UCAB Universidad Católica ANDRÉS BELLO

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA INFORMÁTICA

SISTEMA ADMINISTRATIVO PARA MANEJAR LAS CUENTAS DE LOS CLIENTES Y PROVEEDORES DE ELEMÉTRICA

Este Jurado; una vez realizado el examen del presente trabajo ha evaluado su contenido con el resultado: Veinte (20) puntos.

J U R A D O E X A M I N A D O R

Firma:	Firma:	Firma:
Nombre: Fanny Hernández	Nombre: Henry Martínez	Nombre: Gloria Tarrío V.

REALIZADO POR	**DANIEL LORENTE**
TUTOR EMPRESARIAL	**EYRA RODRÍGUEZ**
TUTOR ACADÉMICO	**GLORIA TARRÍO**
FECHA	**OCTUBRE 2017**

UCAB Universidad Católica ANDRÉS BELLO

FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

SISTEMA ADMINISTRATIVO PARA MANEJAR LAS CUENTAS DE LOS CLIENTES Y PROVEEDORES DE ELEMÉTRICA

TRABAJO INSTRUMENTAL DE GRADO
Presentado ante la
UNIVERSIDAD CATÓLICA ANDRÉS BELLO
Como parte de los requisitos para optar al título de
INGENIERO EN INFORMÁTICA

REALIZADO POR	**DANIEL LORENTE**
TUTOR EMPRESARIAL	**EYRA RODRÍGUEZ**
TUTOR ACADÉMICO	**GLORIA TARRÍO**
FECHA	**OCTUBRE 2017**

Dedicatorias

A mis padres Aurelio y Carmen

Por haberme brindado todo su apoyo durante mi carrera universitaria, por guiarme e inculcarme desde pequeño valores y principios que me han servido para lograr las metas que me he propuesto en la vida.

Gracias a ellos hoy en día tengo las herramientas necesarias para lograr construir un gran futuro.

A mis abuelos Silda y Aurelio

Por haberme recibido estos años de carrera universitaria en su casa, por haberme brindado su apoyo y comprensión y por estar pendientes de que no me falte nada.

Ellos son parte importante de este logro.

A mi abuela Tolosa

Por quererme tanto y siempre darme consejos para crecer como persona y cada día ser un mejor ser humano.

A Dios.

Por haberme dado salud para lograr mis objetivos propuestos.

Contenido

Dedicatorias I
Índice de Tablas VI
Sinopsis VII
Capítulo I – Presentación 1
Necesidades de la Empresa 1
Solución Propuesta 3
Objetivo General 4
Objetivos Específicos 4
Aporte Funcional 5
Aporte Tecnológico 5
Alcance 5
Limitaciones 16
Justificación 17
Capítulo II – Marco Teórico 18
1. Sistema administrativo 18
2. Sistema de Cobranza 18
2.1 Cuentas por Cobrar 19
2.2 Cuentas por Pagar 19
3.1. Retención de Impuesto sobre la Renta 20
3.2. Retención de IVA 20
4. Impuesto Sobre la Renta 20
5. Contribuyente Especial: 21
6. Patrones de Diseño 22
6.1 Patrón de diseño Modelo Vista Controlador (MVC) 23
7. Framework Ruby on Rails 24
8. Manejador de Base de Datos SQLite 24
9. Highcharts 24
3.1 Fase de Análisis. 26
3.2 Fase de diseño: 27
3.3 Fase de codificación: 27

3.4 Fase de pruebas: 28
3.5 Fase de mantenimiento: 28
Justificación de la Metodología 28
Capítulo IV – Desarrollo 30
4.1 Análisis de Requisitos 30
4.1.1 Levantamiento de la Información 30
4.1.2 Análisis 31
4.1.3 Evaluación de lenguajes de Programación 31
4.1.4 Proposición de Soluciones 32
4.2 Diseño 33
4.2.1 Diseños de Interfaz 33
4.2.2 Arquitectura del Sistema 33
4.2.3 Diseño e implementación de la Base de Datos 34
4.3 Implementación 35
4.3.1 Módulo de Registro 37
4.3.2 Módulo de Carga y Visualización de datos provenientes de archivos de movimientos de cuentas 39
4.3.3 Módulo para el manejo de los clientes de la empresa 40
4.3.4 Módulo para el manejo de los proveedores de la empresa 41
4.3.5 Módulo para el manejo de cuentas por cobrar de Elemétrica 42
4.3.6 Módulo para el manejo de cuentas por pagar a Elemétrica 42
4.3.7 Módulo de configuraciones 43
5.3.7.1. Descuentos 43
5.3.7.2. Tarifas 44
5.3.7.3. Gestión de Roles 45
5.3.7.4. Recordatorios 45
4.3.8 Módulo de Correos Electrónicos 46
4.3.9 Módulo de Estadísticas y Gráficos 47
4.3.9.1 Monto facturado vs Monto pagado (Último trimestre). 49
4.3.9.2 Monto facturado vs Monto pagado (Histórico). 49
4.3.9.3 Monto facturado vs Monto pagado (Mensual: correspondiente al último trimestre). 49

4.3.9.4 Top clientes deudores (por monto adeudado). 50
4.3.9.5 Top clientes deudores (por meses adeudados). 50
4.3.9.6 Instalaciones. 50
4.3.10 Módulo de descarga de Reportes 51
4.3.11 Módulo de generación de Informes 51
Aportes Funcionales 52
4.3.12 Rediseñar proceso de cobranza de Elemétrica. 52
4.3.13 Rediseñar el proceso de soporte a los clientes de Elemétrica. 53
Aportes Tecnológicos 55
4.3.14 Evaluación de tres Frameworks de desarrollo de aplicaciones web para el desarrollo del sistema administrativo. 55
4.3.15 Integrar sistema de gestión administrativa con el portal web Admin de la empresa 56
Capítulo V – Resultados 58
Capítulo VI – Conclusiones y Recomendaciones 67
Conclusiones 67
Recomendaciones 68
Referencias Bibliográficas 70
Apéndices 73
Apéndice A – Diagramas del proceso de Cobranza 73
Apéndice B – Diagramas del proceso de Soporte 75
Apéndice C – Evaluación de Framework de desarrollo de aplicaciones web 77
Apéndice D – Historias de usuario 86
Historia #1 – Manejo de clientes de la empresa y sus cuentas por cobrar. 86
Historia #2 – Manejo de proveedores de la empresa y sus cuentas por pagar. 87
Historia #3 – Generación de gráficas. 87
Historia #4 – Envío de correos electrónicos a los clientes de la empresa. 88
Historia #5 – Generación de informes. 88
Historia #6 – Generación de recordatorios de pago. 89
Historia #7 – Importación de lista de data. 89
Apéndice E – Modelo Entidad-Relación 90
Apéndice F– Pantallas del sistema 91

Índice de Figuras

Ilustración 1 Metodología Cascada 25
Ilustración 2 Patrón de diseño Modelo, Vista, Controlador 34
Ilustración 3 Proceso de Autentificación 57
Ilustración 4 Diagrama de Flujo del Antiguo Proceso de Cobranza 73
Ilustración 5 Diagrama de Actividades del nuevo Proceso de Cobranza 74
Ilustración 6 Diagrama de Flujo del Antiguo Proceso de Soporte 75
Ilustración 7 Diagrama de Flujo del Nuevo Proceso de Soporte 76
Ilustración 8 Modelo Entidad Relación 90
Ilustración 9 Pantalla Inicio Sesión 91
Ilustración 10 Pantalla Registro 91
Ilustración 11 Pantalla Index Clientes 92
Ilustración 12 Formulario Creación Cliente 92
Ilustración 13 Pantalla Ver Cliente 93
Ilustración 14 PDF Lista de Clientes 93
Ilustración 15 Pantalla Index Proveedores 94
Ilustración 16 Pantalla Index Cuentas por Cobrar 94
Ilustración 17 Pantalla Index Clientes 95
Ilustración 18 Pantalla Index Correos Electrónicos 95
Ilustración 19 Gráfico Monto Facturado vs Monto Pagado Mes Actual 96
Ilustración 20 Gráfico Estado de Cuentas Mes Actual 96
Ilustración 21 Gráfico Top 10 Clientes Deudores por Meses Adeudados 97
Ilustración 22 Gráfico Instalaciones Mes Actual 97
Ilustración 23 Pantalla Registro de Pago Clientes 98
Ilustración 24 Correo Electrónico a Cliente 98
Ilustración 25 Recordatorio Usuario Administración 99
Ilustración 26 Correo Notificación Registro de Pago 99
Ilustración 27 Informe 100

Índice de Tablas

Tabla 1: Módulos del Sistema 35
Tabla 2: Descuentos Manejados en la Empresa 43
Tabla 3 Recordatorios manejados en el Sistema 46
Tabla 4 Comparación Criterios de evaluación 78
Tabla 5 Escala de calificación para la evaluación del uso de "Frameworks" para la implementación del sistema de cobranza. 80
Tabla 6 Tabla de ponderación para la evaluación del el uso de "frameworks" para la implementación del Sistema de Cobranza 85
Tabla 7 ponderación final de evaluación del uso de "Frameworks" para la implementación del aplicativo web 86

Sinopsis

El siguiente trabajo de Grado titulado **"Sistema Administrativo para manejar las cuentas de los Clientes y Proveedores de Elemétrica"** fue desarrollado con la finalidad de permitirle al personal del departamento administrativo manejar la información de los distintos clientes y proveedores de la empresa. Adicionalmente, se manejan cada una de las cuentas por cobrar y pagar respectivamente, ofreciéndoles a los clientes de Elemétrica la posibilidad de registrar sus pagos, facilitando la tarea de cobranza por parte del departamento administrativo.

Cada uno de los módulos que posee el sistema, fue desarrollado con el fin de permitir al departamento administrativo gestionar la cobranza a los distintos clientes, el pago a los proveedores, la visualización de datos y análisis de la información, para llevar un control detallado de esta tarea.

Se trabajó bajo la metodología Cascada debido a la naturaleza del proyecto, ya que se buscaba ordenar rigurosamente las etapas del proceso de desarrollo de software, de tal forma que el inicio de desarrollo de cada etapa debía esperar a la finalización de la etapa anterior. De esta forma cualquier error detectado en la etapa de prueba conducía al rediseño y nueva programación del módulo afectado. Esta metodología permitió obtener resultados parciales independientes, en donde al finalizar cada uno de los módulos, estos fueron validados por el Departamento de Administración de Elemétrica.

Capítulo I – Presentación

Necesidades de la Empresa

Elemétrica es una herramienta de visualización de ventas, comportamiento de productos y recurrencia de clientes para la potenciación de negocio, permitiendo monitorear el desempeño de éstos en forma remota, además cuenta con diferentes indicadores que invitan al empresario moderno a establecer nuevas estrategias para mantenerse siendo competitivo y exitoso. Ésta cuenta con una plataforma web y una aplicación móvil, otorgando la flexibilidad de conectarse desde cualquier lugar y en cualquier momento.

La empresa cuenta con un departamento administrativo, el cual está encargado de gestionar las cuentas de la empresa, realizar el cobro a los distintos clientes, pagar a los proveedores que le prestan servicio, liquidar impuestos, registrar y controlar los recursos financieros, realizar la solicitud de cotizaciones a proveedores, entre otras actividades pertinentes del área.

El departamento realiza el registro de las cuentas por pagar y las cuentas por cobrar revisando día a día los movimientos bancarios de la empresa y a medida que logran visualizar nuevos pagos, deben identificar a que cliente corresponde para posteriormente registrar el pago del respectivo cliente, llevando así un control de aquellos que se encuentran al día y aquellos que tienen deudas pendientes con la empresa para ese momento.

Muchos de los clientes de la empresa, no mantienen una puntualidad ni un orden en sus pagos, por lo que es común que estos se retrasen o no paguen la

totalidad adeudada, por lo que el departamento administrativo debe revisar y llevar un control, con el fin de notificar al cliente cualquier inconveniente asociado a estos pagos y corregirlo lo antes posible.

Esta es una tarea minuciosa, que obliga al departamento administrativo a estar constantemente revisando los movimientos de cuentas y creando estrategias que ayuden a garantizar que todos sus clientes se encuentren al día con el pago de los servicios prestados por la empresa.

A medida que transcurre el mes, se debe llevar un control de aquellos clientes que no se encuentren solventes, para informarles la deuda que poseen con la empresa.

Si la cantidad de proveedores y clientes aumenta, toda esta tarea se hace cada vez más larga y engorrosa, ya que aumenta la rotación de días para cuentas por pagar y se incrementa el volumen de cobranza.

Solución Propuesta

Se ha planteado realizar un sistema administrativo que permita revisar y verificar la información de los distintos clientes y proveedores de la empresa y cada una de las cuentas por cobrar y pagar respectivamente. Los clientes registrarán los pagos hechos a la empresa junto al comprobante bancario que lo demuestre, de esta forma se involucra al cliente en el proceso de cobranza y se disminuye la necesidad de realizar la tarea de identificación de pagos por parte del departamento administrativo.

Los clientes, proveedores, las distintas cuentas de la empresa y demás información manejada por el sistema administrativo serán almacenados en una base de datos, de modo que esta perdure en el tiempo y pueda ser consultada por el departamento administrativo, además de permitir obtener históricos de las cuentas de la empresa.

Contará con un módulo de envío de correos, en donde el departamento administrativo podrá enviar comunicados e información administrativa a los clientes, para hacerles saber si poseen pagos pendientes con la empresa. También dispondrá de alertas vía correo electrónico configurables, las cuales tendrán como función recordar a los clientes atrasados en sus pagos y cuando estos no realicen sus pagos completos.

Por otra parte, permitirá generar gráficas y estadísticas que le permitan al departamento administrativo visualizar fácilmente las entradas y salidas de la empresa en determinados períodos de tiempo y calcular proyecciones de esta,

ayudando a la toma de decisiones. Además de la posibilidad de exportarlas, podrán generar informes integrando estas gráficas y estadísticas para informar a los inversionistas y directores información relevante.

El sistema administrativo se integrará al aplicativo web Admin de la empresa, en donde, entre otras cosas, se lleva el control del estado de la conexión del servidor de la empresa con cada uno de los locales comerciales.

Esta integración permitirá que el Departamento de Operaciones esté al tanto del estado de solvencia de cada uno de los locales y pueda basarse en esto para priorizar el soporte para aquellos clientes que se encuentren solventes antes de los que no lo estén.

Objetivo General

Diseñar e implementar un sistema administrativo para manejar las cuentas de los clientes y proveedores de Elemétrica.

Objetivos Específicos

1. Diseñar e implementar un modelo de base de datos donde se registre la información pertinente.
2. Desarrollar módulo de registro para el portal web.
3. Desarrollar módulo de carga y visualización de datos provenientes de archivos de movimientos de cuentas.
4. Desarrollar módulo para la gestión de clientes de Elemétrica.
5. Desarrollar módulo para la gestión de proveedores de Elemétrica.
6. Desarrollar módulo para la gestión de cuentas por cobrar de Elemétrica.

7. Desarrollar módulo para la gestión de cuentas por pagar de Elemétrica.
8. Desarrollar módulo de configuraciones del sistema administrativo.
9. Desarrollar módulo de envío de correos a clientes y proveedores de Elemétrica.
10. Desarrollar módulo de estadísticas y gráficas.
11. Desarrollar módulo de descarga de reportes.
12. Desarrollar módulo de generación de informes.

Aporte Funcional

13. Rediseñar proceso de cobranza de Elemétrica.
14. Rediseñar el proceso de soporte a los clientes de Elemétrica.

Aporte Tecnológico

15. Evaluación de tres Framework[1] de desarrollo de aplicaciones web para el desarrollo del sistema administrativo.
16. Integrar sistema de gestión administrativa con el portal web Admin de la empresa.

Alcance

1. **Diseñar e implementar base de datos donde se registre la información pertinente.**

Se realizará un análisis de la información a utilizar por el sistema administrativo con el fin de implementar una base de datos para su almacenamiento.

[1] **Framework**: Un Framework es un marco de trabajo (un esqueleto, un patrón) para el desarrollo y/o implementación de una aplicación. (Sánchez, 2006)

La información almacenada en esta base de datos será la de los clientes y proveedores de la empresa, junto con las cuentas por pagar y las cuentas por cobrar de esta, además de los distintos datos utilizados por el sistema administrativo para manejar dicha información.

2. Desarrollar módulo de registro para el portal web

Se desarrollará un módulo de registro y acceso por parte del personal del departamento administrativo de la empresa, para resguardar la integridad de los datos contenidos. Para este módulo se utilizará el envío de una clave provisional por correo electrónico, que luego podrá ser modificada por el usuario del sistema administrativo si así lo desea.

El registro en el sistema administrativo solo será necesario si el usuario no posee una cuenta ya creada, en caso contrario deberá iniciar sesión en éste. Este módulo contará con una sección de recuperación de contraseña, en donde aquellos usuarios que hayan olvidado su contraseña puedan recuperarla mediante el envío a la cuenta de correo electrónico registrada una clave provisional.

3. Desarrollar módulo de carga y visualización de datos provenientes de archivos de movimientos de cuentas.

Se diseñará e implementará un módulo para la carga de reportes con movimientos de cuentas de Elemétrica, del cual se obtendrá la información necesaria a almacenar en el sistema administrativo.

En este módulo se podrán importar en caso de ser necesario archivos Excel con una lista de cuentas por cobrar o por pagar, indicando respectivamente a que

cliente o proveedor pertenece la cuenta. De este modo se cargarán masivamente una lista de cuentas determinadas y éstas serán registradas.

4. Desarrollar módulo para el manejo de clientes de Elemétrica.

Se desarrollará un módulo donde se podrá agregar, modificar, consultar y eliminar clientes en el sistema administrativo. Para el desarrollo de este módulo se evaluarán las diferentes herramientas de diseño web para lograr un manejo completo de los clientes de la empresa y sus datos.

De cada uno de los clientes de la empresa, se registrará el código interno dentro de la empresa, Nombre Fiscal, RIF[2], Nombre Comercial, Concepto de pago, Monto a pagar sin IVA[3] y el Descuento a aplicar, si es el caso.

5. Desarrollar módulo para el manejo de proveedores de Elemétrica.

Se desarrollará un módulo donde el usuario podrá agregar, modificar, consultar y eliminar proveedores en el sistema administrativo. Para el desarrollo de este módulo se evaluarán las diferentes herramientas de diseño web para lograr un manejo completo de los proveedores de la empresa y sus datos.

De cada uno de los proveedores de la empresa, se registrará su nombre fiscal, RIF, número de contacto, correo electrónico, el nombre del servicio a pagar y la frecuencia de pago.

---

[2] **RIF**: Por sus siglas RIF, El Registro Único de Información Fiscal es un registro destinado al control tributario de los impuestos, tasas y contribuciones administrados por el Servicio Nacional Integrado de Administración Aduanera y Tributaria. Es decir, es el número de identificación ante este servicio. (SENIAT, 2015)

[3] **IVA**: El IVA, o Impuesto sobre el Valor Agregado es una tasa que se calcula sobre el consumo de os productos, servicios o cualquier tipo de transacción comercial. (definicion.de, 2008)

6. Desarrollar módulo para el manejo de cuentas por cobrar de Elemétrica.

Se desarrollará un módulo en donde se manejarán las cuentas por cobrar de Elemétrica, En este módulo se podrá agregar, modificar, consultar y eliminar cuentas por cobrar. Para el desarrollo de este módulo se evaluarán las diferentes herramientas de diseño web para lograr un manejo óptimo y natural de las cuentas por cobrar de la empresa y sus datos.

De cada una de las cuentas, se debe seleccionar el cliente el cual realizó el pago, el tipo de pago (Instalación del servicio, mensualidad u otro concepto) por parte del cliente, la fecha en la que la empresa recibió el pago y el monto pagado.

