experiencia de usuario modernos como el “Responsive Design” y el “Single page application”; La primera pantalla es del ingreso del sistema en el cual se requiere escribir un usuario y contraseña, y el sistema validará que sean correctos y dependiendo de sus configuración mostrará el contenido correspondiente.

Figura #10 Inicio de sesión del sistema

Elaborado por: Bryan Guamba

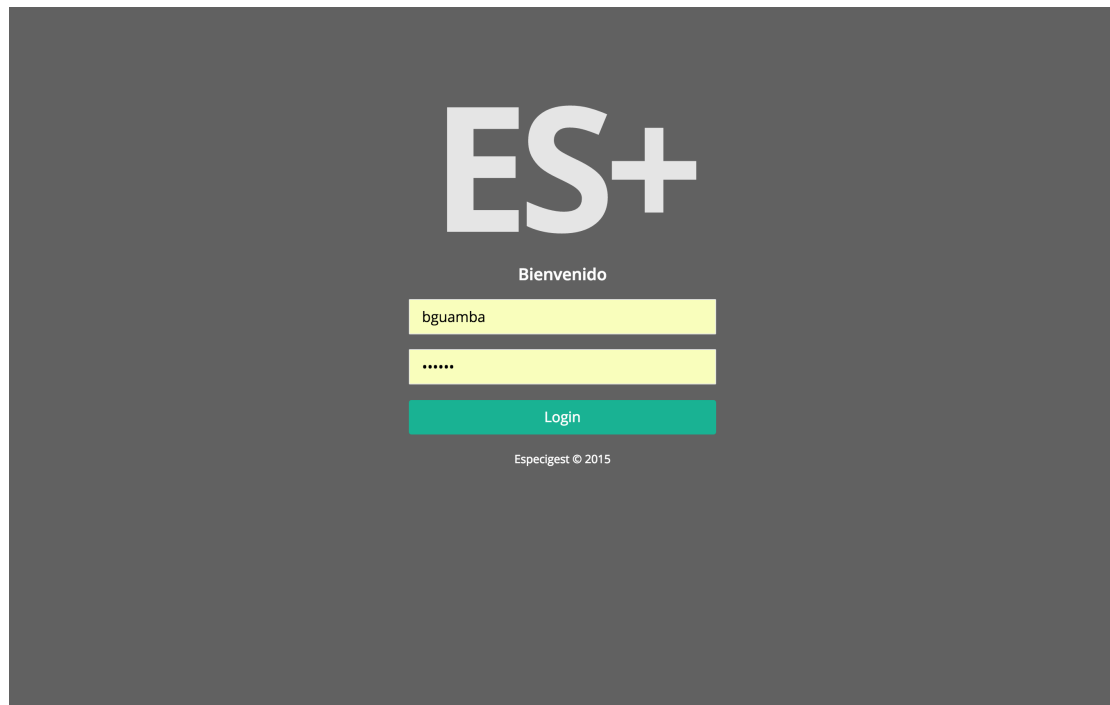


Figura #11 Inicio de sesión fallido

Elaborado por: Bryan Guamba

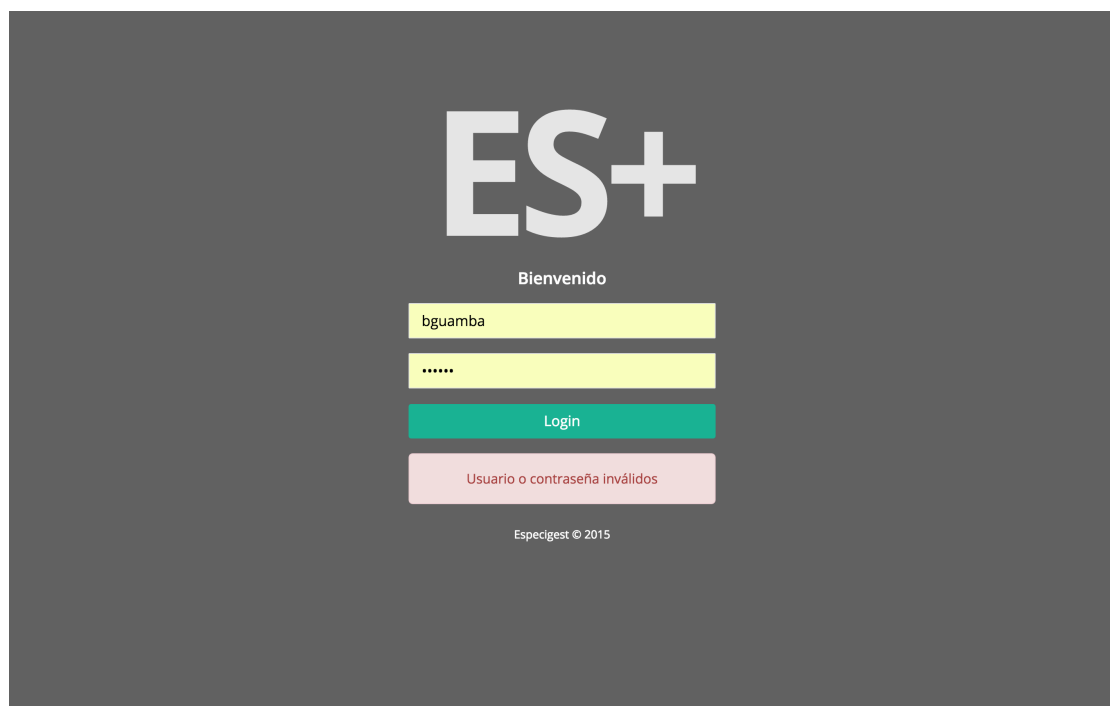


Figura #12 Dashboard del sistema

Elaborado por: Bryan Guamba

The screenshot displays the ESPECIGEST web application interface. At the top, there is a dark header with the ESPECIGEST logo on the left and a 'Salir' button on the right. Below the header is a sidebar menu with icons and labels for 'Inventarios', 'Ventas', 'Compras', 'Clientes', 'Proveedores', and 'Productos'. The 'Clientes' menu item is highlighted in green. The main content area is titled 'Nuevo Cliente' and contains a form for adding a new client. The form is divided into two sections: 'Información General' and 'Información de Contacto'. The 'Información General' section includes fields for 'Cédula/RUC', 'Nombres', 'Apellidos', and 'Teléfono'. The 'Información de Contacto' section includes fields for 'Celular', 'Correo Electrónico', and 'Dirección'. At the bottom of the form are 'Guardar' and 'Limpiar' buttons. The background of the main content area is a dark gray gradient.

