

Universidad Nacional Autónoma de Nicaragua, Managua
Facultad Regional Multidisciplinaria de Chontales
“Cornelio Silva Arguello”
UNAN - Managua / FAREM – Chontales



Seminario de Graduación para optar al Título de
Ingeniero en Sistemas de Información

Tema General

Desarrollo de sistemas web y/o aplicaciones móviles, utilizando arquitectura distribuida

Subtema

Desarrollo de un sistema web para la gestión del servicio de televisión por cable que ofrece la empresa Americable S.A, utilizando arquitectura distribuida de tipo cliente – servidor, en la ciudad de Juigalpa Chontales durante el segundo semestre del año 2016.

Autores:

- Br. Ariel Emmanuel Solano Ortega
- Br. Weslin Aldemaro Vargas Martínez
- Br. Evert Josué Ortiz Morales

Tutores:

- MSc. Miriam Patricia Téllez Marín
- MSc. Saira María Urbina Cienfuegos

Juigalpa – Chontales, 2017

¡A la libertad por la Universidad!

Dedicatoria

A Dios tan bueno y misericordioso que siempre está conmigo y muy en especial a: Mi madre, Eloísa Ortega Matus a quien por su esfuerzo y sacrificio debo mi formación profesional y a mis hermanos, Elvis Antonio Solano Ortega y Elibeth Eloísa Solano Ortega por su apoyo incondicional.

Ariel Emmanuel Solano Ortega

A Dios por haberme dado vida; a mis padres por ser pilares fundamentales e influir en toda mi educación a lo largo de mi vida y a cada uno de mis maestros por haberme brindado su conocimiento y paciencia en todo el transcurso de mi carrera.

Evert Josué Ortiz Morales

A Dios mi padre celestial a quien le debo la vida, mis logros y me ha brindado la sabiduría e inteligencia para seguir adelante, a mis padres por estar siempre apoyándome sin importar las circunstancias difíciles y ser la base de mi educación a lo largo de mi vida.

Weslin Aldemaro Vargas Martínez.

Agradecimiento

A Dios por darme vida, perseverancia en mis estudios y llegar hasta el final del cierre de mi seminario de graduación; a mi madre Eloísa Ortega Matuz por enseñarme a cultivar los valores de la vida, uno de ellos son nuestros estudios y el respeto a nuestros semejantes, a mi hermano Elvis Antonio Solano Ortega por ayudarme económicamente en todo el transcurso de mi carrera y motivarme para culminar; igual a las personas que me quieren y me brindan su amistad y me regalan un lugar en su corazón y a mis tutores Msc. Miriam Téllez y Msc. Saira María Urbina Cienfuegos por brindar sus conocimientos, ayuda, paciencia, comprensión y apoyo incondicional para realizar el seminario de graduación, de igual manera a todos los maestros que aportaron su conocimiento en el transcurso de mi carrera.

Ariel Emmanuel Solano Ortega

A Dios por haberme permitido culminar esta etapa de mi vida y haberme dado salud, perseverancia, fortaleza y valor para lograr esta meta; a mis padres por haberme apoyado en todo momento, por sus consejos, motivación y valores, que me han permitido ser una persona de bien, pero sobre todo por su amor y confianza; a mis amigos y personas que estuvieron en todo momento a mi lado brindándome su apoyo incondicional, sus palabras de aliento en los momentos más difíciles y a todos mis profesores por brindarme sus conocimientos y ser parte fundamental de este crecimiento como persona y profesional.

Evert Josué Ortiz Morales

Agradezco sinceramente a todas las personas que de otra forma contribuyeron a la culminación de este trabajo, muy en especial a: Msc. Miriam Téllez, Msc. Saira María Urbina Cienfuegos por su valiosa asesoría y orientación, además de su ayuda incondicional durante la realización de este trabajo, al cuerpo de docente a quienes debo los conocimientos adquiridos.

Weslin Aldemaro Vargas Martínez.