El sistema administrativo también consultará a nivel de base de datos el estado de las cuentas, cuando un cliente realice un pago incompleto o pague en exceso, se enviará un correo electrónico informando de la situación junto con el monto que resta por pagar o el monto sobrante, dependiendo de la situación.

7. Desarrollar módulo para el manejo de cuentas por pagar de Elemétrica.

Se desarrollará un módulo en donde se manejarán las cuentas por pagar de Elemétrica, En este módulo se podrá realizar distintas tareas relacionadas con las cuentas por pagar; entre ellas están el agregar, modificar, consultar las cuentas y eliminar las cuentas por pagar. Para el desarrollo de este módulo se evaluarán las diferentes herramientas de diseño web para lograr una gestión natural de las cuentas por pagar de la empresa y sus datos.

Para cada una de las cuentas por pagar de la empresa se debe seleccionar el proveedor al cual se le realizó el pago del servicio, tipo de pago, monto a pagar y observaciones del pago.

8. Desarrollar módulo de configuraciones del sistema administrativo.

Se desarrollará un módulo de configuraciones del sistema administrativo, en donde se podrán ajustar el envío de alertas y recordatorios de pagos a los clientes y usuarios del sistema administrativo vía correo electrónico. Las alertas vía correo electrónico serán configurables para poder llevar un mejor control y manejo de las cuentas, así como la administración en general de la empresa, permitiendo establecer que el sistema administrativo envíe estos correos de forma diaria, quincenal o mensualmente, según las preferencias del usuario.

Desde este módulo los usuarios del sistema administrativo podrán configurar en qué momento desean que se envíen las alertas a los clientes recordándoles realizar los pagos que tengan pendientes, si aplica; al igual que las alertas de las cuentas por pagar que deba hacer el departamento administrativo.

9. Desarrollar módulo de envío de correos a clientes y proveedores de Elemétrica.

Se desarrollará un módulo de envío de correos electrónicos, con el fin de permitir al Departamento de Administración de la empresa informar a los clientes y proveedores información pertinente referente a pagos, planes de pago, mensajes que el departamento administrativo crea conveniente.

Estos correos electrónicos enviados desde el sistema administrativo serán almacenados en éste y podrán ser consultados por los usuarios con el fin de poder llevar un seguimiento de estas notificaciones enviadas a los clientes y proveedores.

10. Desarrollar módulo de gráficos y estadísticas de pagos.

Se desarrollará un módulo de gráficos y estadísticas de pagos en el cual los miembros del departamento administrativo de la empresa podrán crear gráficos partiendo de los datos registrados en el sistema administrativo, pudiendo generar diversos tipos de estos, que les permitan llevar un control y posteriormente analizar los datos según sea la naturaleza de la infcrmación. En este módulo también se podrán generar listas de los datos en períodos de tiempo específicos y junto a las gráficas ayuden a interpretar y analizar los datos.

En este módulo se deberá seleccionar un período de tiempo para generar un gráfico en específico. El rango de fechas manejado es mensual, trimestral o anual, ya que estos son los períodos que maneja el departamento administrativo para la generación de esta información.

Las gráficas que serán generadas en este módulo son:

a. Monto facturado vs Monto pagado (Último trimestre).
b. Monto facturado vs Monto pagado (Histórico).
c. Monto facturado vs Monto pagado (Mensual: correspondiente al último trimestre).
d. Top clientes deudores (por monto adeudado).
e. Top clientes deudores (por meses adeudados).

f. Instalaciones.

Para el desarrollo de este módulo se hará uso de una herramienta KDD para el procesamiento de datos, identificación de patrones y manejo de estadísticas que ayuden a la toma de decisiones. Para el desarrollo de este módulo, se tomará en cuenta el Framework seleccionado, para evaluar la herramienta KDD a utilizar.

11. Desarrollar módulo de descarga de reportes.

Se desarrollará un módulo que permita exportar reportes en períodos de tiempo determinado en archivos con extensiones comúnmente usadas como lo son Excel y PDF; con el fin de que el departamento administrativo pueda descargar estos reportes alternos con los datos contenidos en el sistema administrativo de una manera más independiente y sencilla, permitiendo analizar de una forma más detallada la información manejada en el sistema administrativo.

En este módulo se podrán exportar los siguientes datos:

- Información referente a los clientes de la empresa; tanto el detalle de algún cliente en específico como una lista completa de los mismos.
- Información referente a los proveedores de la empresa; tanto el detalle de algún proveedor en específico como una lista completa de los mismos.
- Información referente a las cuentas por cobrar de la empresa; tanto el detalle de una cuenta por cobrar en específico como una lista completa de las mismas

- Información referente a las cuentas por pagar de la empresa; tanto el detalle de una cuenta por pagar en específico como una lista completa de las mismas.
- Gráficas generadas por el sistema administrativo en cualquiera de los formatos comúnmente utilizados tales como PNG y JPEG.

12. Desarrollar módulo de generación de informes.

Se desarrollará un módulo que permita generar informes administrativos de cuentas por cobrar y cuentas por pagar, en donde esté contenida la información mostrada en el sistema administrativo y permitirá integrar las distintas gráficas generadas en el módulo pertinente, facilitando de esta manera la generación de los reportes. En estos informes también se podrán añadir conclusiones de la información generada en el informe, recomendaciones o cualquier otra información que el departamento administrativo desee.

Estos informes podrán ser exportados del sistema administrativo en formato PDF garantizando así, que su contenido no será modificado, para luego ser impresos y utilizados por el departamento administrativo para informar a los directores e inversionistas de la empresa.

Aporte Funcional

13. Rediseñar proceso de cobranza de Elemétrica.

Actualmente el departamento administrativo de la empresa maneja las cuentas por cobrar y las cuentas por pagar de la siguiente manera:

A inicios de mes el departamento administrativo debe generar una lista con los clientes y proveedores junto con sus datos principales (identificación de negocio y los compromisos de pago), y día a día registrar los pagos a medida que los clientes vayan realizándolos, agregando el pago al respectivo cliente en la lista ya generada.

Los administradores invierten mucho tiempo en identificar cuáles son los clientes que mantienen compromisos de pago con la empresa, ya que esto implica revisar los movimientos de cuentas bancarias y seleccionar los clientes que se encuentran al día con sus pagos y aquellos que no.

Es importante mencionar, que la gran mayoría de los pagos se realizan a través de transferencias bancarias, el resto mediante depósitos bancarios.

Se plantea el rediseño del proceso de cobranza mediante el desarrollo de un módulo en el sistema administrativo que permita que los clientes registren sus pagos en el cual el cliente deberá identificar la forma en se realizó el pago (transferencia o depósito), si retuvieron algún tipo de impuesto y adjuntar obligatoriamente el comprobante de pago. Este comprobante tendrá dos formatos de ingreso un PDF para transferencias y una imagen para los depósitos.

En caso de haber pagado mediante depósito bancario deben adjuntar una imagen del recibo de pago generado por el banco y adicionalmente detallar en el formulario todos los datos de dicho pago. En estos casos el departamento administrativo deberá validar que la imagen adjuntada por el cliente concuerde con el monto que este registró como pagado.

Este rediseño del proceso involucra al cliente en la tarea de cobranza y propone registrar los pagos mediante la información registrada por los clientes y no exclusivamente mediante la revisión de los movimientos de las cuentas de la empresa e identificación de clientes por parte del departamento administrativo para registrar la cobranza. (Ver Apéndice B).

Para el desarrollo de este módulo del sistema, se evaluarán y utilizarán librerías para el procesamiento de archivos que permitan obtener los datos de los pagos.

14. Rediseñar el proceso de soporte a los clientes de Elemétrica.

El Departamento de Operaciones de la empresa, quien es el encargado de prestarle soporte técnico a los clientes para resolver aquellos problemas que presenten los locales comerciales, actualmente no está al tanto del estado de solvencia de los clientes, por lo que este factor no se toma en cuenta al momento de prestar soporte.

El orden de atención y soporte que el departamento aplica es el siguiente:

- Locales comerciales internacionales
- Cadenas de locales comerciales
- Resto de locales comerciales que no pertenecen a ninguna de las categorías mencionadas anteriormente.

El sistema administrativo tendrá la capacidad de proporcionarle información al Departamento de Operaciones por medio del aplicativo web Admin el estado de solvencia de cada uno de los clientes.

De esta manera, el Departamento de Operaciones podrá priorizar el soporte de aquellos clientes que estén solventes en sus pagos antes de los que no lo estén.

Aporte tecnológico

15. Evaluación de tres Frameworks de desarrollo de aplicaciones web para el desarrollo del sistema administrativo.

Se evaluarán los siguientes Frameworks de desarrollo de aplicaciones web:

- Framework MVC de JavaScript Angular Universal.
- Framework de desarrollo de Aplicaciones web Ruby on Rails.
- Framework para el desarrollo de aplicaciones web en PHP CodeIgniter.

Se realizará una evaluación tomando en cuenta las ventajas y desventajas de estos Frameworks de programación orientados a la programación de aplicaciones web. Con el fin de que el desarrollo de este sistema sea lo más versátil posible, además de garantizar que el sistema pueda cumplir con todos los requerimientos planteados.

16. Integrar sistema administrativo con la aplicación web Admin de Elemétrica.

La empresa cuenta con una aplicación web Admin, en donde se encuentran los distintos locales comerciales que poseen los clientes, permitiendo entre otras cosas al Departamento de Operaciones supervisar el estatus de conexión de cada uno de los locales comerciales con los servidores de la empresa y la carga en tiempo real de los datos generados por cada uno de estos; para al momento de presentarse

una falla de conexión o de carga de datos prestar soporte a estos, identificando el origen del problema y aplicando distintas acciones dependiendo del tipo de problema para reanudar el correcto funcionamiento. Este aplicativo web también permite al departamento comercial, hacer un seguimiento de los distintos usuarios que ingresan a Elemétrica para monitorear sus negocios, permitiéndoles entre otras cosas asesorarlos sobre la herramienta y detectar las nuevas necesidades de los clientes para planificar la manera en que Elemétrica puede mejorar.

El sistema administrativo será integrado al aplicativo web, ya que si bien, es netamente administrativo, es necesario que el Departamento de Operaciones pueda ver desde el aplicativo aquellos locales comerciales que se encuentren morosos con la empresa, para que este pueda priorizar el soporte a prestar no solo en base al tipo de problema en la comunicación entre el local comercial y los servidores de Elemétrica, sino en la solvencia que estos locales comerciales posean.

Limitaciones

1. Para poder utilizar el sistema administrativo es de vital importancia que el usuario mantenga conexión a internet.
2. El sistema administrativo será desarrollado únicamente para el Departamento de Administración de Elemétrica.
3. Para el desarrollo del sistema administrativo se hará uso de la metodología Cascada.
4. El sistema administrativo será manejado por el departamento administrativo de la empresa, los clientes lo único que podrán hacer en este será registrar sus pagos.

5. El sistema administrativo manejará únicamente clientes con locales ubicados a nivel Nacional.

Justificación

Elemétrica en vista de su crecimiento, se ve en la necesidad de invertir en su proceso de cobranza, de modo de reestructurar este proceso, involucrando más al cliente en esta tarea, facilitándola y disminuyendo en gran parte la necesidad de realizar la tarea de identificación de pagos por parte del departamento administrativo.

Para reestructurar este proceso, se propone desarrollar un proyecto informático que además de permitir manejar las cuentas por cobrar y por pagar de cada uno de los clientes de Elemétrica, que estos puedan registrar sus pagos en una plataforma web facilitando la tarea por parte del departamento administrativo.

También contará con módulos que permitan generar gráficos y tablas con los resultados y análisis de los datos, para que el departamento no se vea en la necesidad de utilizar aplicaciones de terceros para hacer estas operaciones. Sino que puedan manejar todo dentro de un mismo sistema.

Capítulo II – Marco Teórico

Para la realización de este Trabajo Instrumental de Grado, es importante tener conocimiento de los siguientes temas y definiciones.

1. Sistema administrativo

Un sistema administrativo es un conjunto de herramientas que tienen como objetivo llevar a cabo las diferentes actividades y procesos administrativos en las pequeñas y medianas empresas. Garantizando un mejor control de la información, para de esta manera promover a la evolución y desarrollo de la empresa. (Gilli, 1998)

La utilización de los sistemas administrativos es el método más adecuado para el logro de los objetivos centrales de cualquier organización, y así generar la información para el control de los resultados alcanzados.

2. Sistema de Cobranza

La Cobranza es el acto y la consecuencia de cobrar; obtener el dinero correspondiente de una actividad relacionada a la venta de un producto o al pago de deudas. (SENIAT, 2013)

Un sistema de cobranza es un conjunto de herramientas relacionadas entre sí con el objetivo de administrar la tarea de gestión de cobranza de forma ordenada y segura, implementando procesos de cobro de morosidad con mecanismos que promuevan el pago de los servicios o productos ofrecidos por la empresa.

2.1 Cuentas por Cobrar

Las cuentas por cobrar o créditos a cobrar son el conjunto de derechos que posee una empresa sobre terceras personas naturales y/o jurídicas pendientes de cobro. El objetivo de las cuentas por cobrar es proporcionar información cuantificada relacionada al monto total de recuperaciones pendientes de cobro a terceras personas naturales y/o jurídicas por operaciones normalmente del giro específico de una empresa. (solocontabilidad, 2017)

2.2 Cuentas por Pagar

Las cuentas por pagar representan el crédito en cuenta abierta que ofrecen los proveedores de la empresa y que se originan generalmente por la compra de bienes o servicios. Por lo general, los proveedores permiten un tiempo de espera antes del pago. Las cuentas por pagar se muestran como un pasivo corriente debido a que una empresa debe pagar su deuda en un período corto de tiempo. (tributos.net, 2017)

3. Retención tributaria

La retención es la obligación que tiene el comprador de bienes y servicios, de no entregar el valor total de la compra al proveedor, sino de guardar o retener un porcentaje del monto total de la deuda en concepto de impuestos. Este valor debe ser entregado al Estado a nombre del contribuyente, para quien esta retención le significa un prepago o anticipo de impuestos. (kousu, 2014)

3.1. Retención de Impuesto sobre la Renta

Es el pago adelantado que un cliente debe efectuar al fisco nacional luego de la compra de un producto o servicio; este pago podrá ser compensado del cálculo en la autoliquidación definitiva del impuesto sobre la renta del ejercicio fiscal que corresponda. Esta retención debe enterarse posteriormente en una oficina receptora de fondos nacionales dentro de los plazos y formas que establezca la Ley de Impuesto Sobre la Renta. (Microjuris.com, 2011)

3.2. Retención de IVA

Es el deber que tiene un ente al comprar bienes o servicios, de no conceder el valor total de la compra al proveedor, sino realizar una retención en el porcentaje que determine la ley para luego depositar en la oficina receptora de fondos nacionales el valor retenido a nombre del vendedor, para quien este valor constituye un anticipo en el pago de su impuesto. (gerenciaytributos, 2017)

4. Impuesto Sobre la Renta

Es una contribución monetaria de carácter obligatoria para personas naturales y jurídicas, calculada en función de sus ingresos durante el ejercicio fiscal. La entidad encargada de la recolección del ISLR en Venezuela es el SENIAT, tal como expresa Rodríguez (2016).

El Impuesto Sobre la Renta grava la renta, es decir, la ganancia que produce una inversión o la rentabilidad del capital. También puede ser el producto del trabajo bajo relación de dependencia o lo producido por el ejercicio de una profesión liberal. Son cargas obligatorias que tanto las personas naturales como jurídicas deben

cancelar para ayudar a financiar al Estado, pues sin los impuestos el Estado no podría funcionar, por no disponer de fondos para construir Infraestructuras, prestar los servicios públicos de salud, educación, defensa, sistemas de protección social, etc. (Nava, 2008)

5. Contribuyente Especial:

Los contribuyentes especiales son los sujetos que han sido calificados por el SENIAT como tales. Estos entes pueden ser personas naturales o jurídicas y puede que sean o no contribuyentes especiales al valor agregado. No dependiendo de su calificación como contribuyente especial, sino del hecho que estén sujetos o no al impuesto y que realicen o no los hechos imponibles establecidos en la ley. (SENIAT, 2003)

La calificación de contribuyente especial es otorgada a determinados deudores tributarios, de acuerdo a criterios manejados por el SENIAT, tal y como lo expresa Rodríguez (2016).

A continuación, se listan los deberes de los Contribuyentes Especiales:

- Presentar sus Declaraciones al efectuar los pagos por concepto de tributos, multas, intereses y demás accesorios, exclusivamente en el lugar indicado en la respectiva notificación. además, estos deberán presentar sus escritos, recursos, notificaciones y cualquier trámite propio de su condición de Sujeto Pasivo, en la dirección señalada en la respectiva notificación.

- Cumplir con las Fechas de sus Obligaciones Fiscales, según lo establecido en la Providencia Administrativa sobre el Calendario de Sujetos Pasivos Especiales y Agentes de Retención para aquellas obligaciones que deban cumplirse para el año en curso.
- Presentar las Declaraciones Definitivas, Estimadas y/o Sustitutivas relacionadas al Impuesto Sobre La Renta (en lo adelante ISLR) y las Retenciones de ISLR, de conformidad con los plazos establecidos en el Calendario de Sujetos Pasivos Especiales emitido anualmente por la Administración Tributaria. Las mismas deberán ser presentadas por medios electrónicos en la Página Web del Portal de SENIAT.
- Presentar las Declaraciones relacionadas con el Impuesto al Valor Agregado y las Retenciones de IVA, dentro de los plazos establecidos en el Calendario de Sujetos Pasivos Especiales que emite anualmente la Administración Tributaria.

Para el desarrollo de este sistema, se hizo uso de las siguientes herramientas y estándares que permitieron construir la solución:

6. Patrones de Diseño

Los patrones de diseño son un conjunto de buenas prácticas a la hora de desarrollar un software Informático, estos ayudan a prevenir problemas frecuentes en la programación orientada a objetos, donde cada patrón describe un problema que ocurre comúnmente en nuestro entorno, facilitando la creación ordenada de sistemas reutilizables y fáciles de mantener.

En la actualidad existe un gran número de patrones de diseño. A continuación, se detalla con mayor grado el patrón usado en la arquitectura del proyecto:

6.1 Patrón de diseño Modelo Vista Controlador (MVC)

Es un patrón de diseño de software que separa la vista de la lógica de negocio, a través de un módulo de comunicación bidireccional, este patrón posee tres componentes: modelo, vista y controlador. Esto significa que por un lado se define la estructura y manejo de la información y por el otro la interacción con el usuario. (Trujillo, 2014)

A continuación, se describe cada una de las capas o componentes de este patrón:

a. **Modelo:** Es la representación lógica de la información y describe cada una de las funcionalidades del sistema, esta capa se encarga de gestionar los permisos para la integración de dicha información así como la actualización, inserción y eliminación de datos.

b. **Controlador:** Es la capa intermedia entre la vista y el modelo, se encarga de responder los eventos generados por el usuario, los cuales constituyen llamados en el Modelo o a la vista dependiendo del tipo de petición.

c. **Vista:** Es la capa que maneja la interacción con el usuario a través de formularios o listados de información.