2.5 Administración y Seguridad

2.5.1 Acceso a niveles de usuario

El sistema cuenta con un módulo para la administración de perfiles de usuarios, para limitar las acciones y restringir el acceso no autorizado. Se crearon 3 perfiles de usuario:

1. **Administrador:** Tiene el control de todos los módulos del sistema incluyendo su configuración:
 - Inventarios
 - Clientes
 - Proveedores
 - Compras
 - Ventas

- Productos
 - Configuración
2. **Ventas :** Tiene acceso solo al registro de pedidos, ventas, y resumen de ventas, además del inventario de productos
 3. **Compras:** Tiene acceso solo al registro de ordenes, compras, y resumen de ventas, además del inventario de productos.

2.5.2 Auditoría

Todo el sistema esta estructurado para saber que usuario, y cuando realizó una inserción, modificación y/o eliminación de registros en la base de datos.

Cada registro en la base de datos guarda tres campos UPDATE_AT, CREATED_AT y USER_ID en donde se registra la fecha, hora y el usuario que realizó la acción.

Además que la recuperación de datos por eliminación equivocada esta a un paso, ya que el registro no se elimina, solo se crea un campo extra llamado DELETE_AT, y al utilizar el sistema todos los registros que tengan este campo no serán tomados en cuenta.

El motor de base de datos MONGODB, cuenta con un log de acciones realizado a lo largo del uso de la base de datos en donde se almacenan todas las transacciones realizadas, las conexiones y los errores.

CAPITULO III

RESULTADOS

3.1 CONSTRUCCIÓN

3.1.1 Generalidades

El sistema informático web para la gestión y administración de procesos en la empresa Especigest fue desarrollado con tecnología libre y moderna en cuanto al desarrollo informático se refiere.

En el Back-End fue construido con el lenguaje de programación PHP, y el framework LARAVEL que ayuda en la estructuración del código, separando la lógica, de los modelos de interacción con la base de datos, además de aportar una gran cantidad de utilidades que ayudan a la seguridad, al seguimiento y mantenimiento del código.

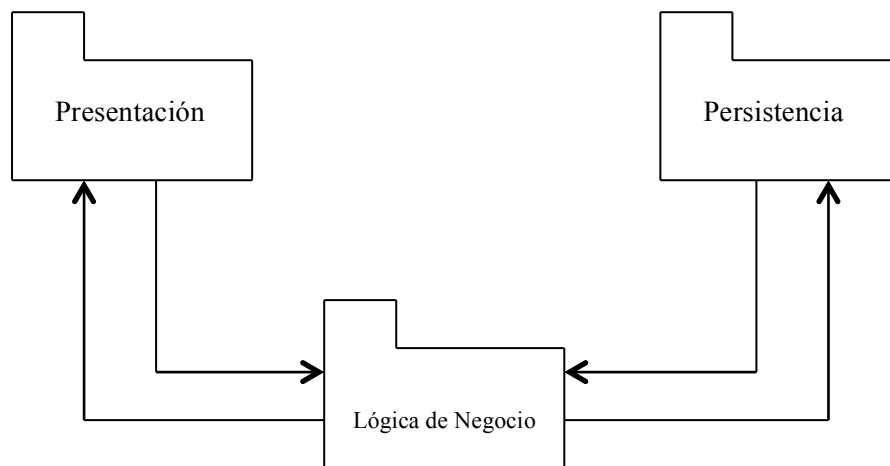
Para el Front-End se utilizó HTML5, y Javascript con el framework ANGULAR JS, que brinda la posibilidad de separar las vistas de la lógica de experiencia de usuario, además este framework es muy poderoso ya que se puede desarrollar un sin número de librerías que aportan una gran funcionalidad al sistema.

Para el diseño del sistema se utilizó BOOTSRAP, un framework CSS proporcionado por Twitter que ayuda a que el sistema se pueda adaptar a todo tipo de pantalla y ahorrar tiempo en el desarrollo.

3.2 Diagramas Finales

3.2.1 Diagrama de paquetes

Figura #13 Diagrama de paquetes
Elaborado por: Bryan Guamba



3.2.2 Diagrama de componentes

Figura #14 Diagrama de paquetes
Elaborado por: Bryan Guamba

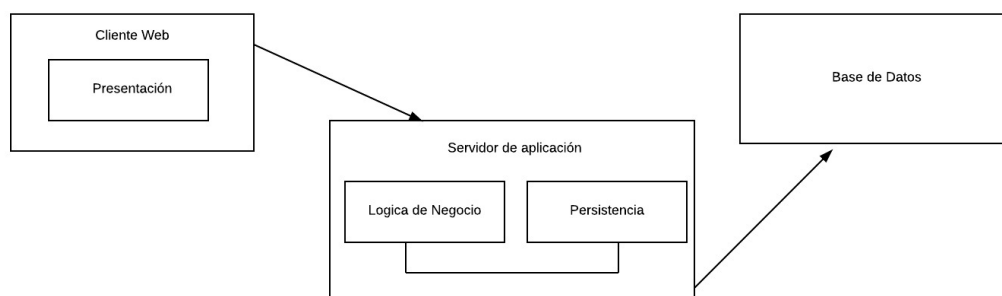


Figura #15 Diagrama de despliegue

Elaborado por: Bryan Guamba



3.2.3 Descripción de componentes

Los componente del sistema son 3 :