Universidad Nacional Autónoma De Nicaragua, Managua
Facultad Regional Multidisciplinaria de Chontales
“Cornelio Silva Arguello”

2017 “Año de la Universidad Emprendedora”

Departamento de Ciencia, Tecnología y Salud

Valoración Docente

En la medida que el Internet ha ganado ventaja como herramienta de comunicación versátil, la idea de los sistemas informáticos de escritorio evolucionó, dando paso a sitios web que se convierten en sistemas de información útiles, fiables y robustos, que pueden interactuar fácil con aplicaciones móviles a través de protocolos estandarizados para este fin.

Los sistemas web, para gestión de información en pequeñas y grandes empresas es la mejor opción a elegir permitiéndoles actualización, innovación y crecimiento planificado a estas entidades. El tema “Desarrollo de un sistema web para la **gestión del servicio de televisión por cable** que ofrece la empresa Americable S.A, utilizando arquitectura distribuida de tipo cliente – servidor, en la ciudad de Juigalpa chontales durante el segundo semestre del año 2016”, se considera pertinente, debido a que por su naturaleza será promovido por Internet, acercándose con facilidad a usuarios finales, al mismo tiempo ofrece control para los principales procesos que se realizan con respecto a la relación usuario-servicio.

Se considera que los estudiantes **Ariel Emmanuel Solano Ortega, Weslin Aldemaro Vargas Martínez, Evert Josué Ortiz Morales**, reúnen los requisitos y méritos para la evaluación del jurado examinador, en la defensa de trabajo de grado para optar al título de ***Ingeniero en Sistemas de Información.***

M.Sc. Miriam Patricia Téllez Marín

M.Sc. Saira María Urbina Cienfuegos

Resumen

El objetivo principal de este documento es el desarrollo de un sistema web para el control del servicio de televisión por cable, ofrecido por la empresa Americable S.A debido a que la empresa no cuenta con un software que permita automatizar los procesos y gestiones dentro de la empresa.

En la Ciudad de Juigalpa existen dos empresas que ofrecen el servicio de televisión por cable; Americable S.A y Claro, donde Americable S.A ha sido el pionero de esta tarea por más de 20 años de existencia pero la empresa aún carece de un sistema que realice el registro de sus clientes.

Se ve la necesidad de determinar los requerimientos funcionales y no funcionales para diseñar una interfaz de acuerdo a los procesos que se realizan, luego se codifican la estructura lógica mediante la arquitectura cliente-servidor, una vez terminada la interfaz y la programación se procede a la evaluación para verificar que cumple con los requisitos encontrados.

La metodología para el desarrollo del sistema es en cascada definida por Ian Sommerville en su libro Ingeniería de Software.

Al cumplir con los objetivos del proyecto permitirá la automatización del servicio de televisión por cable ofrecido por la Empresa Americable de modo que la gestión de los abonados se realiza en poco tiempo y distancia no será un obstáculo para que el usuario reporte una avería o verifique su estado de cuenta

Indice General

1.	Introducción	1
2.	Antecedentes	2
3.	Planteamiento del problema	3
3.1.	Descripción del problema	3
3.2.	Formulación del problema	4
3.3.	Delimitación del problema.....	4
4.	Objetivos	5
4.1.	Objetivo General.....	5
4.2.	Objetivos Específicos	5
5.	Marco Teórico	6
5.1.	Gestión de los Sistemas de Información en las empresas.....	6
5.1.1.	Información en la empresa	6
5.1.2.	Aspectos fundamentales de los sistemas de información en la empresa.....	6
5.1.3.	Aplicación en Empresas	8
5.1.4.	Contribución de los sistemas de información a la economía de las empresas	8
5.1.5.	Definición de los objetivos de un sitio web para las empresas	8
5.1.6.	La web como sistema de información	9
5.2.	Sistemas de información	10
5.2.1.	Evolución de los sistemas de información	10
5.2.2.	Importancia de los Sistemas de Información en la empresa.....	12
5.3.	Sistemas Web.....	13
5.3.1.	Evolución de las aplicaciones web	13
5.3.2.	Ventajas de aplicaciones web	14
5.3.3.	Lenguajes y tecnologías en programación web	15
5.3.4.	Tipos de aplicaciones web.....	15
5.4.	Sistemas Distribuidos	15
5.4.1.	Arquitectura de sistemas distribuidos.....	15
5.4.2.	Arquitectura cliente-servidor	16
5.4.3.	Arquitectura de las aplicaciones Web	17
5.4.4.	Modelo en capas	18