7. Framework Ruby on Rails

Según la documentación oficial del Framework:

> *"Rails es un framework de desarrollo de aplicaciones web escrito en el lenguaje de Ruby. Está diseñado para simplificar la programación de aplicaciones web, asumiendo que todos los desarrolladores necesitan empezar. Este diseño hace que se tenga que escribir menos código que en otros frameworks. Experimentados desarrolladores Rails dicen que el desarrollo de la aplicaciones web se vuelve mucho más divertido".* (api.rubyonrails, 2017)

8. Manejador de Base de Datos SQLite

El manejador de Base de datos SQLite es una herramienta de software libre, que admite el almacenamiento de información en dispositivos empotrados de una forma sencilla, eficaz, potente y rápida. Esta base de datos implementa el estándar SQL92[4] y agrega extensiones que facilitan su uso en cualquier ambiente de desarrollo, permitiendo que SQLite soporte desde las consultas más básicas hasta las más complejas del lenguaje SQL. Una de las características más importantes es que se puede emplear tanto en dispositivos móviles como en sistemas de escritorio. (sqlite.org, 2000)

9. Highcharts

Es una librería desarrollada en lenguaje JavaScript puro, basada en una biblioteca de gráficos con el objetivo de mejorar aplicaciones web mediante la creación de gráficos interactivos. Esta librería ofrece una amplia variedad de gráficos, como gráficos de línea, de área, de barras, circulares y otros. (highcharts.com, 2017)

---

[4] **Estándar SQL-92**: surgió en el año 1992 para cubrir todas las necesidades de los desarrolladores. fue diseñado para ser un estándar para sistemas de gestión de bases de datos relacionales. (Camuña, 2015)

Capítulo III – Marco Metodológico

El modelo en Cascada o también llamado modelo de ciclo básico, propone un enfoque sistemático y secuencial para el desarrollo del software; este empieza con la especificación de los requerimientos por parte del cliente y trasciende a través de la planeación, modelado, construcción y despliegue. (Pressman, 2002)

En este proyecto se utilizó la metodología en Cascada, con sus respectivas fases que la componen, las cuales permitieron el desarrollo de manera secuencial del sistema de cobranza, pudiendo así cumplir con los objetivos descritos con anterioridad.

Con esta metodología los errores de diseño detectados en la etapa de pruebas se someten a un nuevo rediseño. En la Ilustración 1 puede verse el esquema de la metodología.

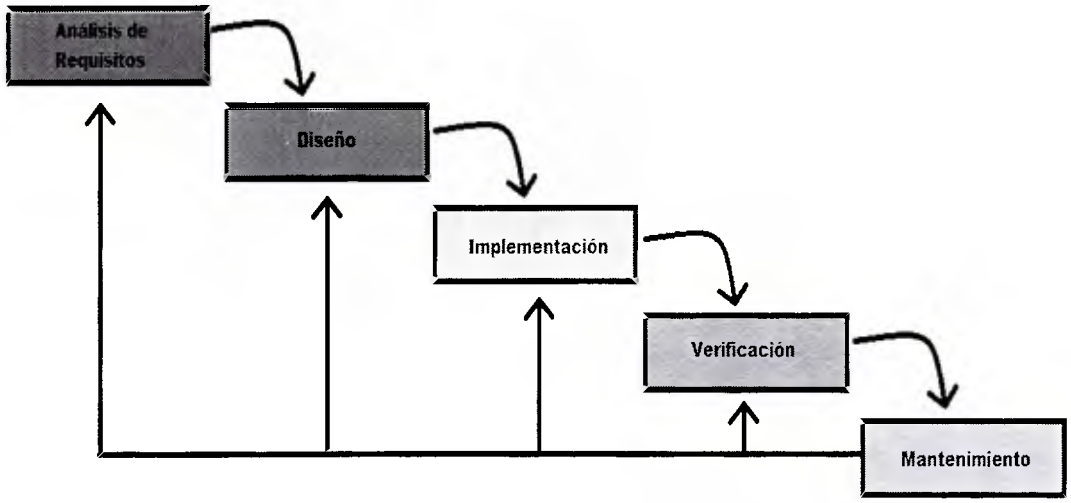

ILUSTRACIÓN 1 METODOLOGÍA CASCADA
Fuente: Elaboración Propia.

3.1 Fase de Análisis.

En esta fase se analizan las necesidades de los usuarios finales del software para determinar qué objetivos debe cubrir. Es importante señalar que en esta etapa se debe consensuar todo lo que se requiere del sistema y será aquello lo que seguirá en las siguientes etapas, no pudiéndose requerir nuevos resultados a mitad del proceso de elaboración del software. (ITSON, 2009)

En esta etapa se realizaron reuniones con el departamento administrativo de la empresa, con el fin de recabar la información necesaria para implementar el sistema.

- Se propuso identificar las necesidades que poseía el proceso de cobranza. Luego se especificaron con mayor profundidad las funcionalidades y características del sistema, pudiendo detectar el alcance de cada una junto a sus limitaciones.
- Se especificó el modelo de datos a utilizar, se realizó la evaluación de las herramientas a manejar y se eligió el marco de trabajo con el cual se desarrolló el proyecto.
- Se plantearon posibles soluciones que cumplían con los objetivos del sistema, con el fin de que el departamento administrativo decidiera cual se ajustaba de una forma más natural y expedita a las necesidades y preferencias.

3.2 Fase de diseño:

Es la fase en donde se realizan los algoritmos necesarios para el cumplimiento de los requerimientos del usuario, así como también los análisis necesarios para saber qué herramientas usar en la etapa de codificación. (ecured, 2017)

Se descompuso y organizó el sistema en elementos que podían elaborarse por separado, posteriormente se realizaron los algoritmos necesarios para el cumplimiento de los requerimientos del usuario, así como también los análisis necesarios para saber que herramientas usar en la etapa de codificación.

3.3 Fase de codificación:

Es la fase de programación o implementación propiamente dicha. Aquí se implementa el código fuente, haciendo uso de prototipos, así como pruebas y ensayos para corregir errores. Dependiendo del lenguaje de programación y su versión se crean las bibliotecas y componentes reutilizables dentro del mismo proyecto para hacer que la programación sea un proceso mucho más rápido. (ITSON, 2009)

En esta etapa se desarrolló el código fuente del sistema administrativo, haciendo uso de prototipos, así como pruebas y ensayos para corregir errores. Se crearon las librerías y componentes reutilizables según las herramientas de desarrollo elegidas.

3.4 Fase de pruebas:

Es la fase en donde los elementos, ya programados, se ensamblan para componer el sistema y se comprueba que funciona correctamente y que cumple con los requisitos, antes de ser puesto. (ITSON, 2009)

En esta etapa, cada uno de los módulos desarrollados se ensambló para componer el sistema en su totalidad, y se comprobó que este funcionara correctamente. Posterior al ensamblado de los módulos se inició el período de pruebas del sistema.

Las pruebas realizadas consistieron en un uso riguroso del sistema, buscando detectar cualquier error generado en la fase de implementación, para su corrección inmediata y así asegurar que se generen los resultados esperados

3.5 Fase de mantenimiento:

En esta etapa del proyecto surgieron algunos cambios, para corregir errores y para introducir mejoras. Se realizó una especificación completa de las características y funcionalidades que debía tener el sistema sin entrar en detalles.

Justificación de la Metodología

Para el desarrollo de este trabajo instrumental de grado, se eligió la metodología en cascada, ya que permite dividir el proyecto en fases o módulos con un alcance definido, que garantizan la culminación exitosa del mismo.

A continuación, se indican las características de la metodología tomadas en cuenta para su elección:

- La cantidad de recursos necesarios para implementar el modelo es mínima. Esto representa un factor importante ya que la empresa tiene restricciones actuales en sus recursos.
- Permite dividir el sistema en distintas funcionalidades construyendo un prototipo funcional tras la realización de cada funcionalidad o módulo del sistema. Este prototipo es requerido por la empresa, ya que esta necesita validar y rectificar cada una de las funcionalidades a implementar lo más pronto posible.
- Cada módulo es desarrollado mediante el ciclo del software permitiendo su análisis, diseño, construcción, pruebas funcionales para ser añadido posteriormente como nueva funcionalidad. Esto cumple con las necesidades de la empresa, ya que permite tener un control de los avances del ciclo del software.
- Cada versión nueva del sistema deja ver el avance del proyecto permitiendo descubrir nuevas funcionalidades, crear nuevos diseños y realizar correcciones.
- La metodología ayuda a mitigar los riesgos del proyecto y dar soluciones a problemas planteados por el departamento administrativo de la empresa.
- Por la estructura de este modelo, facilita el desarrollo del sistema de cobranza.

Capítulo IV – Desarrollo

Para este Trabajo de Grado se realizó un proceso de planificación y desarrollo por fases basadas en la metodología en Cascada, haciendo uso de prototipos, así como pruebas y ensayos para corregir errores. A continuación, se detalla el desarrollo de cada una de estas fases.

4.1 Análisis de Requisitos

En esta fase se procedió a analizar los requerimientos del sistema para conseguir que este cumpliera con las necesidades del departamento administrativo de la empresa (ver Apéndice D), también se establecieron los módulos y funcionalidades que debía tener el producto final.

Se determinó el papel de cada uno de los tipos de usuarios que interactúan con el sistema y cuáles serían las funcionalidades y secciones que brindarían mayor valor al proyecto.

Las actividades en esta fase fueron divididas de la siguiente manera:

4.1.1 Levantamiento de la Información

En las reuniones con el departamento administrativo de la empresa, se indagó sobre la forma en que eran gestionados los clientes y proveedores de la empresa, junto con sus cuentas por cobrar y por pagar respectivamente. También se recolectó la información necesaria para gestionar el resto de características que conforman el proceso de cobranza en su totalidad.

4.1.2 Análisis

En esta etapa se detalló las funcionalidades y características del sistema, especificándose con mayor profundidad, y detectando el alcance de este. Se realizó la evaluación de las herramientas a usar y se eligió el marco de trabajo con el cual se desarrolló el proyecto.

4.1.3 Evaluación de lenguajes de Programación

Al realizar la evaluación de los 3 Frameworks pre seleccionados con anterioridad para el desarrollo del sistema de cobranza se determinó que era conveniente el uso del Framework Ruby on Rails (Ver la sustentabilidad de esta evaluación en el apéndice C), debido a la naturaleza del proyecto y los requerimientos que debía cumplir el sistema, sin embargo, dependiendo de la aplicación a desarrollar, puede ser conveniente utilizar el Framework CodeIgniter o Angular Universal.

Luego de evaluar los Frameworks para el desarrollo de aplicaciones web, se decidió utilizar Ruby on Rails por las siguientes razones (aulaformativa, 2016):

- Permite agilizar el proceso de desarrollo de manera significativa pues da una base para el código y permite reutilizar componentes existentes, de ese modo se ahorra tiempo y costos.
- Está basado en el lenguaje de programación Ruby, lo que presenta grandes ventajas: Permite combinar características de lenguajes dinámicos, incorporando soluciones típicas de lenguajes estáticos.

- El crecimiento de este Framework en los últimos años ha hecho que sea una solución popular al crear aplicaciones web.
- Tanto el Framework como una gran diversidad de librerías son de código abierto. Esto implica que, cualquier proyecto que se realice en Ruby on Rails no tiene ningún coste de licencia.
- Es sencillo de aprender para programadores experimentados
- Una aplicación realizada en Ruby on Rails se puede modificar fácilmente. Añadir nuevas características, aplicar cambios a plantillas y otro tipo de modificaciones pueden realizarse de manera rápida de modo que se ahorra un tiempo valioso.

4.1.4 Proposición de Soluciones

Luego de analizar la información obtenida luego del levantamiento de información inicial, se propuso implementar un sistema de cobranza que permitiese manejar los clientes y proveedores de la empresa, junto con sus cuentas por cobrar y pagar respectivamente, permitiendo el envío de correos electrónicos, con el fin de crear una vía de comunicación en donde el personal del departamento administrativo pudiera trasmitir información que requiera a los clientes de la empresa.

También se propuso desarrollar un módulo de gráficas que permitiese el análisis y evaluación de los datos manejados en el sistema, un módulo de recordatorios para recordar a los clientes los pagos pendientes con la empresa y al personal del departamento administrativo para así, éstos pudieran tener un mejor control del proceso de cobranza, entre otras características descritas más adelante.

4.2 Diseño

En esta fase se tomó como base la información obtenida en la fase de análisis de requerimientos, específicamente en la actividad de levantamiento de información.

Las actividades en esta fase fueron divididas de la siguiente manera:

4.2.1 Diseños de Interfaz

Se realizaron diseños de la interfaz buscando que esta fuera flexible, sencilla y amigable de modo que permitiera un fácil e intuitivo uso para el departamento administrativo y los clientes de la empresa al momento de realizar sus pagos. Se utilizó como base el diseño del sistema Admin de la empresa, se aplicó una gama de colores acordes con los de la empresa.

4.2.2 Arquitectura del Sistema

Se seleccionó para el desarrollo el Framework Ruby on Rails, por lo que se trabajó bajo el patrón de diseño MVC (Modelo, vista, controlador). En la Ilustración 2 se visualiza la arquitectura del patrón de Diseño MVC.

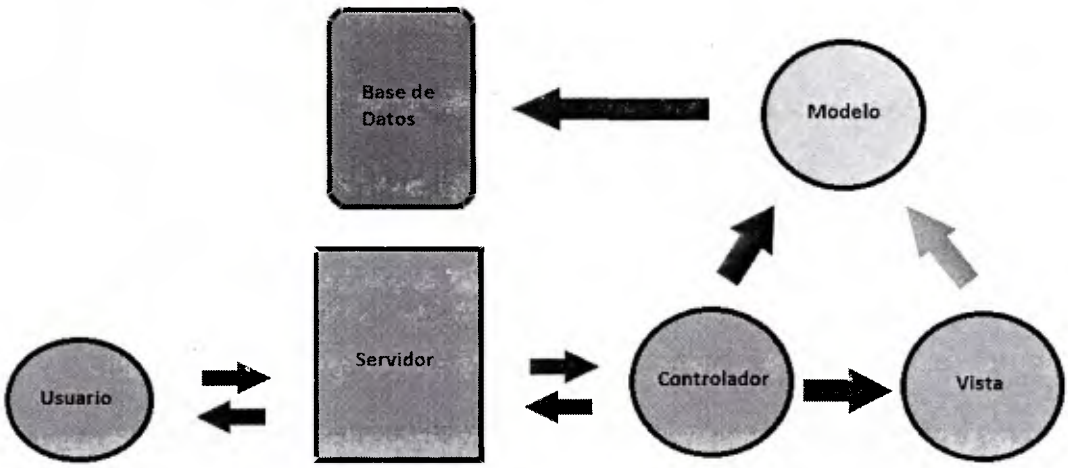

Ilustración 2 Patrón de diseño Modelo, Vista, Controlador
Fuente: Elaboración propia.

Este Framework trabaja con una arquitectura Modelo, Vista, Controlador. Su fundamento es la separación del código en estas tres capas diferentes dependiendo de su responsabilidad.

Para el desarrollo del sistema se utilizó la versión de Ruby 2.3.3p222 y rails 4.2.4, ya que son las versiones estables más recientes de dicho lenguaje de programación y Framework respectivamente.

4.2.3 Diseño e implementación de la Base de Datos

Se diseñó la estructura de la base de datos a través de un modelo Entidad-Relación (Ver Apéndice E). Se decidió usar SQLite3 ya que tiene por características ser ligera, autónoma y al ser una Base de Datos reduce en alto grado la latencia para el acceso de los datos. Además, esta Base de Datos utiliza únicamente la

cantidad de espacio de disco necesario para almacenar realmente la información en ese campo, es decir, no asigna espacios en disco fijos independientemente de la cantidad de información que se almacene. De esta manera, se ahorra el espacio en memoria aumentando la velocidad de respuesta.

4.3 Implementación

El sistema se desarrolló por secciones o módulos, en donde cada uno de estos fue manejado como prototipos, para la ejecución se requirió un proceso de análisis previo para la elaboración de la mejor solución, posteriormente se diseñó la interfaz de dicha sección, para luego implementar las funcionalidades propuestas y al final realizar una tarea de verificación.

Cada uno de estos módulos fue desarrollado según un orden establecido, determinado por la funcionalidad y el valor que cada uno de estos aporta al proyecto. Este orden fue creado a partir de la funcionalidad, el valor agregado a la aplicación final (1 prioridad a 10 menos prioritario) y dependencia de cada módulo. Como se muestra en la Tabla 1.

TABLA 1: MÓDULOS DEL SISTEMA

Módulo	Funcionalidad	Valor Agregado	Dependencia
Modelo de Base de Datos	Gestión de Datos de la Aplicación Web.	**1**	Ninguna
Módulo de registro en el sistema	- Registro de usuarios. - Inicio de sesión. - Bloqueo y Desbloqueo de Cuenta. - Recuperación de Contraseña.	1	Modelo de Base de Datos

Módulo de carga Masiva de Datos	- Carga masiva de Clientes a través de un archivo Excel. - Carga masiva de Proveedores a través de un archivo Excel. - Carga masiva de Cuentas por Cobrar a través de un archivo Excel. - Carga masiva de Cuentas por Pagar a través de un archivo Excel.	6	Modelo de Base de Datos Módulo de manejo de Clientes Módulo de manejo de Proveedores Módulo de manejo de Cuentas por Cobrar Módulo de manejo de Cuentas por Pagar
Módulo de manejo de Clientes	- Añadir clientes. - Modificar Clientes. - Ver Clientes. - Eliminar Clientes.	1	Modelo de Base de Datos
Módulo de manejo de Proveedores	- Añadir Proveedores. - Modificar Proveedores. - Ver Proveedores. - Eliminar Proveedores.	3	Modelo de Base de Datos
Módulo de manejo de Cuentas por Cobrar	- Añadir Cuentas por Cobrar. - Modificar Cuentas por Cobrar. - Ver Cuentas por Cobrar. - Eliminar Cuentas por Cobrar.	2	Modelo de Base de Datos Módulo de manejo de Clientes
Módulo de manejo de Cuentas por Pagar	- Añadir Cuentas por Pagar. - Modificar Cuentas por Pagar. - Ver Cuentas por Pagar. - Eliminar Cuentas por Pagar.	4	Modelo de Base de Datos Módulo de manejo de Proveedores
Módulo de Configuraciones	- Añadir Descuentos. - Editar Descuentos. - Eliminar Descuentos. - Ver descuentos. - Añadir Tarifas. - Editar Tarifas. - Eliminar Tarifas. - Ver Tarifas. - Editar Rol Usuarios. - Eliminar usuarios. - Ver Usuarios. - Ver Recordatorios - Editar Recordatorios	5	Modelo de Base de Datos Módulo de registro en el sistema Módulo de registro en el sistema Módulo de manejo de Clientes Módulo de manejo de Cuentas por Cobrar
Módulo de Correos	- Enviar Correos a Clientes. - Ver Correos. - Modificar Correos. - Eliminar Correos.	6	Modelo de Base de Datos Módulo de manejo de Clientes
Módulo de Estadísticas y Gráficas	- Gráfica Monto facturado vs Monto pagado (Último trimestre, Mes Actual e Histórico).	7	Modelo de Base de Datos Módulo de registro en el sistema

	- Gráfica Top clientes deudores (por monto adeudado). - Gráfica Top clientes deudores (por meses adeudados). - Gráfica Instalaciones. -Grafica por estados de cuenta (Mes actual)		Módulo de manejo de Clientes Módulo de manejo de Cuentas por Cobrar
Módulo de Generación de Reportes	- Generar Reporte PDF y Excel con los Clientes del sistema. - Generar Reporte PDF y Excel con los Proveedores del sistema. - Generar Reporte PDF y Excel con las Cuentas por Cobrar del sistema. - Generar Reporte Excel y PDF con las Cuentas por Pagar del sistema.	9	Modelo de Base de Datos Módulo de manejo de Clientes Módulo de manejo de Proveedores Módulo de manejo de Cuentas por Cobrar Módulo de manejo de Cuentas por Pagar
Módulo de generación de Informes	- Generar Informes. - Ver Informes. - Modificar Informes. - Eliminar Informes.	8	Modelo de Base de Datos Módulo de manejo de Módulo de Estadísticas y Gráficas

Fuente: Elaboración propia.

Las actividades en esta fase fueron divididas de la siguiente manera:

4.3.1 Módulo de Registro

Se realizó un módulo que permite el registro e inicio de sesión por parte del personal del departamento administrativo de la empresa y los clientes de esta.