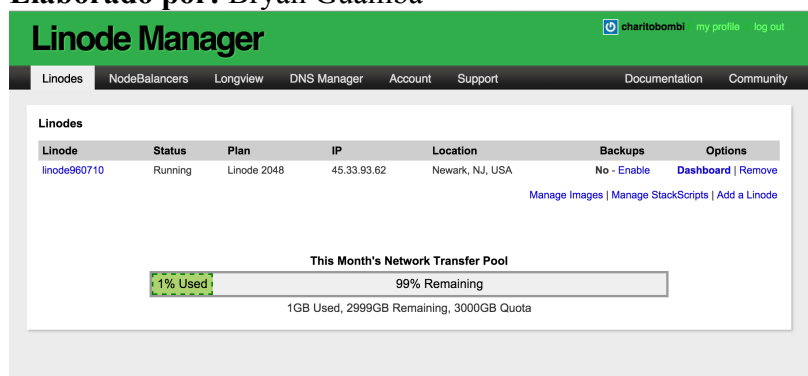
1. Cliente Web: En este componente esta toda la parte visual con la que el usuario interactúa con el sistema.
2. Servidor de Aplicación: En este componente se encuentra toda la lógica de negocio del sistema, donde recibe, procesa y entrega resultados al usuarios.
3. Base de datos: Este componente almacena toda la información que el sistema utiliza para funcionar, entrega resultados en base a las peticiones que realice la aplicación.

3.3 IMPLEMENTACIÓN

Para la implementación del sistema en el servidor en la nube contratado se siguió el siguiente proceso:

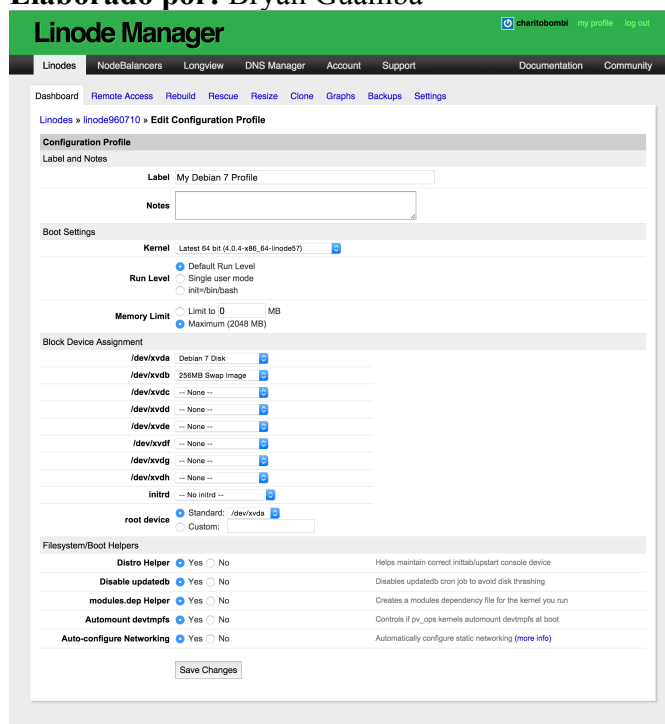
- 1 Ingresar al LINODE Manager mediante la Usuario y contraseña proporcionados por LINODE en el momento de su compra.

Figura #16 LINODE Manager
Elaborado por: Bryan Guamba



- 2 Ingresar en la configuración del servidor y crear un nuevo perfil

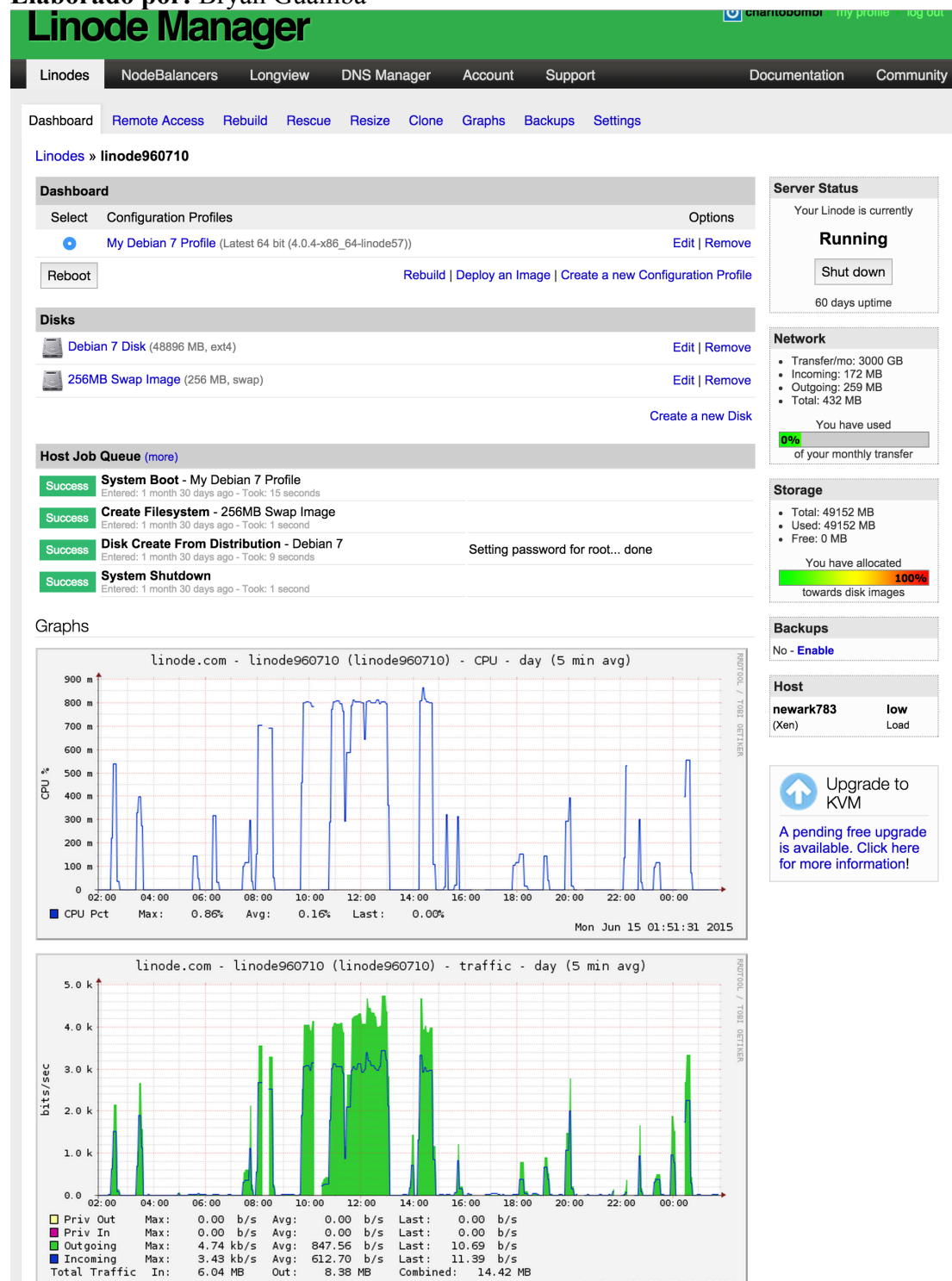
Figura #17 Configurar nuevo perfil LINODE
Elaborado por: Bryan Guamba