5.5.	Protocolo de servicios web	19
5.5.1.	Protocolo de Control de Transmisión (TCP/IP)	19
5.5.2.	Protocolo de Transferencia de Hipertexto (HTTP)	20
5.5.3.	Protocolo de acceso a objetos simple (SOAP)	20
5.6.	Etapas de Desarrollo del sistema	21
5.6.1.	Identificación de problemas, Oportunidades y Objetivos	21
5.6.2.	Determinación de los Requerimiento de Información.....	21
5.6.3.	Análisis de las necesidades del sistema.....	21
5.6.4.	Diseño del sistema recomendado	22
5.6.5.	Desarrollo y Documentación del software	22
5.6.6.	Prueba y Mantenimiento del sistema.....	22
5.6.7.	Implementación y Evaluación del sistema	23
5.7.	ENTITY FRAMEWORK	23
6.	Marco Metodológico	26
6.1.	Hipótesis:	26
6.2.	Sistematización de variables	26
6.2.1.	Operacionalización de variables	26
6.3.	Población y muestra.....	29
6.4.	Tipo de investigación.....	29
6.5.	Métodos y técnicas de recolección de información	29
6.6.	Técnicas y herramientas de procesamiento	29
6.7.	Ciclo de vida de desarrollo de software.....	30
7.	Marco Administrativo	31
7.1.	Cronograma del documento del proyecto	31
7.2.	Presupuesto de trabajo	32
7.2.1.	Presupuesto del desarrollo del software	32
8.	Marco de Resultados	36
8.1.	Cronograma de actividades.....	36
8.2.	Descripción de los resultados obtenidos según los objetivos	36
9.	Conclusiones	40
10.	Recomendaciones	41
11.	Glosario de Términos	42

12.	Referencias	45
13.	Anexos	47
13.1.	Instrumentos de recolección de información.....	49
13.2.	Modelado del Sistema.....	51
13.3.	Diseño de Interfaces.....	53
13.4.	Desarrollo, Documentación y codificación.	58
	Documento SRS	61
	Documento de Validación de Software.....	83
	Manual Técnico.....	102
	Manual de Usuario	13129

Índice de Tablas

Tabla 1.	Ecuaciones del Modelo Básico de COCOMO 81. [Boehm 1981]	32
Tabla 2.	Total de líneas de código por programa	33
Tabla 3.	Presupuesto del proyecto.....	35

Índice de Ilustraciones

Ilustración 1.	Sistema informático-Sistema de información [3].....	7
Ilustración 2.	Arquitectura Cliente-Servidor de tres capas [16]	17
Ilustración 3.	Presentación Distribuida [9].....	17
Ilustración 4.	Separación de Funciones [13]	18
Ilustración 5.	Entity Framework [20]	24
Ilustración 6.	Cronograma del Documento del proyecto.....	31
Ilustración 7.	Cronograma de Actividades	36



1. Introducción

Este documento de investigación contiene los elementos esenciales que atribuyen el desarrollo de un sistema web para el control del servicio de televisión por cable ofrecidos por la empresa Americable S.A. Estos incluyen la conceptualización de términos, herramientas y metodologías que han sido utilizadas para la comprensión de las características del sistema. Así como los conceptos claves que ayuden a entender términos que dan respuesta a la temática de investigación tales como: sistemas de información, sistemas web, sistemas distribuidos, gestión de la información, y las metodologías que se aplican al desarrollo de estos.

La función principal del sistema web, es la automatización de los procesos que involucra registros de clientes y empleados que realiza la empresa de manera manual, haciendo uso de actas y formatos impresos sobre los cuales se escriben contratos, pagos del servicio, seguimientos, entre otros. Los principales controles obtenidos en el sistema son: la lista de clientes activos e inactivos, las fechas que inician el servicio de televisión por cable, fechas de pago y mora. Así mismo, los técnicos de la empresa a través de sus cuentas en línea agilizan la atención a los clientes para dar solución oportuna a las averías reportadas. Los clientes tienen acceso al sistema al momento en que realizan el contrato del servicio de televisión por cable, otorgándoles un nombre de usuario y contraseña generada por el sistema, gracias a esto podrán ver sus estados de cuentas y reportar averías en caso de que sea necesario.