Este módulo cuenta con una interfaz simple que permite al usuario registrarse en el sistema por medio de un correo electrónico y una contraseña. Para realizar esto, el usuario debe ingresar su correo electrónico junto a una contraseña y su confirmación, al mandar la petición de creación de cuenta, es enviado un correo con un enlace para la creación de dicha cuenta. Para culminar la creación se debe

acceder a este enlace, el cual redirige al sistema e ingresar las credenciales creadas en el paso anterior.

Además, se implementó el bloqueo de cuenta, que sucede de forma automática si se intenta ingresar a la aplicación web con una contraseña errónea más de tres veces. Automáticamente la cuenta es bloqueada y el usuario no podrá ingresar de la manera usual. Para desbloquear la cuenta el usuario debe seguir las instrucciones que consisten en seleccionar la opción de desbloqueo de cuentas y acceder al link que el sistema envía vía correo electrónico. Finalmente, el usuario debe establecer una nueva contraseña de acceso al sistema.

A parte de la recuperación de cuenta por bloqueo, el módulo de autentificación de usuarios, posee una opción para cambiar la contraseña cuando el usuario lo desee.

El sistema cuenta con cuatro roles diferentes para controlar el alcance que tienen los usuarios en el sistema.

- Rol administrador: El cual posee todos los privilegios para ver, editar y eliminar la información contenida en el sistema.
- Rol limitado: El cual permite al usuario solo ver y editar la información.
- Rol restringido: El cual permite que el usuario solo pueda visualizar la información contenida en el sistema.

Estos primeros tres roles están hechos para el personal administrativo de la empresa y para terceros que el Departamento de Administración crea pertinente pueda ingresar a este sistema, Aquellos usuarios con rol administrador, son los que

tienen la posibilidad de manejar los roles de cada uno de los usuarios registrados en el sistema.

Desde el módulo de configuraciones pueden modificar según su preferencia el tipo de rol al usuario que deseen.

Por último, el cuarto rol es para los clientes de la empresa, este rol es totalmente diferente a los tres anteriores, ya que al ingresar a la aplicación web, sólo podrán visualizar únicamente el módulo de registro de pagos.

Inicialmente cuando un usuario se registra en el sistema, este le asigna automáticamente el rol de clientes, y posteriormente un usuario administrador tendrá la posibilidad de cambiar esto si así lo desea. Esto con el fin de garantizar la integridad de los datos en el sistema.

Para la realización de este módulo se trabajó con la gema devise, la cual aporta las herramientas necesarias para el manejo de autenticación de usuarios.

4.3.2 Módulo de Carga y Visualización de datos provenientes de archivos de movimientos de cuentas

En cada uno de los módulos para el manejo de Clientes, Proveedores, Cuentas por Pagar y Cuentas por Pagar, se desarrolló una sección para la carga masiva de datos.

En cada una de estas secciones, los usuarios podrán adjuntar un archivo de Excel con una lista de clientes, proveedores, cuentas por pagar o cuentas por pagar

respectivamente, para agregar esta información al sistema de forma masiva, sin tener que introducirla manualmente.

Estas secciones contienen unas breves instrucciones que le indican al usuario la forma que debe tener la información contenida en los archivos de Excel, ya que el sistema debe identificar cada una de las columnas de estos registros y almacenar la información en el campo correspondiente. Por lo cual cada una de estas columnas debe tener un nombre y un formato específico indicado en las instrucciones del módulo pertinente. De no hacerlo así, el sistema mostrará por pantalla luego de la importación las líneas en que se encontraban los registros que no logró registrar debido a algún error en el formato de la información.

4.3.3 Módulo para el manejo de los clientes de la empresa

Se realizó un módulo que permite manejar los clientes de Elemétrica, con capacidad de registrar, modificar, consultar y eliminar cada uno de estos.

Este módulo cuenta con una interfaz sencilla, en donde se pueden registrar los clientes de la empresa. En la página principal de este módulo se muestra una lista con todos los clientes registrados en el sistema.

Además, el módulo posee una casilla de búsqueda por RIF, para que el departamento administrativo pueda encontrar fácilmente el cliente que desee, y una opción de búsqueda avanzada, en donde se puede realizar búsquedas por varios campos a aquellos clientes que cumplan con las características de la búsqueda realizada.

En la pantalla principal de este módulo, donde se encuentra la lista de todos los clientes registrados en el sistema, se realizó una función de ordenamiento alfabéticamente por cada una de las columnas, de modo que se pueda ordenar esta lista alfabéticamente de forma ascendente y descendente por la primera letra de cada campo según se desee, para su fácil manejo.

4.3.4 Módulo para el manejo de los proveedores de la empresa

Se realizó un módulo que permite manejar los proveedores de Elemétrica, con capacidad de registrar, modificar, consultar y eliminar cada uno de éstos.

Este módulo cuenta con una interfaz sencilla, en donde se pueden registrar los clientes de la empresa. En la página principal de este módulo se muestra una lista con todos los proveedores registrados en el sistema.

Además, el módulo posee una casilla de búsqueda por RIF, para que el departamento administrativo pueda encontrar fácilmente el proveedor que desee, y una opción de búsqueda avanzada, en donde se puede realizar búsquedas por varios campos a aquellos proveedores que cumplan con las características de la búsqueda realizada.

En la pantalla principal de este módulo, donde se encuentra la lista de todos los proveedores registrados en el sistema, se realizó una función de ordenamiento alfabéticamente por cada una de las columnas, de modo que se pueda ordenar esta lista alfabéticamente de forma ascendente y descendente por la primera letra de cada campo según se desee, para su fácil manejo.

4.3.5 Módulo para el manejo de cuentas por cobrar de Elemétrica

Se realizó un módulo que permite manejar las cuentas por cobrar de cada uno de los clientes registrados en el sistema con capacidad de agregar, editar, y eliminar cuentas a cada uno de estos.

Este módulo cuenta con una interfaz sencilla, donde se pueden registrar las cuentas por cobrar a los clientes registrados en el sistema.

En el registro de una cuenta por cobrar, también hay una opción para adjuntar el comprobante bancario, en esta sección se pueden adjuntar imágenes o archivos PDF con la información del pago respectivo.

Al igual que el módulo de clientes y proveedores, cuanta con una búsqueda por el número de factura para que el departamento administrativo pueda encontrar fácilmente una cuenta en particular, y una opción de búsqueda avanzada, en donde se puede realizar búsquedas por varios campos a aquellas cuentas por cobrar que cumplan con las características de la búsqueda realizada.

4.3.6 Módulo para el manejo de cuentas por pagar a Elemétrica

Se realizó un módulo que permite manejar las cuentas por pagar a cada uno de los proveedores registrados en el sistema con capacidad de agregar, editar, y eliminar cuentas a cada uno de estos.

Este módulo cuenta con una interfaz sencilla, donde se pueden registrar las cuentas por pagar a los proveedores registrados en el sistema.

Al igual que el módulo de clientes y proveedores y cuentas por cobrar, cuenta con una búsqueda por el número de factura para que el departamento administrativo pueda encontrar fácilmente una cuenta en particular, y una opción de búsqueda avanzada, en donde se puede realizar búsquedas por varios campos a aquellas cuentas por pagar que cumplan con las características de la búsqueda realizada.

4.3.7 Módulo de configuraciones

Se realizó un módulo capaz de manejar las configuraciones y ciertos aspectos en el sistema de cobranza. Este módulo está compuesto por cuatro secciones y fue realizado con una interfaz sencilla, en donde se puede ajustar cada una de las configuraciones adicionales del sistema.

5.3.7.1. Descuentos

Dentro de este módulo se desarrolló una sección para el manejo de los descuentos que tiene la empresa. Estos descuentos son aplicados a los clientes en sus cuentas por cobrar. En la Tabla 2 se detalla los descuentos manejados actualmente en la empresa:

TABLA 2: DESCUENTOS MANEJADOS EN LA EMPRESA

Nombre del Descuento	Descripción
Descuento por Volumen	Este descuento se les aplica a los clientes, los cuales la empresa le presta el servicio a dos o más locales de su pertenencia. El valor de este descuento es un 15% sobre el monto total a pagar según el concepto de pago correspondiente.
	Este descuento se le aplica a los clientes que tienen un determinado tiempo disfrutando del servicio que presta la

Descuento por Fidelidad	empresa, el departamento comercial de esta es el que designa cuales locales se les aplicará este descuento. El valor de este descuento es un 10% sobre el monto total a pagar el concepto de pago correspondiente.
Descuento Especial	Este descuento es aplicado a los clientes por diversas razones, dependiendo del acuerdo que llegue el Departamento de Ventas con el cliente. Este descuento no tiene un valor determinado, el departamento comercial decide su valor.

Fuente: Elaboración propia.

Esta sección fue realizada con la capacidad de poder agregar la cantidad de descuentos que el departamento administrativo junto al departamento comercial desee, esto para que el sistema sea más versátil y no restringir únicamente a los descuentos existentes en la actualidad en la empresa

5.3.7.2. Tarifas

Dentro de este módulo también se desarrolló una sección para el manejo de las tarifas o conceptos de pago manejados por la empresa. En esta sección se pueden ajustar los montos de cada concepto de pago.

Se creó esta sección, ya que cada cierto tiempo la empresa realiza un ajuste de precios a cada uno de las tarifas de pago y las cuentas por cobrar generadas deben ir de acuerdo con los montos por concepto de actualizados.

Actualmente las tarifas existentes en la empresa son:

- Instalación
- Mensualidad

- Mensualidad para Clientes antiguos
- Instalación y Mensualidad
- Reactivación

La instalación para clientes antiguos es un monto preferencial que se les aplica a los clientes que forman parte de la empresa desde sus inicios.

5.3.7.3. Gestión de Roles

El módulo de configuraciones tiene una sección de gestión de roles de usuario disponible únicamente para aquellos usuarios que posean rol administrador. En esta sección los usuarios administradores pueden ver y editar el rol de cada uno de los usuarios registrados en el sistema. También pueden eliminar usuarios registrados si así lo desean.

5.3.7.4. Recordatorios

Por último, este módulo posee una sección para controlar el envío de recordatorios vía correo electrónico a los clientes y el personal administrativo de la empresa. En esta sección se podrá configurar los tiempos en que se envían los recordatorios de pago. En la Tabla 3 se describen los recordatorios manejados por el sistema:

TABLA 3 RECORDATORIOS MANEJADOS EN EL SISTEMA

Nombre Recordatorio	Descripción
Recordatorio de pago	Este recordatorio es enviado diaria, quincenal o mensualmente a los clientes que no han realizado el pago correspondiente al mes actual. Es enviado un correo electrónico a todas las direcciones registradas por cada cliente con la información del pago que deben realizar para mantenerse solventes con la empresa
Recordatorios a Usuarios con rol Administrador	Este recordatorio es enviado diaria, quincenal o mensualmente a los usuarios con rol administrador en el sistema para recordarle información en particular referente a algún cliente o proveedor registrado en el sistema. Esta información es almacenada como comentario en el respectivo cliente o proveedor

Fuente: Elaboración propia.

Los tiempos manejados por el sistema son:

- Diario: Los correos recordatorios son enviados diariamente
- Quincenal: Los correos recordatorios son enviados quincenalmente.
- Mensual: Los correos recordatorios son enviados mensualmente.

4.3.8 Módulo de Correos Electrónicos

Se realizó un módulo capaz de enviar correos electrónicos a los clientes del sistema; Este módulo posee una interfaz sencilla y permite al personal del departamento administrativo enviar correos a los clientes con información relevante relacionada al cobro de los servicios de la empresa. Los correos electrónicos son enviados a todas las direcciones de correo electrónico registradas del cliente seleccionado.

En este módulo también se pueden enviar correos masivamente, permitiendo enviar comunicaciones generales a todos los clientes que se encuentren registrados en el sistema. Las peticiones al servidor son realizadas en segundo plano, garantizando de igual manera que esta tarea no comprometa el tiempo de respuesta del sistema.

Para la realización de este módulo se hizo uso de la gema Delayed Job y Action Mailer. Delayed Job es una librería que proporciona clases y métodos para manejar el encapsulamiento de actividades de forma asincrónica. Esta gema permite almacenar en la base de datos la cola de peticiones realizadas por el cliente para que luego sean ejecutadas en segundo plano.

Action Mailer es una librería que proporciona clases y vistas para permitir el envío de correos electrónicos haciendo uso del protocolo SMTP[5] para enviar estos correos, esta librería permite configurar un Layout[6] que contendrá cada correo electrónico.

4.3.9 Módulo de Estadísticas y Gráficos

Se realizó un módulo capaz de generar gráficos relacionados a métricas para el análisis de la información contenida en el sistema, estas gráficas permiten que el departamento administrativo pueda visualizar la data, identificar patrones de comportamiento de los clientes en relación al pago de sus facturas, hacer

---

5 **Protocolo SMTP**: "(Protocolo simple de transferencia de correo) es un protocolo TCP/IP que se utiliza para enviar y recibir correo electrónico." (IBM, 2010)

6 **Layout**: Término inglés con el que se conocen en diseño los bocetos o maquetas bien acabados. Es la ordenación y colocación de todos los elementos que componen una página web. (Natali, 2009)

estimaciones según los resultados observados y generar mecanismos y acciones para la mejora constante del cobro de las cuentas a los clientes.

Para el desarrollo de este módulo se utilizó la librería highcharts, la cual es una librería escrita en javascript para la creación de gráficas.

A continuación, se describen sus características:

- La librería es compatible con todos los navegadores modernos incluyendo algunos menos modernos como Internet Explorer.
- Todas las características pueden ser personalizadas permitiendo una gran flexibilidad además, está escrito solamente con código Javascript.
- Posee compatibilidad con SQLite3.
- Se basa únicamente en tecnologías de navegador nativo y no requiere plugins de cliente como Flash o Java.
- No es necesario instalar nada en el servidor. Highcharts sólo necesita un archivo JavaScript para ejecutarse.
- Proporciona gran variedad de gráficos, entre ellos (Gráfico de línea, de área, de columnas, de barras, de tortas, entre otros).
- Posee un módulo para exportar las gráficas en formatos PNG, JPG, PDF o SVG o imprimir el gráfico directamente desde la página web.

Esta librería fue implementada a través de la gema llamada chartckick, que se encarga de proveer clases y mecanismos para la integración de highchart a la aplicación.

Para generar cada uno de los gráficos se hizo uso de la herramienta ORM Active Record que proporciona el Framework Ruby on Rails, la cual facilita la creación y manipulación de objetos de negocio quienes requieren ser almacenados persistentemente en la base de datos.

Los gráficos solicitados desde un principio por el departamento administrativo fueron:

4.3.9.1 Monto facturado vs Monto pagado (Último trimestre).

Esta es la combinación de 3 gráficos de barras, en donde se indica en el último trimestre, excluyendo el mes en curso, el monto total facturado y el monto total pagado por los clientes en cada uno de los meses.

Este total facturado es la suma de todos los conceptos de pago de cada una de las cuentas por cobrar con estado facturada generadas en el período, el total pagado es la suma de todos los montos pagados por los clientes en sus cuentas en el periodo dicho período.

4.3.9.2 Monto facturado vs Monto pagado (Histórico).

Este gráfico consta de dos barras, una que indica el monto total facturado por la empresa desde sus inicios, y el monto total pagado por los clientes desde sus inicios.

4.3.9.3 Monto facturado vs Monto pagado (Mensual: correspondiente al último trimestre).

Este gráfico consta de dos barras, una que indica el monto total facturado por

la empresa en el mes en curso, y el monto total pagado por los clientes en el mes en curso.

4.3.9.4 Top clientes deudores (por monto adeudado).

Es un gráfico de pie, que contiene el top 10 clientes deudores por monto adeudado, es decir, los 10 clientes que tienen la deuda más alta.

4.3.9.5 Top clientes deudores (por meses adeudados).

Es un gráfico de pie, que contiene el top 10 clientes deudores por la cantidad de meses adeudados, es decir, los 10 clientes que deben más meses.

4.3.9.6 Instalaciones.

Es un gráfico de barras, que contiene una barra, indicando la cantidad de instalaciones realizadas en el mes en curso.

Como se mencionó anteriormente, cada una de estas gráficas tiene como finalidad y le permiten al departamento administrativo conocer el comportamiento financiero de la empresa, además de estar al tanto de las operaciones realizadas en esta en distintos períodos de tiempo manejados y evaluar el progreso.

Cada uno de los gráficos tiene su respectiva tabla con la información que este representa. Esto con el fin de permitir ver el detalle de los datos mostrados en cada uno de estos gráficos

4.3.10 Módulo de descarga de Reportes

Se desarrolló una sección en los módulos de clientes, proveedores, cuentas por cobrar y cuentas por pagar que permite generar reportes en formato PDF y Excel con una lista de todos los registros contenidos en estos módulos.

Para la exportación de reportes en PDF, se crea un archivo con este formato con la lista de registros ordenados según aparecen en el sistema. Se tomó en cuenta la generación de estos reportes, ya que la información puede ser revisada y transportada garantizando que sus datos no serán corrompidos ni modificados.

Para la exportación de reportes en Excel, se genera un archivo con extensión .xls con la lista de registros ordenados según aparecen en el sistema.

4.3.11 Módulo de generación de Informes

Se desarrolló un módulo capaz de generar informes administrativos, permitiendo adjuntar gráficas generadas por el sistema en el respectivo módulo o cualquier otra imagen que se desee, y añadir un análisis realizado por el departamento administrativo de la data contenida en las gráficas o imágenes seleccionadas.

Luego de generar un informe, este puede ser exportado en formato PDF, para que pueda ser manejado por el personal del departamento administrativo fácilmente.

Aportes Funcionales

4.3.12 Rediseñar proceso de cobranza de Elemétrica.

Anteriormente, el departamento administrativo de la empresa debía generar a inicios de mes una lista con los clientes y proveedores junto con sus datos principales, y día a día registrar los pagos a medida que los clientes iban realizándolos, agregando el pago al respectivo cliente en la lista ya generada.

Esto requería que los administradores invirtiesen grandes cantidades de tiempo en identificar cuáles son los clientes que mantienen compromisos de pago con la empresa, ya que implica revisar los movimientos de cuentas bancarias y seleccionar los clientes que se encuentran al día con sus pagos y aquellos que no. En el apéndice A se muestra el diagrama de actividades del antiguo proceso de cobranza.

Se rediseñó el proceso de cobranza, incluyendo un módulo en donde los clientes de la empresa puedan registrar sus pagos.

Los clientes registrados en el sistema, podrán crearse una cuenta en este y registrar sus pagos pendientes, añadiendo un comprobante, que en el caso de haber realizado el pago mediante una transferencia bancaria será el archivo PDF generado por el banco, y en el caso de un depósito bancario, una imagen del recibo de pago entregado por el banco.

Además de adjuntar el comprobante, deberán rellenar los datos de su pago, y automáticamente al registrarlo.

Al registrar el pago, el sistema le enviará un correo electrónico con esta información e instrucciones adicionales en el caso de que haya pagado un monto mayor o menor al cobrado.

En este módulo, los clientes también podrán visualizar todas las cuentas que la empresa les ha generado para que estos puedan llevar un control detallado de todos los pagos que han realizado. En el apéndice A se muestra el diagrama de actividades del nuevo proceso de cobranza.

Los comprobantes de pago registrados por los clientes, son almacenados por el servidor en una carpeta compartida para que luego puedan ser revisados por el personal del departamento administrativo y puedan verificar los pagos realizados por los clientes.

4.3.13 Rediseñar el proceso de soporte a los clientes de Elemétrica.

Anteriormente el Departamento de Operaciones de la empresa, quien es el encargado de prestarles soporte técnico a los clientes, prestaba soporte sin tener en cuenta el estado de solvencia de los clientes. Esto traía como consecuencia que clientes que no se encontraban al día con sus pagos pudiesen ser atendidos antes que aquellos que si lo estaban.