- 3 Después de guardar las preferencias por defecto, LINODE creará una máquina con el sistema operativo escogido y presentara en el dashboard un resumen general.

Figura #18 Dashboard LINODE

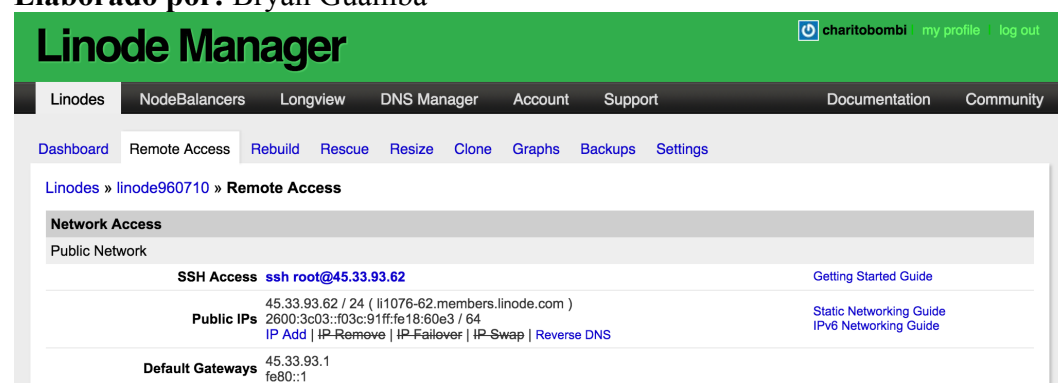
Elaborado por: Bryan Guamba



- Ingresar a la pestaña Remote Access para identificar las formas de acceso remoto al servidor

Figura #19 Acceso Remoto LINODE

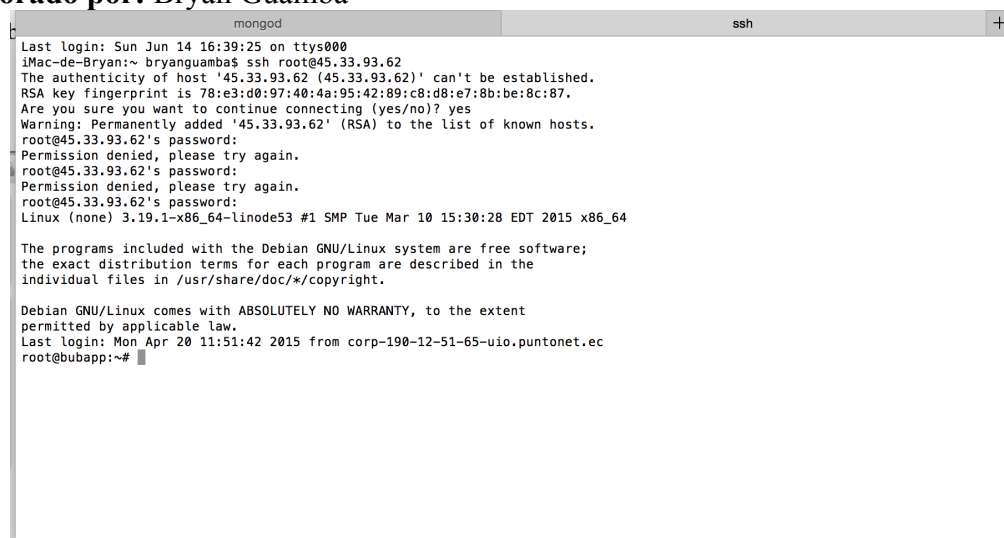
Elaborado por: Bryan Guamba



- Para poder instalar todas las dependencias del sistema se conecta mediante SSH o acceso remoto seguro hacia el servidor en la nube de LINODE con la configuración obtenida.