Con la implementación de este proyecto serán beneficiado los clientes al hacer uso del sistema web de manera fácil e intuitiva desde sus hogares o lugar de trabajo; y la empresa, Americable, porque al agilizar los procesos de gestión de servicios, incrementará sus ganancias y se ubicará en el mercado competitivo.

El documento se divide en distintas secciones que describen el desarrollo de la investigación y del sistema web. En la primera parte se encuentra el planteamiento del problema y descripción del mismo, en la segunda se explica la teoría que da respuesta al tema de investigación, la tercera aborda la metodología utilizada: herramientas que se requieren para la recolección de información y describe el tipo de estudio, en la cuarta sección está el marco administrativo compuesto por el cronograma de trabajo y presupuesto de desarrollo del proyecto, y como último componente se plantea la conclusión en base a los objetivos, recomendaciones de acuerdo al uso del sistema y se agregan, los anexos que contienen la documentación especificación de requerimientos de software y otros documentos que argumentan la investigación.



2. Antecedentes

El desarrollo de las nuevas tecnologías información y comunicación han provocado un impacto a nivel mundial, principalmente en medios como la televisión originados por la revolución de los servicios de TV vía satélite y cable.

América cuenta con la empresa DIRECTV que es un proveedor de servicios de difusión directa por satélite en vivo, transmite televisión digital, incluidos canales de audio y radio a los televisores fijos. Esta empresa vende sus servicios vía televisión e internet de manera global de modo que los procesos han sido automatizados haciendo uso de tecnologías en la web. [1]

Dentro de las empresas nicaragüenses, la empresa Claro vende un servicio llamado Claro TV que cuenta con paquetes de canales de televisión, introduciéndose como el número uno y líder del mercado a nivel nacional. El sistema web de claro cuenta con la venta de muchos servicios, controlando y gestionando la información de manera que se ha involucrado competitivamente a nivel internacional. [2]

En la ciudad de Juigalpa existen dos empresas que brindan el servicio de televisión por cable: Americable y Claro; donde Americable ha sido el pionero en esta tarea desde hace veinte años; pero sus gestiones continúan siendo de forma manual, sin embargo, la competencia (Claro) cuenta con servicios automatizados que permite la gestión correcta de los procesos, permitiendo una ventaja competitiva en el negocio.

La empresa Americable no ha utilizado ningún tipo de software o sistema de información que les permita la automatización de los servicios de registro de clientes y empleados, reportes, pagos de servicios, entre otros.



3. Planteamiento del problema

En esta sección se define y estructura formalmente la idea de investigación mediante los siguientes procedimientos:

3.1. Descripción del problema

El presente trabajo pretende dar respuesta a las necesidades del manejo de la información a la Empresa Americable la cual brinda el servicio de televisión por cable, con varios años de estar funcionando en el mercado local de la ciudad de Juigalpa y ha sido líder en este servicio, actualmente expandida a unos municipios principales del departamento.

La situación actual que presenta la Empresa es que el control de los servicios aún se lleva de forma manual, en libros de gran volumen, provocando problemas en la administración de la información tales como: pérdida de datos personales de clientes, acumulación de atrasos en los pagos mensuales del servicio, respuestas lentas a solicitudes nuevas y averías reportadas, entre otras.

Los contratos, las averías y los pagos son realizados manualmente lo que da lugar a la pérdida y redundancia de información.

Las averías son reportadas por los clientes por medio de llamadas a las oficinas de Americable, los contratos son realizados mediante la visita del cliente y los pagos son realizados por medio de un cobrador que se envía al lugar de residencia de los abonados que tienen el servicio contratado.

Los contratos normalmente son registrados en actas y formatos impresos, por lo que dificulta controlar con exactitud la lista de clientes y fechas de ingresos. El control de los pagos se realiza de igual manera, esto conlleva a la mayor problemática de la empresa debido a que la mayor parte del tiempo los clientes no pagan en tiempo y forma o simplemente no pagan. El problema reside en la pérdida de ganancias de la empresa reduciendo la competitividad en el sector en el que operan.