El orden de atención y soporte que el departamento aplicaba era el siguiente:

1. Locales comerciales internacionales
2. Cadenas de locales comerciales
3. Resto de locales comerciales que no pertenecen a ninguna de las categorías mencionadas anteriormente.

En el apéndice B se muestra el diagrama de actividades del antiguo proceso de soporte.

Se rediseñó el proceso de soporte a los clientes de la empresa, ya que se configuró el aplicativo web Admin, que como fue mencionado anteriormente es utilizado por el departamento encargado del soporte de la empresa para validar el estado de conexión de cada uno de los locales con el servidor de Elemétrica y verificar si hay algún retraso en la carga de ventas en tiempo real de algún local.

En el aplicativo web Admin, en la sección donde se encuentra la lista de los locales que tienen problemas de conexión con el servidor o presentan retraso en la carga de sus ventas se añadió una columna con la última fecha de pago de cada local contenido en esta lista.

Esto con el fin de que el Departamento de Soporte priorice el servicio no sólo en base al tipo de problema que presentan los locales, sino a su última fecha de pago, es decir, aquellos locales que se encuentren con el mismo tipo de problema, serán atendidos de primero los locales que se encuentren al día con sus pagos y luego aquellos que no lo estén.

Esto con el fin de brindarles un mejor servicio a los clientes de la empresa, garantizando que aquellos que estén comprometidos con sus pagos podrán ser tomados en cuenta más rápidamente.

Luego del rediseño, el orden de atención y soporte es el siguiente:

1 Locales comerciales internacionales

2 Cadenas de locales comerciales solventes con la empresa.

3 Cadenas de locales comerciales insolventes con la empresa.

4 Resto de locales comerciales solventes con la empresa que no pertenecen a ninguna de las categorías mencionadas anteriormente.

5 Resto de locales comerciales insolventes con la empresa que no pertenecen a ninguna de las categorías mencionadas anteriormente.

Si el periodo de insolvencia con la empresa es prolongado, el departamento comercial puede tomar la decisión de suspender el cliente. En el apéndice B se muestra el diagrama de actividades del nuevo proceso de soporte.

Aportes Tecnológicos

4.3.14 Evaluación de tres Frameworks de desarrollo de aplicaciones web para el desarrollo del sistema administrativo.

En la fase de análisis de requisitos de la metodología (Ver punto 4.3.1). Se realizó una evaluación de tres Frameworks de desarrollo de aplicaciones web. Se determinó cual era el Framework adecuado para realizar el proyecto.

Se le hizo una serie de preguntas a cinco desarrolladores y a cinco gerentes de la empresa Elemétrica luego de haberse documentado sobre las ventajas y desventajas de estos tres Frameworks para el desarrollo de aplicaciones web. A los cuales ellos respondieron de acuerdo a su criterio y experiencia, y en base a los resultados obtenidos se determinó que el Framework más adecuado para desarrollar el proyecto era Ruby on Rail. En el apéndice C se muestra la evaluación realizada a los empleados de la empresa.

4.3.15 Integrar sistema de gestión administrativa con el portal web Admin de la empresa

Con el fin de que el aplicativo web Admin de la empresa pudiese mostrar la fecha del último pago de aquellos locales con problemas de conexión o retraso en la carga de datos, se incorporó un api *RESTFul*[7] al sistema de cobranza que fuese capaz de suministrarle esta información al aplicativo.

El API responde en formato JSON y fue desarrollado de modo que solo responda a peticiones GET previa autentificación por medio de tokens.

Para el desarrollo de esta API se hizo uso de JBuilder[8], el cual permite crear de forma sencilla fuentes de datos complejas en formato JSON.

Cada cliente devuelto en la lista posee el localId, y la última fecha de pago. De esta manera el aplicativo web identifica al cliente por medio del localId y obtiene la última fecha de pago de este para posteriormente mostrar en la sección de soporte, los clientes que presenten problemas de conexión con el servidor de Elemétrica o retrasos en la carga de factura junto con la fecha de su pago más reciente.

De esta manera el departamento encargado del soporte puede priorizar en base al estado de solvencia de cada cliente.

---

[7] **RESTFul** : "Es un estilo arquitectónico que especifica restricciones" (Oracle, 2013)

[8] JBuilder: Jbuilder es un motor de plantillas que sirve para generar respuestas en JSON. Fue creada por David Heinemeier Hansson. (Railscast, 2012)

La ilustración 3 muestra el proceso de solicitud de datos por parte del aplicativo Admin:

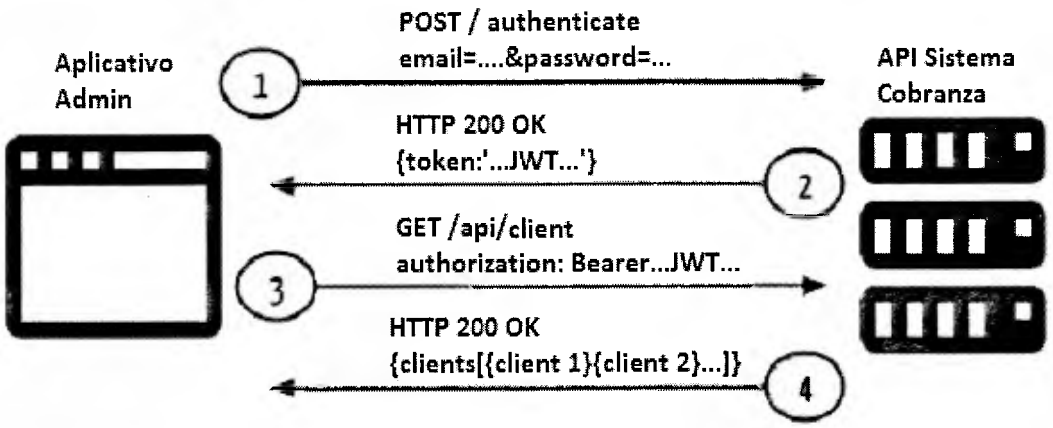

ILUSTRACIÓN 3 PROCESO DE AUTENTIFICACIÓN
FUENTE: ELABORACIÓN PROPIA

1. El aplicativo web envía una solicitud de autenticación al servicio RESTFul con un correo y una contraseña.
2. El servicio RESTFul valida los datos del usuario y envía un token de autentificación.
3. La aplicación cliente hace la petición GET y envía en la cabecera HTTP las credenciales (correo y token).
4. El servicio RESTFul valida las credenciales y envía la lista de clientes.

Capítulo V – Resultados

Desarrollar un sistema administrativo para manejar las cuentas de los clientes y proveedores de Elemétrica.

Se desarrolló una aplicación web capaz de manejar los distintos clientes y proveedores de la empresa, junto con sus cuentas por cobrar y pagar respectivamente. (Ver imágenes de la aplicación en el apéndice F)

A continuación, se describen los resultados obtenidos por objetivos:

Objetivo:

- Diseñar e implementar el modelo de base de datos para el sistema.

Resultados:

Para el desarrollo de este objetivo, se recopilaron todos los datos e información necesaria para cumplir con los objetivos de este trabajo de grado. Se representó a través de un modelo entidad-relación toda la información recopilada anteriormente (Ver apéndice E). Se utilizó como manejador de Base de Datos sqlite3.Ver Cap. IV punto 4.2.3

Objetivo:

- Desarrollar Módulo de registro para el portal web

Resultados:

Se desarrolló un módulo que permitiese que los distintos usuarios del sistema (personal administrativo de la empresa y los clientes de la misma) pudiesen registrarse para acceder y manejar el proceso de cobranza en el caso del personal del departamento y registrar sus pagos en el caso de los clientes de la empresa. Este módulo se explica con más detalle en el Cap. IV punto 4.3.1

Objetivo:

- Desarrollar de carga y Visualización de datos provenientes de archivos de movimientos de cuentas

Resultados:

Se desarrolló una funcionalidad que permitiese importar masivamente clientes, proveedores, cuentas por cobrar y cuentas por pagar de la empresa. Esta funcionalidad se encuentra en cada uno de los módulos correspondientes y permite a los usuarios importar un archivo Excel con una lista con información correspondiente. Esto permite registrar en el sistema múltiples registros simultáneamente, o editarlos si estos registros ya existen en el sistema de cobranza. Este módulo se explica con más detalle en el Cap. IV punto 4.3.2

Objetivo:

- Desarrollar módulo para el manejo de clientes de Elemétrica

Resultados:

Se desarrolló un módulo que permitiese manejar los clientes de la empresa, en este módulo se puede registrar, editar, ver o eliminar clientes. En esta sección

del sistema, se manejan todos los datos necesarios para poder llevar un control de los clientes incluyendo los datos necesarios para poder generar las cuentas por cobrar a cada uno de estos. Este módulo se explica con más detalle en el Cap. IV punto 4.3.3

Objetivo:

- Desarrollar módulo para el manejo de proveedores de Elemétrica

Resultados:

Se desarrolló un módulo que permitiese manejar los proveedores de la empresa, en este módulo se puede registrar, editar, ver o eliminar proveedores. En esta sección del sistema, se manejan todos los datos necesarios para poder llevar un control de los proveedores incluyendo los datos necesarios para poder generar las cuentas por pagar a cada uno de estos. Este módulo se explica con más detalle en el Cap. IV punto 4.3.4

Objetivo:

- Desarrollar módulo para el manejo de cuentas por cobrar de Elemétrica

Resultados:

Se desarrolló un módulo que permitiese manejar las cuentas por cobrar de la empresa, en este módulo se puede registrar, editar, ver o eliminar cuentas por cobrar de cada uno de los clientes. En esta sección del sistema, se manejan todos los datos necesarios para poder llevar un control de las cuentas por cobrar. Este módulo se explica con más detalle en el Cap. IV punto 4.3.5

Objetivo:

- Desarrollar módulo para el manejo de cuentas por pagar de Elemétrica

Resultados:

Se desarrolló un módulo que permitiese manejar las cuentas por pagar de la empresa, en este módulo se puede registrar, editar, ver o eliminar cuentas por pagar de cada uno de los proveedores. En esta sección del sistema, se manejan todos los datos necesarios para poder llevar un control de las cuentas por pagar. Este módulo se explica con más detalle en el Cap. IV punto 4.3.6

Objetivo:

- Desarrollar módulo de configuraciones del sistema administrativo

Resultados:

Se desarrolló un módulo capaz de manejar las configuraciones del sistema y parámetros generales de éste. En este módulo se puede establecer la frecuencia de los correos electrónicos enviados por el sistema para el recordatorio de pago a los clientes, además se pueden manipular los descuentos que poseen la empresa, las tarifas que maneja y modificar los roles de cada uno de los usuarios del sistema. Este módulo se explica con más detalle en el Cap. IV punto 4.3.7

Objetivo:

- Desarrollar módulo para el manejo de envío de correos electrónicos

Resultados:

Se desarrolló un módulo capaz de enviar y almacenar correos electrónicos. Este módulo permite enviar correos con información relacionada a pagos y servicios prestados por la empresa. Los correos electrónicos se envían a cada una de las direcciones de correo que posea el cliente seleccionado, buscando informar a todos los involucrados. También permite enviar correos masivos. Este módulo se explica con más detalle en el Cap. IV punto 4.3.8

Objetivo:

- Desarrollar módulo de estadísticas y gráficas

Resultados:

Se desarrolló un módulo capaz de generar gráficas que representen la información manejada por el sistema, permitiendo al personal de departamento administrativo de la empresa tomar decisiones y observar cómo se comporta el proceso de cobranza en la empresa. Estos gráficos son generados en varios períodos de tiempo (Mensual, trimestral e Histórico), para poder analizar de una mejor manera el comportamiento de los datos. Se manejan distintos tipos de gráficos en el sistema, dependiendo del tipo de datos a representar. Este módulo se explica con más detalle en el Cap. IV punto 4.3.9

Objetivo:

- Desarrollar módulo de descarga de reportes

Resultados:

Se desarrolló un módulo capaz de descargar reportes en formato PDF y Excel, con una lista de clientes, proveedores, cuentas por cobrar y cuentas por pagar contenidas en el sistema, con el fin que el personal de departamento administrativo pueda usar en reuniones con inversionistas y personal ejecutivo. Estos reportes están compuestos por todos los datos de los clientes, proveedores, cuentas por cobrar o pagar. Este módulo se explica con más detalle en el Cap. IV punto 4.3.10

Objetivo:

- Desarrollar módulo de generación de informes

Resultados:

Se desarrolló un módulo capaz de generar informes en formato PDF, con el fin que el personal de departamento administrativo pueda usar en reuniones con inversionistas y personal ejecutivo. Estos gráficos están compuestos por el título, permite adjuntar un gráfico del sistema exportado con anterioridad o cualquier otra imagen que se desee y el análisis del gráfico adjuntado. Este módulo se explica con más detalle en el Cap. IV punto 4.3.11

Aportes Funcionales:

Rediseñar el proceso de cobranza a los clientes de Elemétrica.

Se rediseñó el proceso de cobranza a los clientes de la empresa, al desarrollar una sección en donde los clientes puedan registrar sus pagos.

Para que los clientes puedan ver o registrar sus pagos deben crearse una cuenta en el sistema, compuesta de un correo y una contraseña. Únicamente los clientes registrados en el sistema por el departamento administrativo tendrán la posibilidad de establecer una contraseña para su cuenta e ingresar al sistema. De modo que, si un tercero quiere ingresar a este, no es posible a menos que contenga las credenciales de un cliente ya registrado. Esto se explica con más detalle en el Cap. IV punto 4.3.12

Rediseñar el proceso de soporte a los clientes de Elemétrica.

Se rediseñó el proceso de soporte a los clientes de la empresa, al permitir la integración del sistema administrativo al aplicativo web Admin disponible en Elemétrica, utilizado por el Departamento de Operaciones para visualizar el comportamiento y carga de datos en tiempo real de cada uno de los locales por los servidores de la empresa. Incorporando la última fecha de pago de cada uno de los clientes. Con el fin de añadir un nuevo criterio de soporte a los ya existentes hasta el momento. Permitiendo atender a aquellos locales los cuales se encuentren al día en sus pagos, antes de los que no lo estén.

Para cumplir con este aporte, se integró al proyecto un API RESTFul, la cual se encarga de responder a las peticiones GET por parte del aplicativo web Admin de la empresa.

Esta autenticación y la validación de credenciales se desarrollaron con el uso de autenticación por Token. Esto se explica con más detalle en el Cap. IV punto 4.3.13

Aporte tecnológico:

Integrar sistema de gestión administrativa con el portal web Admin de la empresa.

Para el cumplimiento de este aporte, se integró un API RESTFul al sistema de cobranza, de modo que el aplicativo web Admin consumiera la información requerida de éste. Se creó una autentificación por medio de tokens para proporcionar seguridad y que la data esté protegida.

Esta integración se realizó con el fin de que el aplicativo web pudiese brindar información de la última fecha de cada local, en la lista de locales que requieren soporte por algún problema de conexión o de carga de datos por parte del servidor de la empresa. Esto se explica con más detalle en el Cap. IV punto 4.3.14

Evaluación de tres Framework de desarrollo de aplicaciones web para el desarrollo del sistema administrativo.

Para el cumplimiento de este aporte, se realizó una evaluación a 5 desarrolladores y 5 gerentes de la empresa, utilizando una escala de ponderaciones

para determinar las ventajas y desventajas del uso de los Frameworks de desarrollo de aplicaciones web. Como resultado de esta evaluación, se concluyó que era conveniente el uso del Framework Ruby on Rails para la implementación del sistema de Cobranza, sin embargo, se determinó que dependiendo del tipo de aplicación a desarrollar, puede ser conveniente el uso del Framework CodeIgniter o Angular Universal. Esto se explica con más detalle en el Cap. IV punto 4.3.15

Capítulo VI – Conclusiones y Recomendaciones

Este capítulo tiene como objetivo mostrar las conclusiones y recomendaciones recopiladas por los objetivos desarrollados a lo largo de este trabajo de grado.

Conclusiones

En el presente Trabajo Instrumental de Grado se propuso desarrollar un Sistema administrativo capaz de manejar los distintos Clientes y Proveedores de Elemétrica, junto con sus cuentas por cobrar y por pagar respectivamente, para hacer de la tarea de la cobranza un proceso más sencillo y reducir el trabajo que este proceso requiere involucrando a los clientes.

A partir de los datos presentados anteriormente se puede concluir lo siguiente:

- El sistema administrativo debe ser manejado por personal del departamento que conozca en su totalidad el proceso de cobranza que se maneja en la empresa, ya que el sistema tiene reglas de negocio específicas, limitando que este sistema pueda ser usado a juicio particular.
- El módulo de registro de pago es capaz de reconocer gran cantidad de comprobantes de pago, pero se restringió a archivos con formatos conocidos, tales como .pdf, .jpg. .png. Debido a que estos son los formatos utilizados por los bancos para generar dichos comprobantes.

- El uso de la metodología en Cascada permitió dividir el desarrollo del proyecto en pequeñas partes medibles para un desarrollo efectivo.
- La aplicación brinda la oportunidad de manejar gran cantidad de aspectos del proceso de cobranza, haciendo que esta actividad pueda realizarse en una única plataforma.
- Se rediseñó el proceso de cobranza al involucrar al cliente en éste. De esta manera se disminuye la tarea de identificación de movimientos bancarios por parte del personal administrativo de la empresa
- Se rediseñó el proceso de soporte a los clientes de Elemétrica al integrar el sistema de cobranza con el aplicativo Web Admin de la empresa. Por medio de la API RESTFul que se incorporó al sistema, el aplicativo Web Admin es capaz de realizar peticiones GET solicitando la lista de los clientes de la empresa en formato JSON con la última fecha de pago de estos. Permitiendo prestar soporte en base al estado de solvencia de los clientes

Recomendaciones

Durante la etapa de diseño, desarrollo y recolección de datos de este trabajo, surgieron distintas recomendaciones que de aplicarse permitirán crear una versión mejorada del sistema presentado.

A continuación, las recomendaciones propuestas:

- Se recomienda desarrollar nuevos módulos, que aumenten la versatilidad del sistema y hagan que el proceso de cobranza sea más completo.

- Se recomienda mejorar el algoritmo de procesamiento de comprobantes de pagos, para que sea más versátil y sea capaz de adaptarse a una mayor variedad de comprobantes incluyendo el procesamiento de datos de las imágenes tomadas por los clientes.
- Se recomienda mejorar la carga masiva de datos de clientes y proveedores, de forma que el algoritmo de procesamiento de archivos permita otro tipo de archivos tales como .txt, archivos .sql, .pdf, entre otros.
- Se recomienda desarrollar una aplicación móvil para que los clientes de la empresa puedan registrar sus pagos más fácilmente.
- Se recomienda implementar otras gráficas en el módulo pertinente que faciliten el análisis de la data al departamento administrativo.

Referencias Bibliográficas

aulaformativa. (2016). Razones para usar la framework Ruby on Rails. 2016, de aulaformativa Sitio web: http://blog.aulaformativa.com/razones-para-usar-la-framework-ruby-on-rails/

Camuña, Jesús F. (2015). Lenguajes de definición y modificación de datos SQL. IFCT0310. España: ic editorial.

Cardenas, Mario. (2016). Aprendiendo Ruby on Rails. Colombia: railsenespañol.co.

definicion.de. (2008). Definición de IVA. 2008, de definicion.de Sitio web: https://definicion.de/iva/

definicionabc. (2007). Definición de Impuesto al valor agregado. 2017, de definicionabc Sitio web: https://www.definicionabc.com/economia/impuesto-al-valor-agregado.php

Microjuris. (2011). Retenciones del impuesto sobre la renta. 2011, de Microjuris.com Sitio web: https://aldiavenezuela.microjuris.com/2011/01/26/retenciones-del-impuesto-sobre-la-renta/

ecured. (2017). Modelo cascada. 2017, de ecured.cu Sitio web: https://www.ecured.cu/Modelo_en_cascada

EllisLab. (2014). CodeIgniter at a Glance. 2016, de codeigniter.com Sitio web: https://codeigniter.com/user_guide/overview/at_a_glance.html

Erii Kusu . (2014). AGENTES DE RETENCIÒN. 2014, de prezi.com Sitio web: https://prezi.com/ivvxcqofdeaf/agentes-de-retencion/

Gilli, Juan José. (1998). Sistemas administrativos. Buenos Aires: Docencia 3era edición.