Figura #20 Consola Acceso Remoto al servidor por SSH

Elaborado por: Bryan Guamba



6 Una vez en el servidor se instala la base de datos con los siguientes comandos:

```
"sudo apt-key adv --keyserver keyserver.ubuntu.com --recv 7F0CEB10"

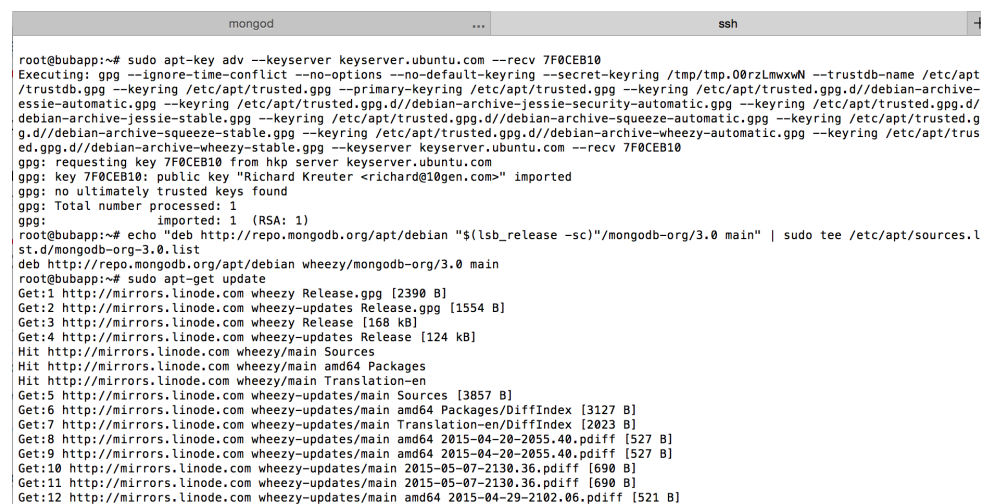
"echo "deb http://repo.mongodb.org/apt/debian "$(lsb_release -sc)"/mongodb-org/3.0
main" | sudo tee /etc/apt/sources.list.d/mongodb-org-3.0.list"

"sudo apt-get update"

"sudo apt-get install -y mongodb-org"
```

Figura #21 Instalación base de datos MongoDB

Elaborado por: Bryan Guamba



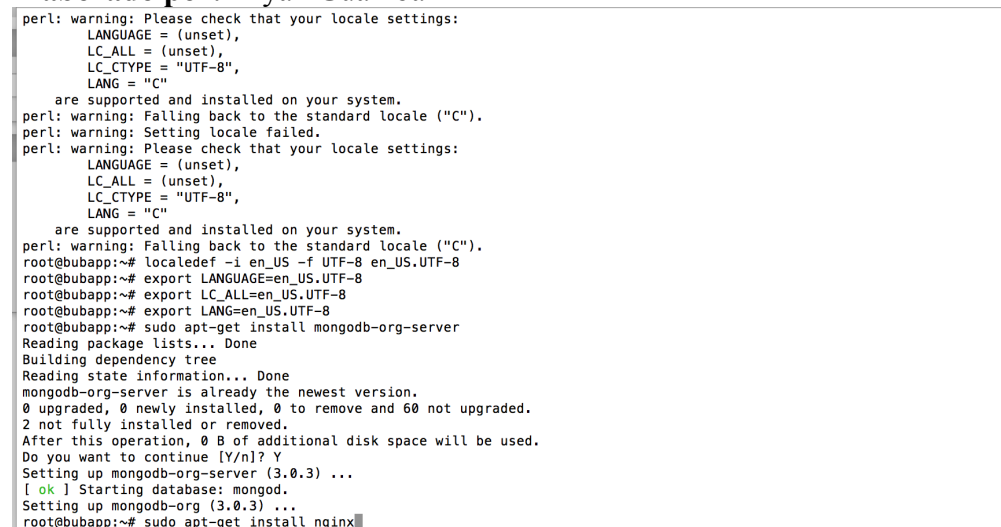
```
root@bubapp:~# sudo apt-key adv --keyserver keyserver.ubuntu.com --recv 7F0CEB10
Executing: gpg --ignore-time-conflict --no-options --no-default-keyring --secret-keyring /tmp/tmp.00rzLmwxwN --trustdb-name /etc/apt/
/trustdb.gpg --keyring /etc/apt/trusted.gpg --primary-keyring /etc/apt/trusted.gpg --keyring /etc/apt/trusted.gpg.d//debian-archive-j
essie-automatic.gpg --keyring /etc/apt/trusted.gpg.d//debian-archive-jessie-security-automatic.gpg --keyring /etc/apt/trusted.gpg.d//
debian-archive-jessie-stable.gpg --keyring /etc/apt/trusted.gpg.d//debian-archive-squeeze-automatic.gpg --keyring /etc/apt/trusted.gp
g.d//debian-archive-squeeze-stable.gpg --keyring /etc/apt/trusted.gpg.d//debian-archive-wheezy-automatic.gpg --keyring /etc/apt/trust
ed.gpg.d//debian-archive-wheezy-stable.gpg --keyserver keyserver.ubuntu.com --recv 7F0CEB10
gpg: requesting key 7F0CEB10 from hkp server keyserver.ubuntu.com
gpg: key 7F0CEB10: public key "Richard Kreuter <richard@10gen.com>" imported
gpg: no ultimately trusted keys found
gpg: Total number processed: 1
gpg:      imported: 1 (RSA: 1)
root@bubapp:~# echo "deb http://repo.mongodb.org/apt/debian "$(lsb_release -sc)"/mongodb-org/3.0 main" | sudo tee /etc/apt/sources.li
st.d/mongodb-org-3.0.list
deb http://repo.mongodb.org/apt/debian wheezy/mongodb-org/3.0 main
root@bubapp:~# sudo apt-get update
Get:1 http://mirrors.linode.com wheezy Release.gpg [2390 B]
Get:2 http://mirrors.linode.com wheezy-updates Release.gpg [1554 B]
Get:3 http://mirrors.linode.com wheezy Release [168 kB]
Get:4 http://mirrors.linode.com wheezy-updates Release [124 kB]
Hit http://mirrors.linode.com wheezy/main Sources
Hit http://mirrors.linode.com wheezy/main amd64 Packages
Hit http://mirrors.linode.com wheezy/main Translation-en
Get:5 http://mirrors.linode.com wheezy-updates/main Sources [3857 B]
Get:6 http://mirrors.linode.com wheezy-updates/main amd64 Packages/DiffIndex [3127 B]
Get:7 http://mirrors.linode.com wheezy-updates/main Translation-en/DiffIndex [2023 B]
Get:8 http://mirrors.linode.com wheezy-updates/main amd64 2015-04-20-2055.40.pdiff [527 B]
Get:9 http://mirrors.linode.com wheezy-updates/main amd64 2015-04-20-2055.40.pdiff [527 B]
Get:10 http://mirrors.linode.com wheezy-updates/main 2015-05-07-2130.36.pdiff [690 B]
Get:11 http://mirrors.linode.com wheezy-updates/main 2015-05-07-2130.36.pdiff [690 B]
Get:12 http://mirrors.linode.com wheezy-updates/main amd64 2015-04-29-2102.06.pdiff [521 B]
```