3.2. Formulación del problema

Debido al incremento de la población y el avance de las nuevas tecnologías en la ciudad de Juigalpa en la Empresa Americable surge la necesidad de automatizar los procesos que se realizan, acción que proveerá estabilidad, control y orden en la compañía.

¿Cómo desarrollar un sistema web que gestione de manera eficiente el servicio de televisión por cable ofrecido por la empresa Americable en Juigalpa Chontales en el segundo semestre del año 2016?

3.3. Delimitación del problema

Teniendo en cuenta la problemática observada en la institución, se hace indispensable el desarrollo de un sistema web que ayude a mejorar la gestión de la Empresa Americable. Y de esta manera ofrecer un servicio que facilite la administración de la empresa automatizando sus procesos.



4. Objetivos

4.1. Objetivo General

Desarrollar un sistema web para la gestión del servicio de televisión por cable que ofrece la empresa Americable S.A, utilizando arquitectura distribuida de tipo cliente – servidor, en la ciudad de Juigalpa Chontales durante el segundo semestre del año 2016.

4.2. Objetivos Específicos

1. Determinar los requerimientos funcionales y no funcionales para la automatización de la gestión de los servicios de abonados de la empresa Americable.
2. Diseñar procesos e interfaz de usuario de acuerdo a los requerimientos determinados para la gestión de los servicios ofrecidos por la empresa Americable.
3. Aplicar la metodología de programación en capas y tecnología web para el funcionamiento requerido del sistema a desarrollar.
4. Evaluar el cumplimiento de los requerimientos funcionales y no funcionales en el sistema web desarrollado a través de pruebas de validación.



5. Marco Teórico

5.1. Gestión de los Sistemas de Información en las empresas

Los Sistemas de Información (SI) y las Tecnologías de Información (TI) han cambiado la forma en que operan las empresas y organizaciones actuales. A través de su uso se logran importantes mejoras, se automatizan los procesos operativos, además suministran una plataforma de información necesaria para la toma de decisiones y, lo más importante, su implantación logra ventajas competitivas o reducir la ventaja de los rivales.

5.1.1. Información en la empresa

Toda persona, toda empresa, y en general toda organización, está continuamente captando una serie de datos, gran parte de los cuales no tienen significación alguna para ella, pero en cambio existen otros datos que le sirven para conocer mejor el entorno que le rodea y también para conocerse mejor. Estos datos, que constituyen la llamada información, le van a permitir tomar decisiones más acertadas.

La información constituye un factor esencial para la empresa en cuanto que la posesión o no de las informaciones oportunas va a ser un factor determinante de la calidad de las decisiones que se adopten y, en consecuencia, de la estrategia que pueda en un momento determinado diseñarse y posteriormente ponerse en práctica. Una información bien elaborada puede evitar en gran medida problemas derivados de la incertidumbre sobre el entorno, ya sea por la falta de claridad sobre algunos aspectos, o por una gran acumulación de datos donde se requiere tomar una decisión en un corto periodo de tiempo. [3]

En las empresas actualmente la información de los clientes es la principal prioridad a resguardar, es por ello que estos datos o información son esenciales y se ve la necesidad de evitar que se filtren a la competencia porque de esta forma podría perder la competitividad con las demás empresas del sector.

5.1.2. Aspectos fundamentales de los sistemas de información en la empresa

Todo sistema se puede dividir en subsistemas. Dado que la empresa se comporta como un sistema, es posible fragmentar sus partes en subsistemas. El sistema de información se relaciona con el resto de sistemas y con el entorno. Un sistema de información en la empresa debe servir para captar la información que esta necesite y ponerla, con las transformaciones necesarias, en poder de aquellos



miembros de la empresa que la requieran, bien sea para la toma de decisiones, bien sea para el control estratégico, o para la puesta en práctica de las decisiones adoptadas

La definición de sistema de información que dan Andreu, Ricart y Valor (1991). Según estos autores, el sistema de información: «Es el conjunto formal de procesos que operando sobre una colección de datos estructurada de acuerdo con las necesidades de una empresa, recopila, elabora y distribuye la información necesaria para la operación de dicha empresa y para las actividades de dirección y control correspondientes, apoyando, al menos en parte, los procesos de toma de decisiones necesarios para desempeñar las funciones de negocio de la empresa de acuerdo con su estrategia».