Google. (2016). Angular universal. 2016, de universal.angular.io Sitio web: https://universal.angular.io/

highcharts.com. (2017). Welcome to the Highcharts Options Reference. 2017, de highcharts.com Sitio web: https://www.highcharts.com/blog/products/highcharts/

IBM. (2010). SMTP (Protocolo simple de transferencia de correo). 2010, de .ibm.com Sitio web: https://www.ibm.com/support/knowledgecenter/es/ssw_i5_54/rzair/rzairemcommnd.htm

ITSON. (2009). Fases del Modelo Cascada. 2009, de fasesmodelocascada.blogspot Sitio web: http://fasesmodelocascada.blogspot.com/

Jimenez, Geraldo. (2013). CODEIGNITER. 2013, de Prezi Sitio web: https://prezi.com/_qyrvujdmxci/codeigniter/

Julián Pérez Porto. (2006). Definición de cobranza. 2017, de definicion.de Sitio web: https://definicion.de/cobranza/

lancetalent. (2012). 6 Buenos Motivos Para Trabajar Con PHP. 2012, de lancetalent Sitio web: https://www.lancetalent.com/blog/6-buenos-motivos-para-trabajar-con-php/

London, Camilo. (2017). GERENCIA Y TRIBUTOS. 2017, de gerenciaytributos Sitio web: https://gerenciaytributos.blogspot.com/

Natali, Nelly. (2009). CSS: ¿QUÉ ES UN LAYOUT?. 2009, de nnatali.com Sitio web: http://blog.nnatali.com/2009/03/10/css-que-es-un-layout/

Nava, Melvin. (2008). ISLR: Impuesto sobre la Renta en Venezuela. 2016, de venelogia.com Sitio web: http://www.venelogia.com/archivos/2239/

Nwamba, Chris. (2016). Server-Side rendering in Angular 2. 2016, de scotch.io Sitio web: https://scotch.io/tutorials/server-side-rendering-in-angular-2-with-angular-universal

Pressman, Roger S.. (2002). Modelos de Procesos. En Ingeniería del software. Un enfoque prático 7ma edición(767). Mexico D.F: Mc Graw Hill.

Railscast. (2012). JBuilder. 2012, de Railscast Sitio web: http://railscasts.com/episodes/320-jbuilder?language=es&view=asciicast

Rodriguez, Manuel. (2016). ¿Qué implicaciones tiene ser un Contribuyente Especial?. 2016, de actualidadvenezuela.org Sitio web: https://actualidadvenezuela.org/2016/11/21/que-implicaciones-tiene-ser-un-contribuyente-especial/

rubyonrails.org.es. (2010). Ruby on Rails: El desarrollo web que no molesta. 2016, de rubyonrails.org.es Sitio web: http://rubyonrails.org.es/

rubyonrails.org. (2010). Guía de seguridad de Ruby on Rails. 2016, de rubyonrails.org.es Sitio web: http://guides.rubyonrails.org/security.html

Sánchez, Jordi . (2006). ¿Qué es un 'framework'?. 2006, de jordisan.net Sitio web: https://jordisan.net/blog/2006/que-es-un-framework/

SENIAT. (2003). Designación de los agentes de retención. 2017, de seniat.gob.ve Sitio web: http://www.ciec.org.ve/documentos/PROVIDENCIACONTRIBUYENTESESPECIALES.pdf

SENIAT. (2013). Sistema de Declaración de Sucesiones . 2013, de SENIAT Sitio web: http://declaraciones.seniat.gob.ve/portal/page/portal/MANEJADOR_CONTENIDO_SENIAT/05MENU_HORIZONTAL/5.1ASISTENCIA_CONTRIBUYENTE/5.1.2ORIENTACION_GENERA/5.1.2.1TRAMITES_ELECTRONI/InstructivoSucesiones.pdf

SENIAT. (2015). Registro Único de Información Fiscal: documento de control tributario. 2015, de SENIAT Sitio web: http://noticias.seniat.gob.ve/index.php/en/noticias-anteriores/825-registro-unico-de-informacion-fiscal-documento-de-control-tributario

sqlite.org. (2000). About SQLite. 2017, de sqlite.org Sitio web: http://www.sqlite.org/about.html. (2000). About SQLite. 2017, de sqlite.org Sitio web: http://www.sqlite.org/about.html

solocontabilidad. (2017). Cuentas por CObrar. 2017, de solocontabilidad Sitio web: http://www.solocontabilidad.com/tema/cuentas-por-cobrar

tributos.net. (2017). DEFINICIÓN DE CUENTA POR PAGAR. 2017, de tributos.net Sitio web: https://www.tributos.net/definicion-de-cuenta-por-pagar-813/

Trujillo, Felipe. a. (2014). Entendiendo el Patrón MVC (Modelo, Vista, Controlador). 2014, de andresfelipetrujillo.com Sitio web: https://andresfelipetrujillo.com/2014/08/07/entendiendo-el-patron-mvc-modelo-vista-controlador/

ukic Ado . (2016). Angular 2 Authentication. 2016, de dzone Sitio web: https://dzone.com/articles/angular-2-authentication-tutorial

Welcome to Rails. (2003). Welcome to Rails. 2017, de api.rubyonrails.org Sitio web: http://api.rubyonrails.org/

w3ii.com. (2017). CodeIgniter Seguridad. 2017, de w3ii Sitio web: http://www.w3ii.com/es/codeigniter/codeigniter_security.html

Apéndices

Apéndice A – Diagramas del proceso de Cobranza

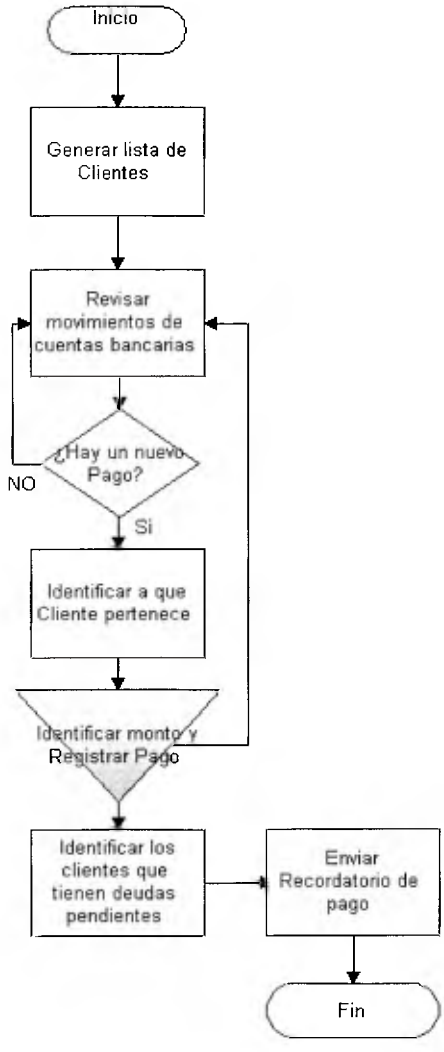

ILUSTRACIÓN 4 DIAGRAMA DE FLUJO DEL ANTIGUO PROCESO DE COBRANZA

Fuente: Elaboración propia.

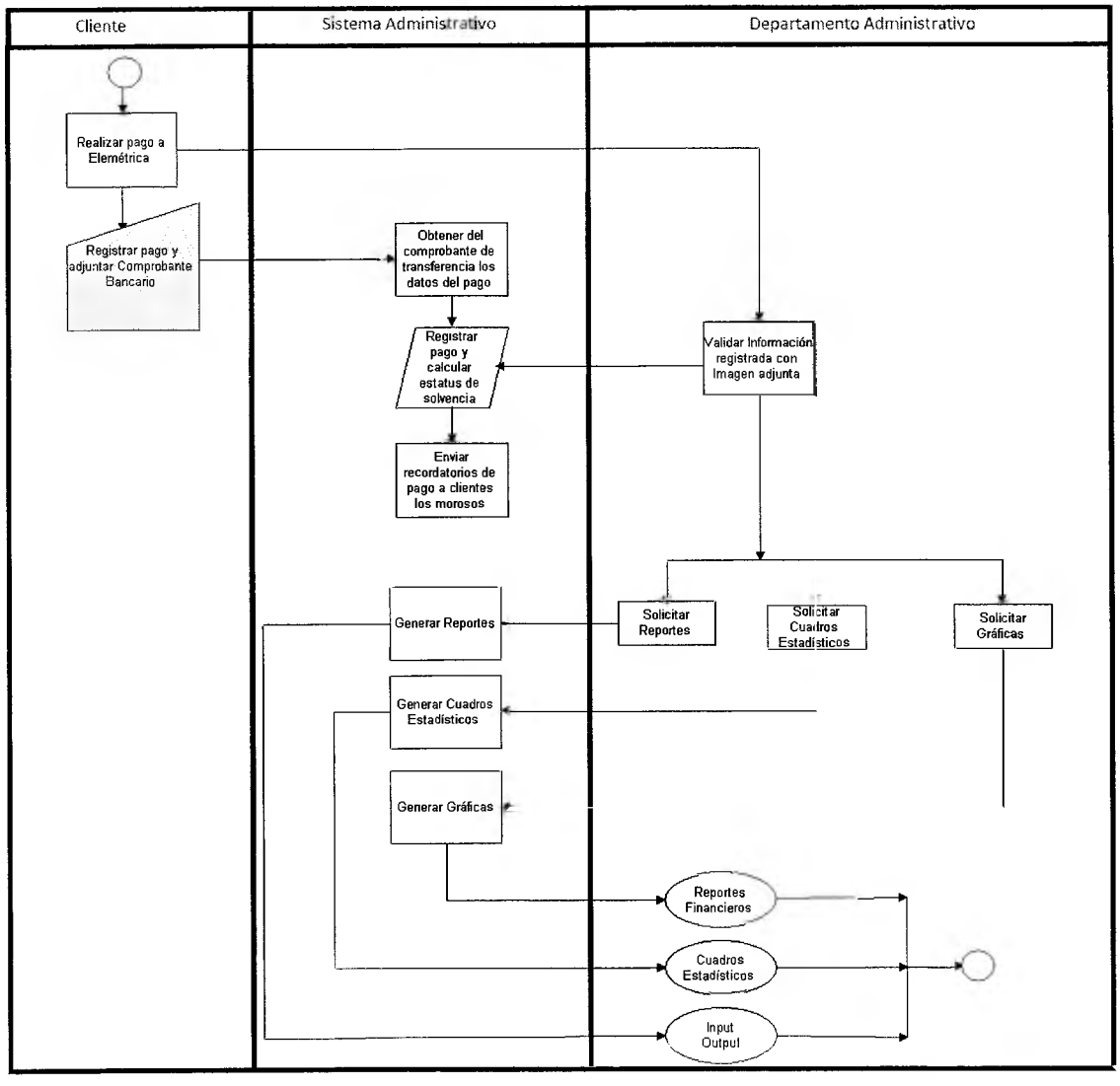

ILUSTRACIÓN 5 DIAGRAMA DE ACTIVIDADES DEL NUEVO PROCESO DE COBRANZA
Fuente: Elaboración propia

Apéndice B – Diagramas del proceso de Soporte

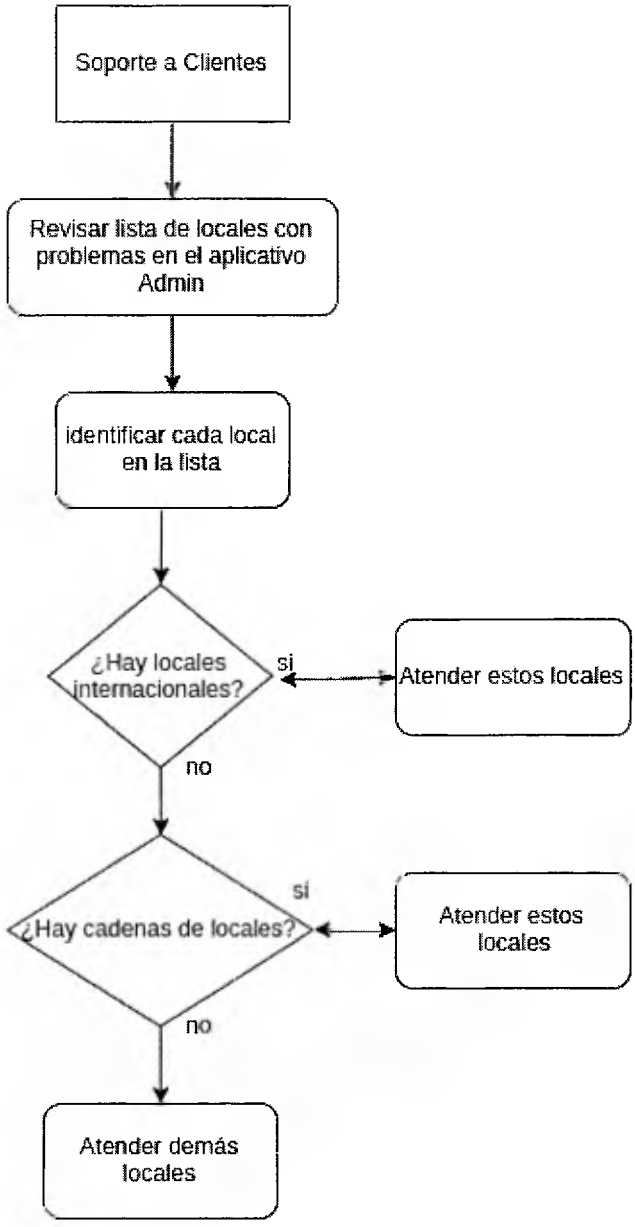

ILUSTRACIÓN 6 DIAGRAMA DE FLUJO DEL ANTIGUO PROCESO DE SOPORTE
Fuente: Elaboración propia

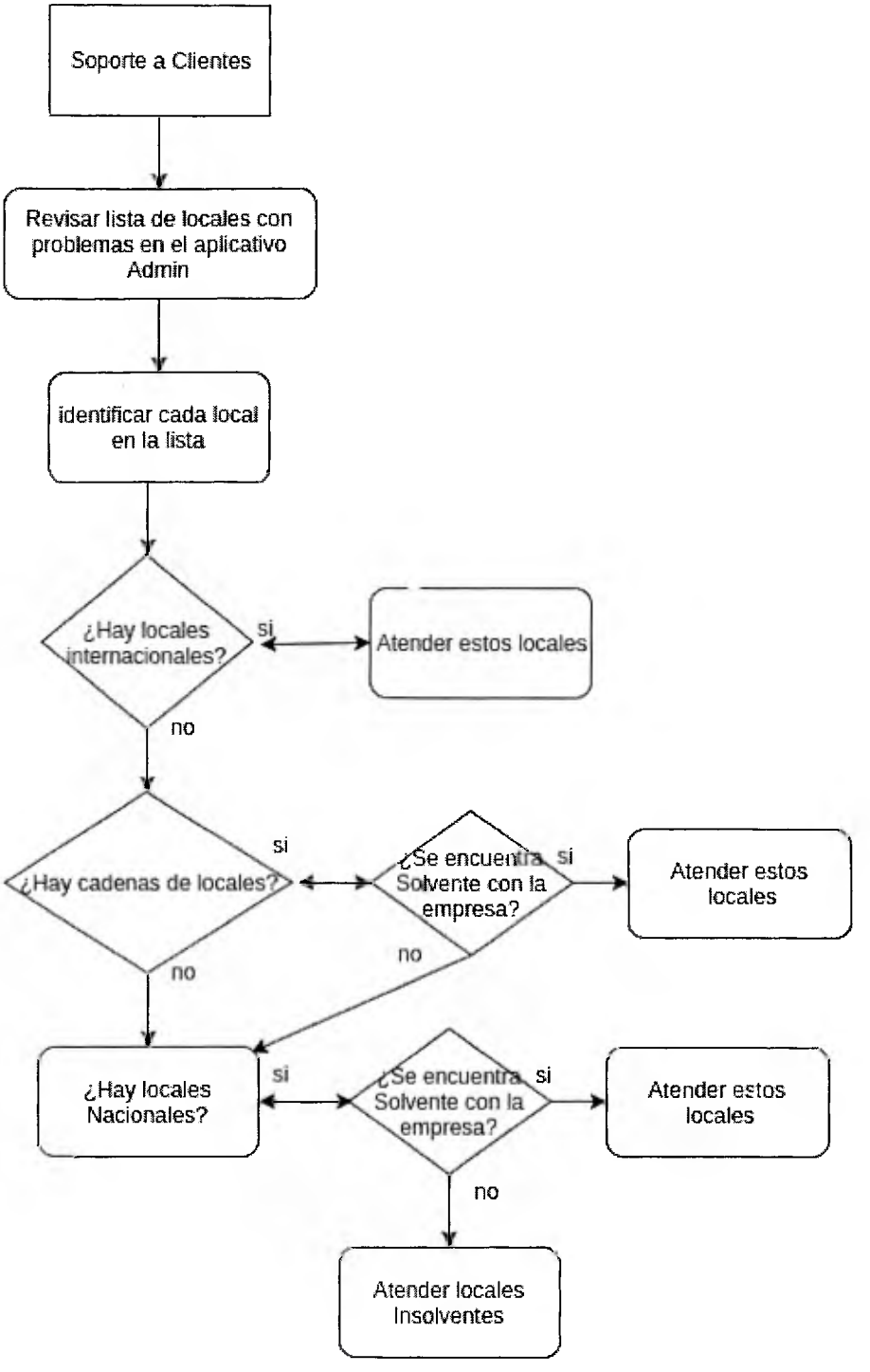

ILUSTRACIÓN 7 DIAGRAMA DE FLUJO DEL NUEVO PROCESO DE SOPORTE
Fuente: Elaboración propia

Apéndice C – Evaluación de Framework de desarrollo de aplicaciones web

Un Framework es un marco de trabajo que sirve para poder desarrollar o escribir código de una manera más sencilla, permitiendo llevar un orden y control a la hora de desarrollar. Una de las ventajas que proporcionan los Frameworks es que permiten reutilizar código, generando así una mayor productividad, capacidad para minimizar costos en cuanto al desarrollo y reducir en gran medida los errores que pueden producirse al desarrollar software. Para la evaluación de Frameworks de desarrollo de aplicaciones web, se decidió tomar como comparación 3 Frameworks de desarrollo.