7 Se instala el servidor web NGINX y las librerías PHP con el siguiente comando:

```
"sudo apt-get install nginx"
```

Figura #22 Instalación del servidor web NGINX

Elaborado por: Bryan Guamba



```
perl: warning: Please check that your locale settings:
LANGUAGE = (unset),
LC_ALL = (unset),
LC_CTYPE = "UTF-8",
LANG = "C"
are supported and installed on your system.
perl: warning: Falling back to the standard locale ("C").
perl: warning: Setting locale failed.
perl: warning: Please check that your locale settings:
LANGUAGE = (unset),
LC_ALL = (unset),
LC_CTYPE = "UTF-8",
LANG = "C"
are supported and installed on your system.
perl: warning: Falling back to the standard locale ("C").
root@bubapp:~# localedef -i en_US -f UTF-8 en_US.UTF-8
root@bubapp:~# export LANGUAGE=en_US.UTF-8
root@bubapp:~# export LC_ALL=en_US.UTF-8
root@bubapp:~# export LANG=en_US.UTF-8
root@bubapp:~# sudo apt-get install mongodb-org-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
mongodb-org-server is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 60 not upgraded.
2 not fully installed or removed.
After this operation, 0 B of additional disk space will be used.
Do you want to continue [Y/n]? Y
Setting up mongodb-org-server (3.0.3) ...
[ ok ] Starting database: mongod.
Setting up mongodb-org (3.0.3) ...
root@bubapp:~# sudo apt-get install nginx
```

- 8 Copiar los archivos del sistema al servidor mediante SSH con el siguiente comando:

“sudo scp especigest root@45.33.93.62:/”

Figura #23 Copiando archivos del sistema web al servidor en la nube

Elaborado por: Bryan Guamba

```
mongod ... ssh bash +
Last login: Sun Jun 14 20:58:25 on ttys001
iMac-de-Bryan:~ bryanguamba$ sudo apt-get install nginx
Password:
sudo: apt-get: command not found
iMac-de-Bryan:~ bryanguamba$ ls -l
total 8
drwxr-xr-x  6 bryanguamba  staff   204 Feb 22 21:29 AndroidStudioProjects
drwx-----  3 bryanguamba  staff   102 May 17 22:46 Applications
drwx-----+ 67 bryanguamba  staff  2278 Jun 14 21:17 Desktop
drwx-----+ 45 bryanguamba  staff  1530 Jun  7 21:20 Documents
drwx-----+ 72 bryanguamba  staff  2448 Jun 14 20:34 Downloads
drwx-----@ 14 bryanguamba  staff   476 Jun 14 21:00 Google Drive
drwx-----@ 55 bryanguamba  staff  1870 Jun  3 14:05 Library
drwx-----+  3 bryanguamba  staff   102 Dec 20 05:13 Movies
drwx-----+  4 bryanguamba  staff   136 Dec 27 23:13 Music
drwxr-xr-x  5 bryanguamba  staff   170 Apr 12 11:21 PhpstormProjects
drwx-----+ 13 bryanguamba  staff   442 Mar  1 11:03 Pictures
drwxr-xr-x  6 bryanguamba  staff   204 May 31 19:58 Projects
drwxr-xr-x+  6 bryanguamba  staff   204 Feb 17 10:42 Public
drwxr-xr-x  3 bryanguamba  staff   102 Feb 26 22:56 PycharmProjects
drwxr-xr-x  3 bryanguamba  staff   102 Feb 26 23:33 RubymineProjects
drwxr-xr-x  7 bryanguamba  staff   238 Jan  9 20:16 VirtualBox VMs
-rw-r--r--  1 bryanguamba  staff  1363 Feb 23 01:00 keyAndroid.jks
drwxr-xr-x  4 bryanguamba  staff   136 Dec 21 22:18 openshift
-rw-r--r--  1 bryanguamba  staff    0 Jun  7 21:57 pslog.txt
iMac-de-Bryan:~ bryanguamba$ sudo scp especigest root@45.33.93.62:/
The authenticity of host '45.33.93.62 (45.33.93.62)' can't be established.
RSA key fingerprint is 78:e3:d0:97:40:4a:95:42:89:c8:d8:e7:8b:be:8c:87.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '45.33.93.62' (RSA) to the list of known hosts.
root@45.33.93.62's password:
```

- 9 Una vez copiado los archivos para generar las estructuras básicas para el funcionamiento del sistema se ejecuta el siguiente comando:

“php artisan migrate:refresh --seed”