Un sistema de información es algo más que un sistema informático. El sistema de información es indisoluble del sistema organización-entorno, y en el proceso de adopción de decisiones no se puede pretender que toda la información necesaria sea predeterminada, formalizada e informatizada. La información circula por toda la organización como si fuera un fluido, por cauces formales e informales, y en sentido horizontal y vertical. El sistema de información constituye la estructura organizativa que debe administrar dichos flujos de información con la máxima eficacia y eficiencia para llevar a cabo las funciones de una empresa determinada de acuerdo con su planteamiento o estrategia de negocio. [3]



Ilustración 1. Sistema informático-Sistema de información [3]



5.1.3. Aplicación en Empresas

La información se ha colocado en un buen lugar como uno de los principales recursos que poseen las empresas actualmente. Los entes que se encargan de las tomas de decisiones han comenzado a comprender que la información no es sólo un subproducto de la conducción empresarial, sino que a la vez alimenta a los negocios y puede ser uno de los tantos factores críticos para la determinación del éxito o fracaso de éstos. Es necesario respaldar, procesar y sacar provecho a la información que es la base de todo negocio empresarial; la mejor manera de lograr esto es a través de los sistemas de información que ayudan a gestionar los datos de manera confiable, el cual están diseñados para dar solución a problemas agilizando procesos que se manejan de manera manual.

5.1.4. Contribución de los sistemas de información a la economía de las empresas

Tenemos presente la era de la información, donde el conocimiento es poder. Hoy en día las organizaciones están usando la información (y tecnologías de información) para ganar y sostener una estabilidad, y en el mundo de los negocios obtener una ventaja competitiva. Jamás encontraras una organización cuyo eslogan sea “Lo que no sabes no puede hacerte daño”, en esta era se sobreentiende que el no saber puede ser un factor clave en el trayecto y exitoso cumplimiento de los objetivos planteados por la empresa.

Un sistema de información es una combinación de 5 elementos clave: datos, hardware, software, personas, y telecomunicaciones que son creadas y usadas por personas, para recolectar, crear, y distribuir información útil típicamente en entornos organizacionales.

El soporte y ayuda que proveen las tecnologías de información han significado de muchas formas la ventaja competitiva adquirida en las organizaciones, pero no es solo las tecnologías en sí mismas si no el uso que la organización les da.

La contribución que realizan los sistemas de información a la economía de la empresa en muchos casos es difícil de determinar con precisión, porque la empresa puede estar midiendo los beneficios erróneamente. A pesar de que los SI tienen beneficios reales estos pueden ser no detectados, esto se debe a que no saben qué área están midiendo. [4]

5.1.5. Definición de los objetivos de un sitio web para las empresas

Los objetivos de un sitio web pueden ser muy diferentes y variados en función del tipo de negocio al que se dedique la empresa u organización a la que pertenece el sitio web. Definir estos objetivos



puede ser una tarea complicada y posiblemente no se sepa por dónde comenzar. Es importante definir unos objetivos adecuados y correctos para el tipo de negocio o sector al que va dirigido el sitio web, ya que nos van a permitir medir el funcionamiento de una compañía. Los objetivos de un sitio web tienen que ser concretos no generales, con unas metas claras, con una temporalidad definida, con una estrategia a seguir y acotados. Se recomienda comenzar definiendo un objetivo principal y posteriormente definir varios objetivos secundarios, escribiendo una frase que comience con un verbo.

Los objetivos deben estar en concordancia con las necesidades de la empresa e integrados en la estrategia de negocio. Por ello la forma más rápida y sencilla de definirlos es revisar cuales son los objetivos de la organización y en función de ellos se pueden obtener las claves que nos van a permitir desarrollarlos. Dependiendo de los objetivos de un sitio web, nos encontramos con diferentes categorías en función del público objetivo [4]

5.1.6. La web como sistema de información

Los sistemas de información han venido evolucionando a gran escala de modo que estos pueden ser manipulados directamente en internet. La web ha venido a ser el componente principal para que los clientes puedan comunicarse directamente con el sistema desde cualquier lugar y en cualquier momento, siendo un puente entre la interfaz y usuarios.