- Framework Ruby on Rails
- Framework Angular Universal
- Framework CodeIgniter

Se decidió evaluar estos 3 Frameworks de desarrollo de aplicaciones web en dos tipos de experiencias que se explican a continuación:

- Experiencia para el desarrollador: percepciones que se tendrán durante el desarrollo de la aplicación web.
- Experiencia para el proyecto: efectos que se tendrán en el proyecto Sistema de Cobranza

TABLA 4 COMPARACIÓN CRITERIOS DE EVALUACIÓN

Criterio	Framework Ruby on Rails	Framework Angular Universal	Framework CodeIgniter
Filosofía	La filosofía de este Framework está orientada a no repetir código, y convención antes de configuración. "Proporciona la posibilidad de desarrollar aplicaciones del mundo real escribiendo menos código que con otros Frameworks y con un mínimo de configuración". (Cardenas, 2016)	La filosofía de este Framework es hacer que el desarrollo de las aplicaciones tenga una experiencia agradable (Nwamba, 2016)	La filosofía de este Framework está orientada a permitir desarrollar proyectos rápidamente proporcionando una gran variedad de librerías para tareas comunes, permitiendo realizar una estructura lógica e interfaces sencillas "permite centrarse creativamente en tu proyecto al minimizar la cantidad de código necesario para una tarea determinada". (EllisLab, 2014)
Sostenibilidad	Rails fue creado en 2003 por David Heinemeier Hansson y desde entonces ha sido extendido por el Rails core team, más de 2.100 colaboradores y soportado por una extensa y activa comunidad. (rubyonrails.org.es, 2010)	La segunda versión de Angular fue lanzada oficialmente en septiembre del año 2016. Esta versión cuenta con novedades y funcionalidades como modularidad, que facilitan el uso de librerías externas, actualizaciones a través de 3 versiones semánticas, herramientas para móviles e interface para línea de comandos	La primera versión pública de CodeIgniter fue lanzada el 28 de febrero de 2006 . Este Framework proporciona facilidad en el proceso de instalación, flexibilidad, ligereza, compatibilidad en cualquier servidor, entre otras características. (Jimenez, 2013)
Documentación	Posee una gran cantidad de foros y listas de correo no solo en inglés, sino también en otros lenguajes incluyendo el español	Por ser un Framework relativamente nuevo no posee gran cantidad de fuentes de documentación como los otros dos	La comunidad de CodeIgniter cuenta con un foro configurado con MyBB. Este foro tiene como objetivo que los

	como ror-es (Ruby on Rails en Español).	Frameworks a evaluar. En este sentido la información es limitada. La comunidad es más reducida que los otros dos Frameworks a comparar	programadores puedan pedir ayuda o discutir problemas que tienen con el marco de trabajo. Este foro se archiva en el servidor de EllisLab. El Framework cuenta con variedad de foros en la red.
Seguridad	Posee métodos auxiliares inteligentes, contra la inyección sql, css, ajax, permite manejar sesiones para los usuarios que ingresan al sistema, esta sesiones consiste en un hash de valores y un ID de sesión. Existen varios complementos de autenticación disponibles para Rails. (rubyonrails.org, 2010)	Posee métodos para el manejo de la seguridad mediante la autentificación basada en tokens, permitiendo controlar los accesos a las aplicaciones desarrolladas. (Ado, 2016)	Posee Prevención XSS (cross-site scripting) para evitar inyección de código javascript malicioso, previene la inyección sql, proporciona mecanismos para la prevención CSRF (falsificación de petición de sitios cruzados). (w3ii.com, 2017)
Curva de Aprendizaje	Este Framework está basado en el lenguaje de programación Ruby el cual es un lenguaje de programación dinámico y de código abierto enfocado en la simplicidad y productividad. Rails es "Un conjunto de librerías, automatismos y convenciones destinados a resolver los problemas más comunes a la hora de desarrollar una aplicación web, para que el programador pueda concentrarse en los aspectos únicos y	Este Framework está basado en el lenguaje angular 2, el cual es relativamente nuevo. El Framework proporciona una vista renderizada del servidor que mejora en gran medida el rendimiento percibido y la experiencia general del usuario. Según la investigación de Google , la diferencia de sólo 200 milisegundos en el rendimiento de la carga de la página tiene un impacto en el comportamiento del usuario. (Gogle, 2016)	Este Framework está basado en el lenguaje de programación PHP el cual es un lenguaje totalmente libre y abierto. Tiene una curva de aprendizaje muy baja, su sintaxis es simple y cumple estándares básicos de la programación orientada a objetos. (lancetalent, 2012)

	diferenciales de su proyecto en lugar de los problemas recurrentes." (rubyonrails.org.es, 2010)		

Fuente: elaboración propia.

Para la evaluación de estos criterios, se tomó como escala de calificación la siguiente:

TABLA 5 ESCALA DE CALIFICACIÓN PARA LA EVALUACIÓN DEL USO DE "FRAMEWORKS" PARA LA IMPLEMENTACIÓN DEL SISTEMA DE COBRANZA.

Efectividad	Calificación	Definición
Excelente	4	Siempre supera la satisfacción del criterio evaluado.
Muy bueno	3	A veces supera la satisfacción del criterio evaluado.
Bueno	2	Satisface el criterio evaluado.
Regular	1	Satisface el criterio evaluado con dificultades.
Malo	0	No satisface el criterio evaluado.

Fuente: elaboración propia.

A continuación, se explica el método de evaluación de cada experiencia y sus criterios a evaluar.

Experiencia para el desarrollador: se le hizo una serie de preguntas a 5 desarrolladores de la empresa Elemétrica luego de haberse documentado sobre las ventajas y desventajas de estos 3 Frameworks para el desarrollo de aplicaciones web. En esta experiencia se evaluaron los siguientes criterios.

- Filosofía: Este criterio permite asegurarse que el Framework se adapte a las necesidades. Una herramienta desarrollada por profesionales para sus propias necesidades y que cubra las demandas de otros profesionales.

  "En una escala del 0 (mala) al 4(excelente) evalué la filosofía de estos 3 frameworks de desarrollo web

  Framework Ruby on Rails: _____

  Framework Angular Universal: _____

  Framework CodeIgniter: _____"

- Sostenibilidad: Este criterio permite evaluar la capacidad de los Frameworks de mantenerse al día a través del tiempo, tanto en el mantenimiento como en la modernización de las librerías y componentes.

  "En una escala del 0 (mala) al 4 (excelente) evalué la sostenibilidad de estos 3 Frameworks de desarrollo web

  Framework Ruby on Rails: _____

Framework Angular Universal: _____

Framework CodeIgniter: _____"

- Documentación: Este criterio permite evaluar la facilidad de encontrar respuestas a preguntas y obtener ayuda. Identifica el soporte que está disponible, desde el propio desarrollador, una comunidad e incluso empresas de servicio (desarrollo, apoyo, formación).

"En una escala del 0 (mala) al 4 (excelente) evalué la calidad y cantidad de documentación que posee cada Framework de desarrollo web:

Framework Ruby on Rails: _____

Framework Angular Universal: _____

Framework CodeIgniter: _____"

- Seguridad: Este criterio permite evaluar la vulnerabilidad de los Frameworks de desarrollo web.

"En una escala del 0 (mala) al 4 (excelente) evalué la seguridad que proporciona cada Framework de desarrollo web:

Framework Ruby on Rails: _____

Framework Angular Universal: _____

Framework CodeIgniter: _____"

- Curva de aprendizaje: Este criterio permite evaluar el tiempo requerido para que el desarrollador llegue a dominar la herramienta de desarrollo.

  "En una escala del 0 (mala) al 4 (excelente) evalué la curva de aprendizaje para el tipo de desarrollo indicado:

  Framework Ruby on Rails: _____

  Framework Angular Universal: _____

  Framework CodeIgniter: _____"

Experiencia para el proyecto: se le hizo una serie de preguntas a 5 gerentes de la empresa Elemétrica luego de haberse documentado sobre las ventajas y desventajas de estos 3 Frameworks para el desarrollo de aplicaciones web. En esta experiencia se evaluaron los siguientes criterios.

- Fácilmente escalable: se define como la facilidad de agregar nuevas funcionalidades a la aplicación en un futuro. Se le indicó al gerente del Departamento de Tecnología de la empresa que respondiera la siguiente pregunta:

"En una escala del 0 (mala) al 4 (excelente) evalué la facilidad de agregar nuevas funcionalidades a la aplicación para el tipo de desarrollo indicado:

Framework Ruby on Rails: _____

Framework Angular Universal: _____

Framework CodeIgniter: _____"

- Integración con el negocio: se define como las librerías que presta el tipo de desarrollo para la creación de la aplicación web del sistema de Cobranza. Se le indicó al gerente del Departamento de Tecnología que respondiera la siguiente pregunta:

  "En una escala del 0 (mala) al 4 (excelente) evalué la integración con el negocio para el tipo de desarrollo indicado:

  Framework Ruby on Rails: _____

  Framework Angular Universal: _____

  Framework CodeIgniter: _____"

En la Tabla 6 se muestra la matriz donde se puede observar el promedio de las evaluaciones obtenidas en cada criterio a evaluar.

TABLA 6 *TABLA DE PONDERACIÓN PARA LA EVALUACIÓN DEL EL USO DE "FRAMEWORKS" PARA LA IMPLEMENTACIÓN DEL SISTEMA DE COBRANZA.*

Criterio a Evaluar	Framework Ruby on Rails	Framework Angular Universal	Framework CodeIgniter
Filosofía	3	2	3
Sostenibilidad	4	3	3
Documentación	3	2	3
Seguridad	3	2	4
Curva de Aprendizaje	4	3	3
Escalabilidad	3	3	4
Integración con el Negocio	4	2	3

Fuente: elaboración propia

Para obtener una ponderación final para cada Framework de desarrollo, se decidió sumar los puntajes de cada experiencia, el promedio de las calificaciones obtenidas en los criterios a evaluar, es decir, se aplicó la siguiente fórmula:

$$Ponderación\ final\ del\ tipo\ de\ desarrollo\ = \sum_{Experiencia} PROM(CCEE)$$

CCEE = calificación de criterio a evaluar de experiencia

En la tabla 6 se muestra la matriz donde se puede observar las ponderaciones finales de cada Framework.

TABLA 7 PONDERACIÓN FINAL DE EVALUACIÓN DEL USO DE "FRAMEWORKS" PARA LA IMPLEMENTACIÓN DEL APLICATIVO WEB

Framework	Ponderación final
Ruby on Rails	3.6
Angular Universal	2.4
CodeIgniter	3.2

Fuente: elaboración propia

Apéndice D – Historias de usuario

Historia #1 – Manejo de clientes de la empresa y sus cuentas por cobrar.

El sistema de cobranza debe permitir manejar los clientes de la empresa y las cuentas por cobrar de estos

Condiciones de satisfacción:

- Una vez realizada la autenticación en la aplicación, la aplicación debe poseer una sección donde se muestre información general de todos los clientes de la empresa.
- El sistema debe poseer una sección con las cuentas por cobrar de cada uno de los clientes registrados en el sistema

- En la sección de clientes se podrá consultar, agregar, editar, y eliminar clientes de la empresa, al igual que cada una de sus cuentas. Además, estas secciones deben poseer filtros de ordenamiento y de búsqueda para su fácil ubicación.

Historia #2 – Manejo de proveedores de la empresa y sus cuentas por pagar.

El sistema de cobranza debe permitir manejar los proveedores de la empresa y las cuentas por pagar a cada uno de estos.

Condiciones de satisfacción:

- Una vez realizada la autenticación en la aplicación, la aplicación debe poseer una sección donde se muestre información general de todos los proveedores de la empresa.
- El sistema debe poseer una sección con las cuentas por pagar a cada uno de los proveedores registrados en el sistema

  En la sección de proveedores se debe poder consultar, agregar, editar, y eliminar proveedores de la empresa, al igual que cada una de sus cuentas. Además, estas secciones deben poseer filtros de ordenamiento y de búsqueda para su fácil ubicación.

Historia #3 – Generación de gráficas.

El sistema de cobranza debe permitir generar gráficas manejadas por el sistema administrativo para la visualización de la data del proceso de cobranza

Condiciones de satisfacción:

- El sistema debe contener una serie de secciones con cada una de las gráficas manejadas por el departamento administrativo.
- Los periodos de tiempo manejados para las gráficas que lo requieran deben ser mensual, trimestral y el histórico.

Historia #4 – Envío de correos electrónicos a los clientes de la empresa.

El sistema de cobranza debe permitir enviar correos electrónicos a los distintos clientes de la empresa

Condiciones de satisfacción:

- El sistema debe contener sección en donde se pueda seleccionar a que cliente se le desea enviar un correo Electrónico, asignar un título y contenido al correo.
- El módulo debe permitir enviar correos electrónicos masivos a los clientes de la empresa

Historia #5 – Generación de informes.

El sistema de cobranza debe permitir generar informes con información del sistema de cobranza.

Condiciones de satisfacción:

- El sistema debe contener sección en donde se pueda generar informes con análisis realizados por el departamento administrativo de la empresa
- Los informes deben poder exportarse en archivos con formato PDF.
- El sistema debe permitir incorporar gráficas a estos informes.

Historia #6 – Generación de recordatorios de pago.

Como usuario del sistema debo poder configurar los recordatorios de pago

Condiciones de satisfacción:

- El sistema debe contener sección en donde se pueda configurar el envío de recordatorios de pago a los clientes de la empresa
- La frecuencia de los recordatorios debe ser diaria, quincenal o mensual

Historia #7 – Importación de lista de data.

Como usuario del sistema de cobranza quiero poder importar data masivamente

Condiciones de satisfacción:

- Al ingresar en los módulos de clientes, proveedores, cuentas por cobrar y por pagar debe haber una opción para importar grandes cantidades de data masivamente
- El sistema debe tener la capacidad de indicar el estado de la importación