Figura #24 Corriendo estructura de base de datos

Elaborado por: Bryan Guamba

```
mongod ... ssh bash bash +
iMac-de-Bryan:sae bryanguamba$ clear

iMac-de-Bryan:sae bryanguamba$ php artisan migrate:refresh --seed
*****
* Application In Production! *
*****
Do you really wish to run this command? [y/N] y
*****
* Application In Production! *
*****
Do you really wish to run this command? [y/N] y
Rolled back: 2015_04_17_191022_create_banks_collection
Rolled back: 2015_04_16_221223_create_users_collection
Rolled back: 2015_04_16_184347_create_roles_collection
Rolled back: 2015_04_16_002631_create_warehouses_collection
Rolled back: 2015_04_13_044248_create_customers_collection
*****
* Application In Production! *
*****
Do you really wish to run this command? [y/N] y
Migrated: 2015_04_13_044248_create_customers_collection
Migrated: 2015_04_16_002631_create_warehouses_collection
Migrated: 2015_04_16_184347_create_roles_collection
Migrated: 2015_04_16_221223_create_users_collection
Migrated: 2015_04_17_191022_create_banks_collection
Migrated: 2015_04_20_165051_create_statementConfiguration_collection
Migrated: 2015_04_20_194121_create_bankAccount_collection
Migrated: 2015_04_21_045441_create_checkbookRegister_collection
Migrated: 2015_04_21_190850_create_offices_collection
Migrated: 2015_04_21_190908_create_bonus_collection
Migrated: 2015_04_21_190915_create_discounts_collection
Migrated: 2015_04_28_035936_create_departments_collection
*****
* Application In Production! *
*****
```

3.3.4 Capacitación

La capacitación se realizó en diferentes sesiones a cada uno de los departamentos, detallando el funcionamiento principal, y la estructura general del sistema.

- Departamento de sistemas: Se explicó el funcionamiento, estructura y tecnología utilizada por el sistema para identificar si es que existe algún problema durante su funcionamiento.
- Gerencia General y Finanzas: Capacitación sobre la configuración principal del sistema, tanto como la parametrización y la generación de los perfiles de usuario; así como también el cambio de contraseñas y la generación de reportes.
- Departamento de Ventas: Se explicó la manera de registrar una venta en el sistema, y el cálculo realizado para el IVA y los descuentos respectivos, así como también la consulta de inventarios.
- Departamento de Compras: Se explicó la manera de registrar una factura de compra en el sistema y la consulta de inventarios.

3.3.5 Explotación y Beneficios

Después de la implementación del sistema, y los primeros días de uso, los departamentos identificaron la eficacia con la que pueden tener reportes de los ingresos y gastos que han obtenido en poco tiempo.

Gracias a la utilidad de subida masiva en los productos, proveedores, y clientes se ha ahorrado muchísimo tiempo en el ingreso de datos al sistema.

La gerencia pudo identificar que con un dispositivo móvil como una laptop, Tablet o Smartphone y una conexión a internet se puede ver los resúmenes que proyectaban las ventas de la semana.

El sistema fue aprobado por la gerencia para su uso diario, y tienen proyecciones de crecimiento para el mismo.

3.3.6 Mantenimiento

3.3.6.1 Preventivo

Durante el desarrollo del sistema se realizaron test de caja blanca y caja negra para garantizar que el sistema funcione correctamente ante cualquier cambio que este pueda tener.

3.3.6.2 Correctivo

En el sistema de logs del servidor web NGINX se encuentra un registro de los errores que está provocando el sistema de forma detallada lo que facilita su identificación y corrección.

CAPÍTULO IV

DISCUSIÓN

4.1 CONCLUSIONES

- Se logró realizar el levantamiento de requerimientos mediante reuniones semanales para identificar los puntos críticos en cada uno de los procesos en los cuales se basa el sistema.
- En las constantes reuniones se pudo identificar otros puntos necesarios dentro del flujo de trabajo como la carga masiva de datos y la construcción jerárquica de categorías para productos.
- Se logró diferenciar las funciones y actividades que realizan los usuarios de la empresa Especigest para establecer los perfiles de acceso y así controlar los permisos dentro del sistema.
- Se estructuró la base de datos que permite un manejo dinámico y de alto crecimiento de la información ya que la tecnología de las bases de datos no relacionales permite obtener un crecimiento horizontal en cualquier momento.
- Se diseño y desarrollo el sistema informático web para la gestión y administración de procesos con las normas y principios de calidad de software que establece el código limpio mediante el uso de los frameworks más robustos y populares de la actualidad.

- Se contrató un servidor en la nube con las características necesarias para que el sistema funcione correctamente con el proveedor de servicios cloud LINODE.
- Se realizaron las pruebas para comprobar el funcionamiento y compatibilidad del sistema en las diferentes estaciones de trabajo con las que cuenta Especigest.
- El ahorro de los recursos económicos se pudo apreciar notablemente con la utilización de software libre.
- La aplicación de nuevas tecnologías en el desarrollo de software como Laravel, MongoDB y AngularJs es importante ya que permite desarrollar grandes sistemas informáticos en menor tiempo, y adaptables en el futuro.
- La tecnología en la nube permite un crecimiento único ya que una empresa mediana o pequeña jamás podría implementar una infraestructura tan grande, segura y robusta para garantizar la integridad de su información.

4.2 RECOMENDACIONES

- Se recomienda hacer un seguimiento con las herramientas proporcionadas por el proveedor en la nube para tener estadísticas de funcionamiento del servidor y si es necesario aumentar la capacidad de los discos o de otros elementos de hardware.
- Se recomienda revisar constantemente la configuración de los respaldos de datos programados en el administrador del servidor LINODE para asegurar que la información esté resguardada.
- Se recomienda guardar los nombres de usuarios y claves de accesos al sistema porque son de vital importancia para el uso del mismo, ya que si son entregados a cualquier persona o puestos a la vista de terceros se podría comprometer la integridad de la información.
- Es importante revisar de forma paulatina los registros de actividades del servidor y de la base de datos para identificar cualquier problema que pueda presentar el sistema y sea corregido inmediatamente.

BIBLIOGRAFÍA

- Abrahamsson P., Salo O., Ronkainen J., & Warsta J. (2002). *Agile Software development methods Review and analysis*. VTT Publications.
- Carlos Paramio. (2011, abril 26). El concepto NoSQL, o cómo almacenar tus datos en una base de datos no relacional. Recuperado 23 de marzo de 2014, a partir de <http://www.genbetadev.com/bases-de-datos/el-concepto-nosql-o-como-almacenar-tus-datos-en-una-base-de-datos-no-relacional>
- Damián Pérez Valdés. (2007, octubre 26). ¿Qué son las bases de datos? -. *Maestros del Web*. Recuperado 23 de marzo de 2014, a partir de <http://www.maestrosdelweb.com/editorial/%c2%bfque-son-las-bases-de-datos/>
- Desarrollo del software - Monografias.com. (s. f.). Recuperado 23 de marzo de 2014, a partir de <http://www.monografias.com/trabajos39/desarrollo-del-software/desarrollo-del-software.shtml>
- Esteban Saavedra. (2012, noviembre 22). *Bases de Datos Documentales*. Recuperado a partir de <http://www.slideshare.net/estebansaavedra/bases-de-datos-documentales-15300505>
- José H. Canós, Patricio Letelier, & Carmen Penadés. (2012). Metodologías Ágiles en el Desarrollo de software. Recuperado a partir de http://noqualityinside.com.ar/nqi/nqifiles/XP_Agil.pdf
- Korstanje, M. (2012). Una introducción al pensamiento de Cass Sunstein. A Contracorriente. Recuperado a partir de <http://es.wikipedia.org/wiki/Riesgo>
- Leydi Lizbeth Ramirez Rivas. (2012, julio 26). Ciclo de Vida del Software. *Ciclo de Vida del Software*. Recuperado a partir de <http://ciclodelsoftware.blogspot.com/>
- Roldán, J. I. H. (2011, enero 17). TDD como metodología de diseño de software. *Paradigma*. Recuperado 23 de marzo de 2014, a partir de <http://www.paradigmatecnologico.com/blog/tdd-como-metodologia-de-diseno-de-software/>
- Roger S. Pressman. (2010). *Ingeniería del Software* (5ta ed.). Mc Graw Hill.
- Canós, J. H., Letelier, P., & Penadés, C. (2012). Letelier, P. (15 de 01 de 2006). Ciencia y Técnica Administrativa. Recuperado el 03 de 06 de 2015, a partir de <http://www.cyta.com.ar/ta0502/v5n2a1.htm>

- Zuboff, S. (2011). Osmosis Latina. Recuperado el 03 de 06 de 2015 a partir de <http://www.osmosislatina.com/administracion/>
- Yanvary, E. (Septiembre de 2006). Monografias. Obtenido de <http://www.monografias.com/trabajos39/desarrollo-del-software/desarrollo-del-software.shtml>
- Beedle, M., & Schwaber, K. (2001). Agile Software Development with SCRUM. Prentice Hall.
- Gómez, R. P. (18 de 07 de 2003). Monografias. Recuperado el 03 de 06 de 2015, de Monografias: <http://www.monografias.com/trabajos14/herramicase/herramicase.shtml>
- Slideshare. (15 de 01 de 2014). Recuperado el 03 de 06 de 2015, de Slideshare: http://es.slideshare.net/lordXDie/diagrama-de-flujo-30071959?next_slideshow=1
- Valdés, D. P. (26 de 10 de 2007). Maestros Del Web. Recuperado el 03 de 06 de 2015, de Maestros Del Web: <http://www.maestrosdelweb.com/que-son-las-bases-de-datos/>
- Kioskea. (Junio de 2015). Kioskea.net. Obtenido de Kioskea.net: <http://es.kioskea.net/contents/66-introduccion-bases-de-datos>
- Alegsa, L. (2009). Alegsa. Recuperado el 03 de 06 de 2015, de Alegsa: <http://www.alegsa.com.ar/Dic/sistema%20informatico.php>
- Álvarez, M. A. (01 de 01 de 2001). Desarrollo Web. Recuperado el 03 de 06 de 2015, de Desarrollo Web: <http://www.desarrolloweb.com/articulos/que-es-html.html>
- Heilmann, C. (26 de 09 de 2008). Mosaic. Recuperado el 03 de 06 de 2015, de Mosaic: <http://mosaic.uoc.edu/ac/le/es/m6/ud1/index.html#whatiscss>
- Solis, J. (26 de 09 de 2014). ARWEB. Recuperado el 03 de 06 de 2015, de ARWEB: <http://www.arweb.com/chucherias/editorial/%C2%BFque-es-bootstrap-y-como-funciona-en-el-diseno-web.htm>
- The PHP Group. (2015). PHP. Recuperado el 03 de 06 de 2015, de PHP: <http://php.net/manual/es/intro-what-is.php>
- Patricio. (21 de 03 de 2013). Desarrollando Webs Dinámicas. Recuperado el 03 de 06 de 2015, de Desarrollando Webs Dinámicas: <http://desarrollandowebsdinamicas.blogspot.com/2013/03/que-es-laravel.html>
- Azaustre, C. (09 de 09 de 2013). Carlosazaustre. Recuperado el 03 de 06 de 2015, de Carlosazaustre: <https://carlosazaustre.es/blog/empezando-con-angular-js/>
- Fernández, R. (03 de 02 de 2014). Genbeta. Recuperado el 03 de 06 de 2015, de Genbeta: <http://www.genbetadev.com/bases-de-datos/mongodb-que-es-como-funciona-y-cuando-podemos-usarlo-o-no>

EcuRed. (24 de 09 de 2012). EcuRed. Obtenido de EcuRed:
http://www.ecured.cu/index.php/IDE_de_Programaci%C3%B3n