Apéndice E – Modelo Entidad-Relación

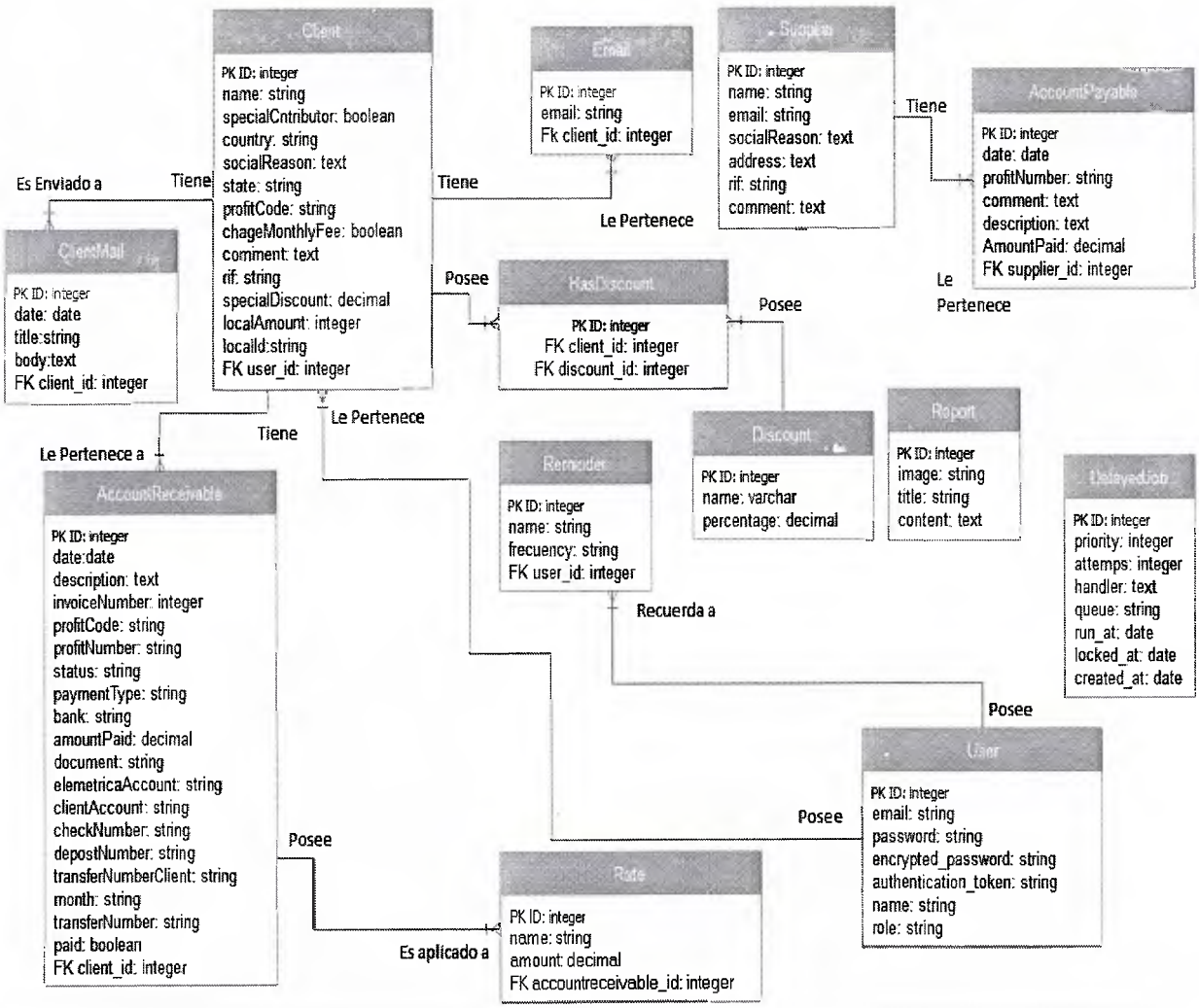

ILUSTRACIÓN 8 MODELO ENTIDAD RELACIÓN
FUENTE: ELABORACIÓN PROPIA

Apéndice F– Pantallas del sistema

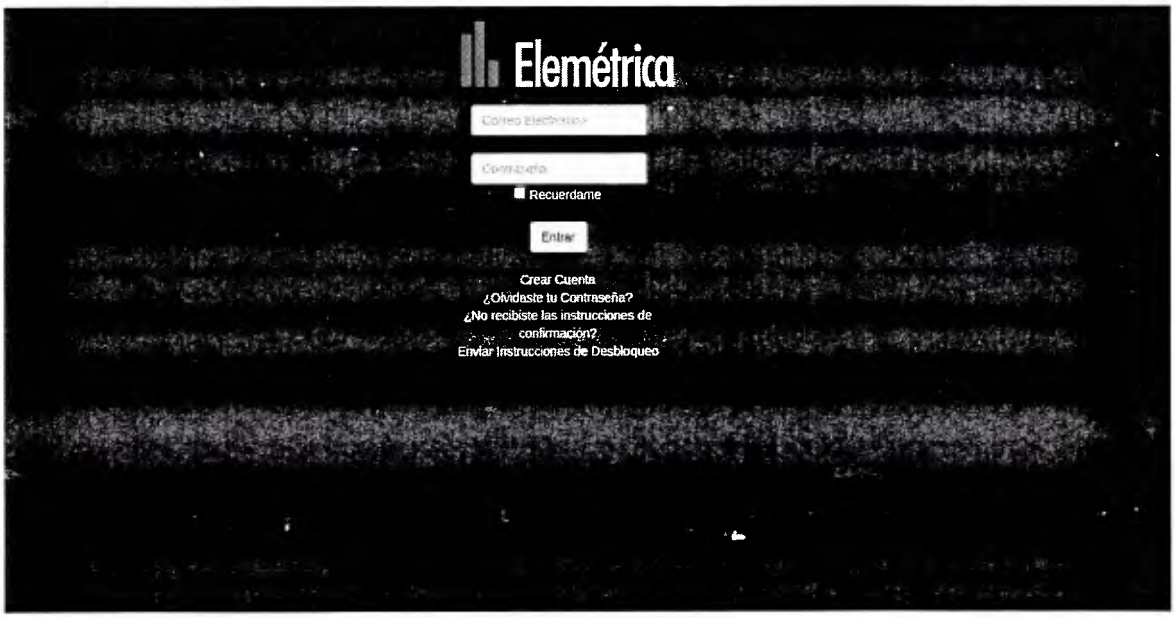

ILUSTRACIÓN 9 PANTALLA INICIO SESIÓN
FUENTE: ELABORACIÓN PROPIA

ILUSTRACIÓN 10 PANTALLA REGISTRO
FUENTE: ELABORACIÓN PROPIA

ILUSTRACIÓN 11 PANTALLA INDEX CLIENTES
FUENTE ELABORACIÓN PROPIA

ILUSTRACIÓN 12 FORMULARIO CREACIÓN CLIENTE
FUENTE: ELABORACIÓN PROPIA

ILUSTRACIÓN 13 PANTALLA VER CLIENTE
FUENTE: ELABORACIÓN PROPIA

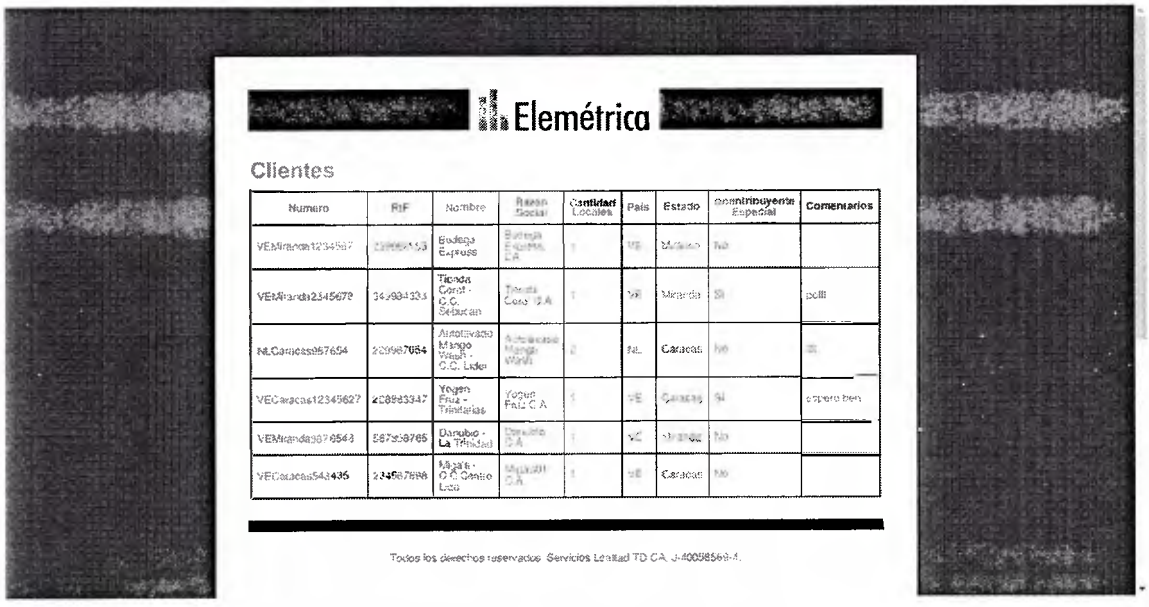

ILUSTRACIÓN 14 PDF LISTA DE CLIENTES
FUENTE: ELABORACIÓN PROPIA

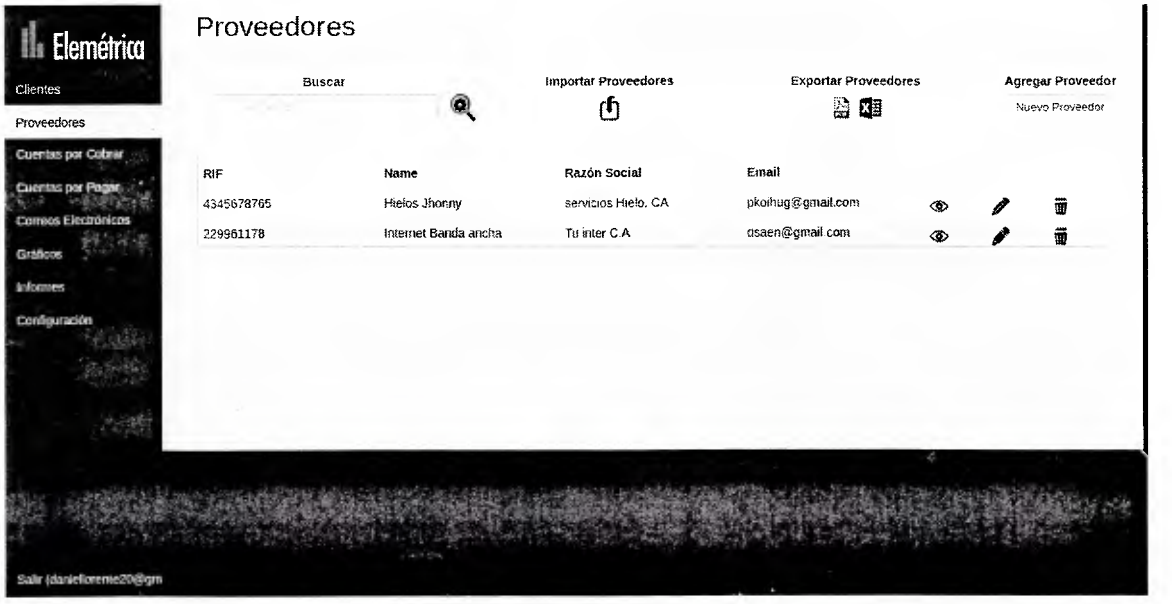

ILUSTRACIÓN 15 PANTALLA INDEX PROVEEDORES
FUENTE: ELABORACIÓN PROPIA

ILUSTRACIÓN 16 PANTALLA INDEX CUENTAS POR COBRAR
FUENTE: ELABORACIÓN PROPIA

ILUSTRACIÓN 17 PANTALLA INDEX CLIENTES
FUENTE: ELABORACIÓN PROPIA

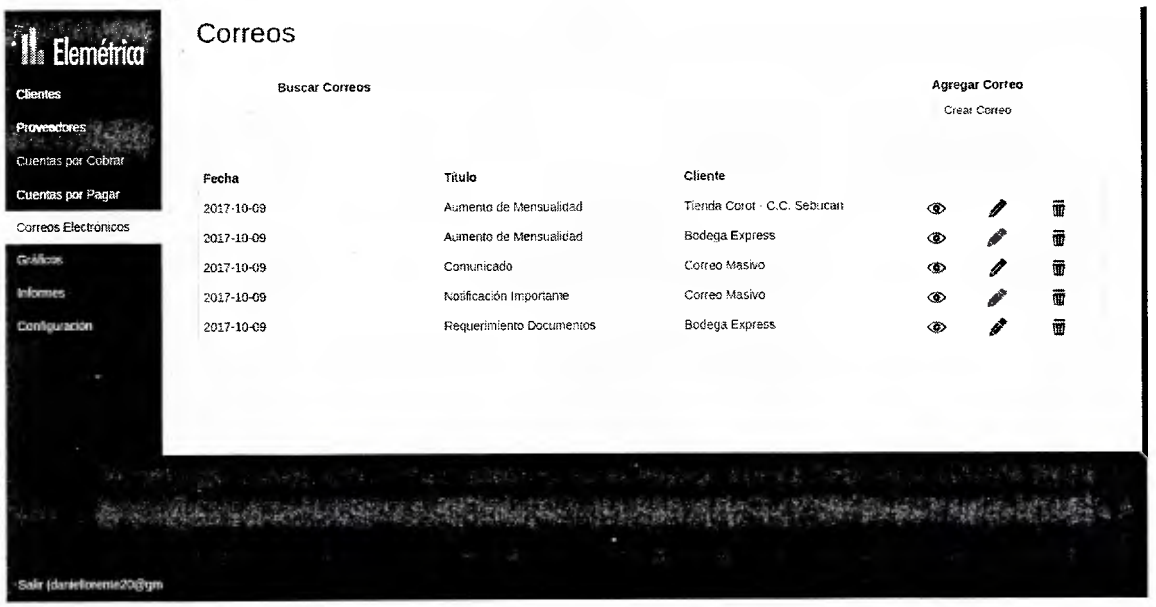

ILUSTRACIÓN 18 PANTALLA INDEX CORREOS ELECTRÓNICOS
FUENTE: ELABORACIÓN PROPIA

ILUSTRACIÓN 19 GRÁFICO MONTO FACTURADO VS MONTO PAGADO MES ACTUAL
FUENTE: ELABORACIÓN PROPIA

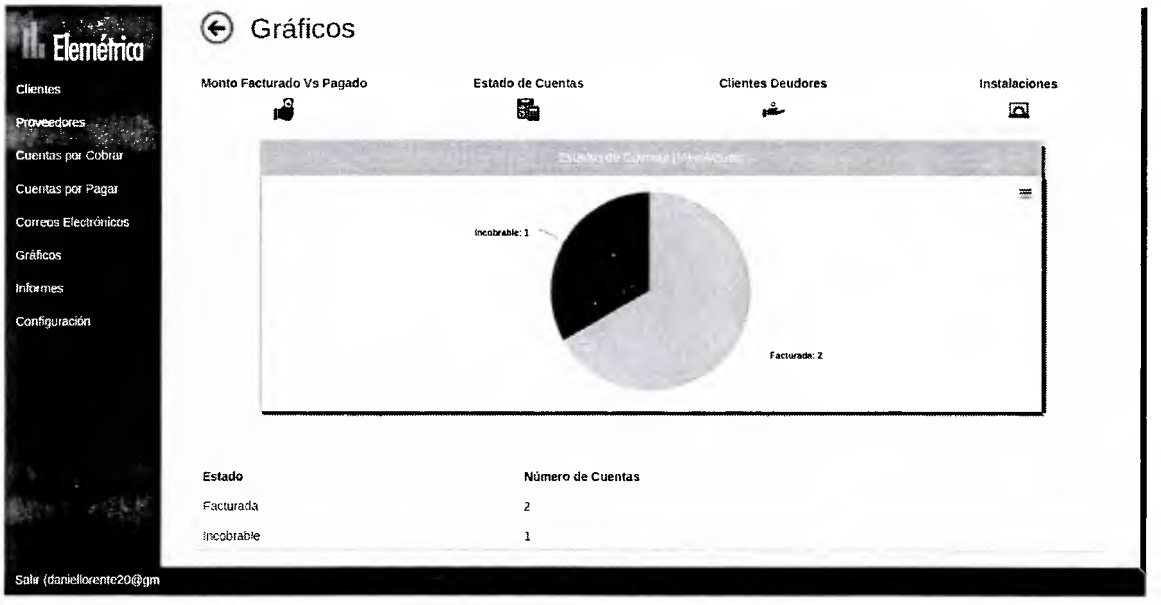

ILUSTRACIÓN 20 GRÁFICO ESTADO DE CUENTAS MES ACTUAL
FUENTE: ELABORACIÓN PROPIA

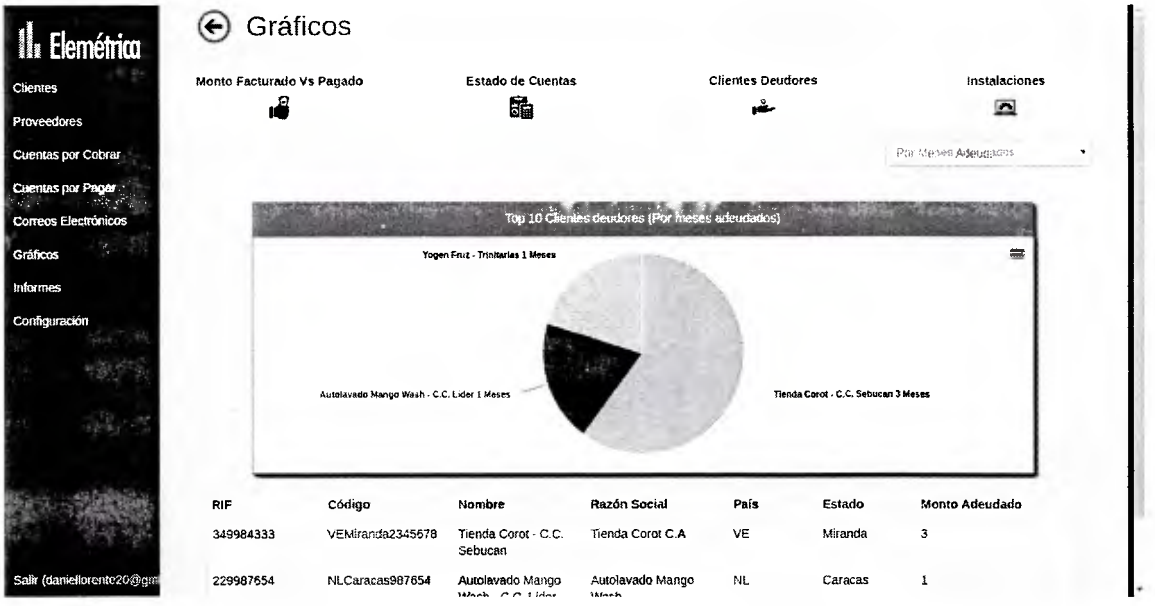

ILUSTRACIÓN 21 GRÁFICO TOP 10 CLIENTES DEUDORES POR MESES ADEUDADOS
FUENTE: ELABORACIÓN PROPIA

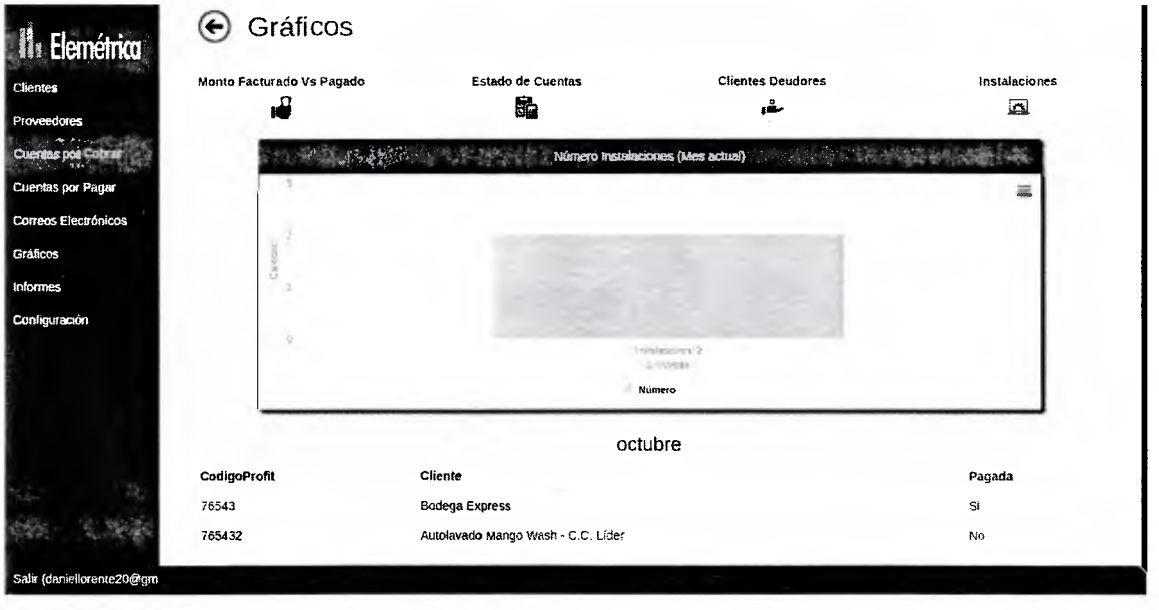

ILUSTRACIÓN 22 GRÁFICO INSTALACIONES MES ACTUAL
FUENTE: ELABORACIÓN PROPIA

ILUSTRACIÓN 23 PANTALLA REGISTRO DE PAGO CLIENTES
FUENTE: ELABORACIÓN PROPIA

ILUSTRACIÓN 24 CORREO ELECTRÓNICO A CLIENTE
FUENTE: ELABORACIÓN PROPIA

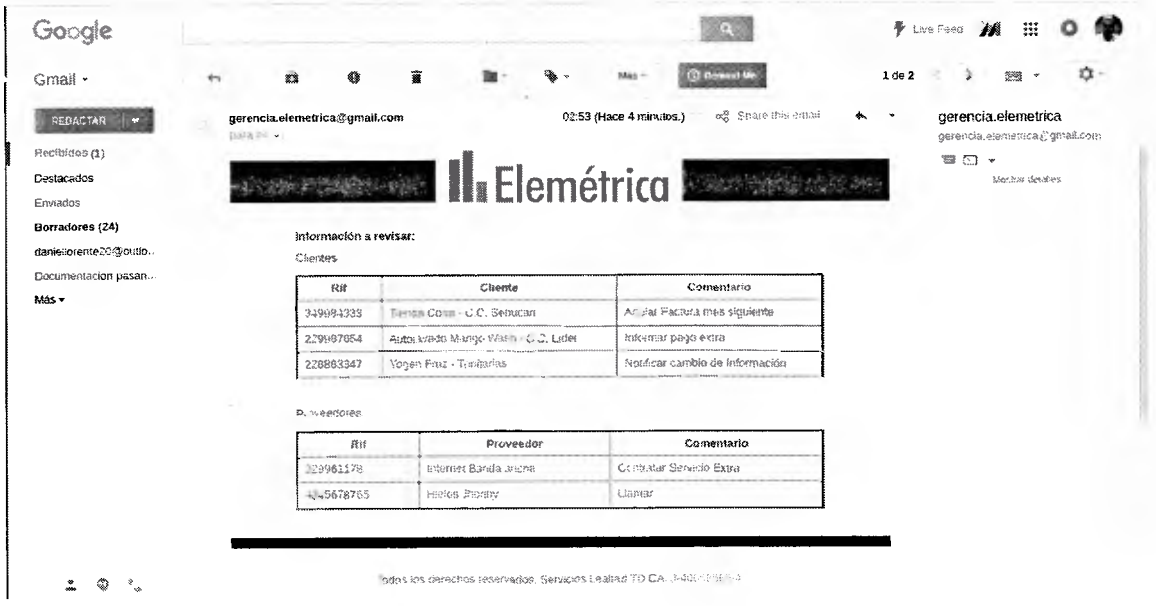

ILUSTRACIÓN 25 RECORDATORIO USUARIO ADMINISTRACIÓN
FUENTE: ELABORACIÓN PROPIA

ILUSTRACIÓN 26 CORREO NOTIFICACIÓN REGISTRO DE PAGO
FUENTE: ELABORACIÓN PROPIA

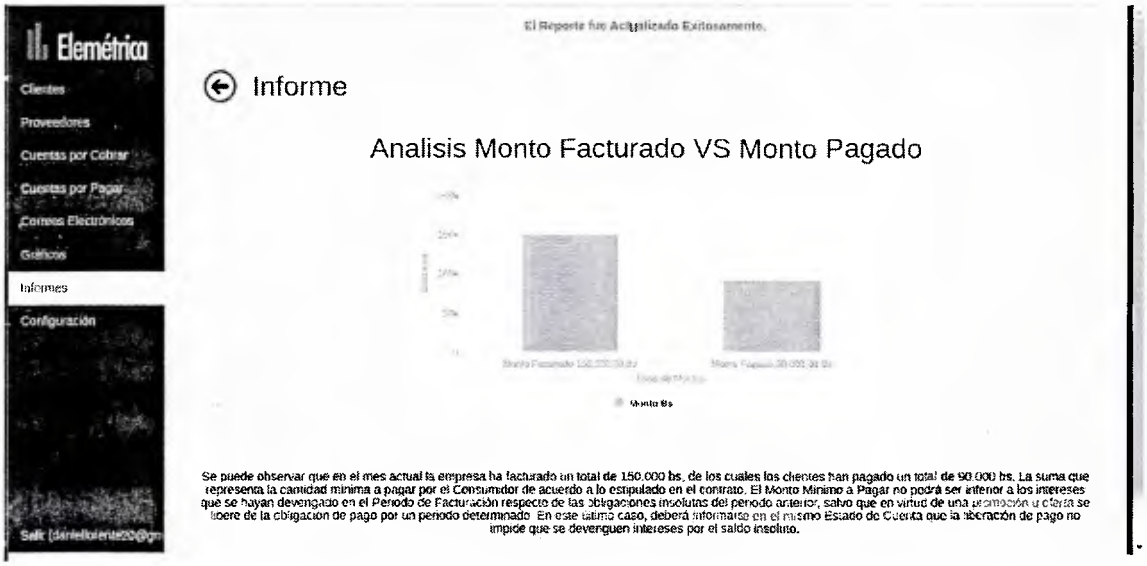

ILUSTRACIÓN 27 INFORME
FUENTE: ELABORACIÓN PROPIA