La evolución de internet como red de comunicación global y el surgimiento y el desarrollo del sistema web como servicio imprescindible para compartir información, creó un excelente espacio para la interacción del hombre con la información hipertextual, a la vez que sentó las bases para el desarrollo de una herramienta integradora de los servicios existentes en internet. Los sitios web como expresión de sistemas de información, deben poseer los siguientes componentes:

- Usuarios.
- Mecanismos de entrada y salida de la información.
- Almacenes de datos, información y conocimiento.
- Mecanismos de recuperación de información.

Se pudiese definir entonces como sistemas de información al conjunto de elementos relacionados y ordenados, según ciertas reglas que aportan al sistema objeto, es decir, a la organización a la que sirve y que marca sus directrices de funcionamiento la información necesaria para el cumplimiento



de sus fines; para ello, debe recoger, procesar y almacenar datos, procedentes tanto de la organización como de fuentes externas, con el propósito de facilitar su recuperación, elaboración y presentación. Actualmente los sistemas de información se encuentran al alcance de grandes masas de usuarios por medio de internet; así se crean las bases de un nuevo modelo, en el que los usuarios interactúan directamente con los sistemas de información para satisfacer sus necesidades. [5]

5.2. Sistemas de información

Los sistemas de información son un conjunto de herramientas relacionadas entre sí encargadas del análisis, procesamiento y distribución de la información necesaria para el funcionamiento de la Empresa apoyando al acto de la toma de decisiones y el control. Dichos componentes son: hardware, equipos computacionales, software, programa ejecutados por las computadoras, recursos humanos, fuentes de datos de información, las telecomunicaciones y las reglas de operación de apoyo a las empresas en la toma de decisiones para llevar un mejor control de la misma. [6]

5.2.1. Evolución de los sistemas de información

En la década de los 70, Richard Nolan un conocido autor y profesor de la escuela de Harvard, desarrollo una teoría que influyo, sobre el proceso de planeación de los recursos y las actividades de informática según Nolan, las funciones de la informática en las organizaciones evolucionan a través de ciertas etapas de crecimiento, las cuales se 1 14^a44 4101512continuación:

Etapas de inicio. Algunas de las características más importantes de esta etapa son:

- Comienza con la adquisición de la primera computadora y normalmente se justifica por el ahorro de mano de obra y el exceso de papeles.
- Las aplicaciones típicas que se implementan son los sistemas transaccionales, tales como nominas o contabilidad.
- El tipo de administración empleada es escaso y la función de los sistemas suele ser manejada por un administrador
- El personal que labora en este pequeño departamento consta, a lo sumo de un operador y un programador.



Etapas de contagio o expansión: los aspectos sobresalientes que permiten diagnosticar que una empresa se encuentre en esta etapa son:

- Se inicia con la implantación exitosa del primer sistema de información en la organización
- Las aplicaciones que con frecuencia se implantan en esta etapa son el resto de los sistemas transaccionales no desarrollados en la etapa de inicio tales como, facturación, inventario, control de pedidos de clientes, proveedores cheques etc.
- Un aspecto sobresaliente es la proliferación de aplicaciones en todas las organizaciones que debido a la falta de estándares e infraestructura adecuada se realiza de manera desordenada y sin control.

Etapas de control o formalización: para identificar a una empresa que transita por esta etapa es necesario considerar los siguientes elementos

- Evolución de la información dentro de la empresa
- Las aplicaciones están orientadas a facilitar el control de las operaciones
- El departamento de sistemas de la empresa suele ubicarse en una posición gerencial
- En esta etapa se inicia el desarrollo y la implementación de estándares de trabajo dentro del departamento

Etapas de integración. Las características de esta etapa son las siguientes:

- La integración de datos y de los sistemas surge como resultado directo de la centralización del departamento de sistemas.
- Las nuevas tecnologías relacionadas con bases de datos
- El costo del software disminuye por el cual estuvo al alcance del usuario

Etapas de administración de datos. Entre las características que se destacan en esta etapa son:

- La información es un recurso muy útil y muy valioso que debe estar accesible para todos los usuarios
- Administrar los datos de la manera adecuada
- Responsabilidad de la información. [7]



Etapa de madurez. Entre los aspectos sobresalientes que indican que una empresa se encuentra en esta etapa, se incluye lo siguiente:

- La informática de la organización se encuentra definida
- Se introducen las aplicaciones desarrolladas en la tecnología base de datos
- En muchos casos los servicios de cómputo.
- Suele existir una planeación rigurosa de los recursos de computo [7]

5.2.2. Importancia de los Sistemas de Información en la empresa

Debido a los enormes cambios sufridos por el mercado en los últimos años con la incorporación de tecnologías informáticas que facilitarán la administración de los datos, con el fin de ofrecer mejoras en la toma de decisiones gerenciales, en la actualidad todas las empresas, Pymes, requieren de la implementación de un sistema de información que colabore con los procesos de gestiones empresariales.

Con el fin de mejorar la productividad y el rendimiento de una organización competitiva, es fundamental evaluar las técnicas actuales y la tecnología disponible para desarrollar sistemas que brinden eficiencia y eficacia de la gestión de la información relevante.

La implementación de sistemas de información en una compañía, brindan la posibilidad de obtener grandes ventajas, incrementar la capacidad de organización de la empresa, y tornar de esta manera los procesos a una verdadera competitividad.

Para ello, es necesario un sistema eficaz que ofrezca múltiples posibilidades, permitiendo acceder a los datos relevantes de manera frecuente y oportuna.

Por eso, quienes deseen convertir su emprendimiento en una verdadera empresa competitiva insertada en el mercado actual, deberán analizar detalladamente la implementación necesaria de sistemas de información precisos.

Este concepto ha invadido en los despachos de los encargados de la gerencia de compañías en los últimos años, ya que hace un par de décadas atrás la información no era considerada un bien tan útil dentro de los procesos productivos y las tomas de decisiones.

Sin embargo, a raíz de los cambios en la economía mundial y la globalización, los datos relativos a todo el proceso productivo de una compañía se han vuelto uno de los elementos fundamentales para lograr el éxito comercial. [8]



De esta manera, con los años se han ido incorporando los avances de la tecnología informática para brindar las herramientas necesarias en la creación de sistemas de información confiable y eficaz.

No obstante, en la actualidad aún existen empresas que observan con recelo la posible implementación de sistemas de información en sus procesos, debido a que implican un enorme cambio en las estructuras organizativas e institucionales de las compañías.

Es por ello, que muchas veces los sistemas de información pueden llegar a fallar, no por errores tecnológicos originados en el aspecto informático, sino por visiones culturales opuestas a la incorporación de este tipo de herramientas.

Cabe destacar, que los sistemas de información ofrecen una gran cantidad de ventajas subyacentes, más allá del rédito que significa para la toma de decisiones y los procesos productivos.

Un sistema de información adecuado además ofrece una importante y notable satisfacción en los usuarios que lo operan, debido a su facilidad de uso y su acceso constante, que puede resultar en que los empleados logren alcanzar los objetivos planteados por la compañía.

Por todo ello, es importante destacar que la implementación de un sistema de información implica un cambio organizativo, ya que no sólo afecta a la administración de la empresa, sino también a sus empleados y habilidades, con el fin de crear una plataforma acorde a las responsabilidades que se deben tener frente a este tipo de sistema.

Por otra parte, es de vital importancia utilizar tecnologías de información y comunicación adecuadas para el procesamiento y transmisión de los datos que se gestionarán en el sistema de información. [8]

5.3. Sistemas Web

Los sistemas web nos permiten procesar y mostrar información dinámicamente para el usuario, se pueden utilizar a través de cualquier navegador web, estos sistemas no necesitan ser instalados ya que están alojados en un servidor donde se aloja el sistema y así se tiene el acceso directo a ellos.

5.3.1. Evolución de las aplicaciones web

La capacidad de enlazar un texto con otro para crear un hipertexto es la característica más destacable de las páginas web. Aunque su éxito es relativamente reciente, sus orígenes se remontan al sistema Memex ideado por Vannevar Bush ("*As we may think*", Atlantic Monthly, julio de 1945). El término hipertexto lo acuñó Ted Nelson en 1965 para hacer referencia a una colección de documentos



(nodos) con referencias cruzadas (enlaces), la cual podría explorarse con la ayuda de un programa interactivo (navegador) que nos permitiese movernos fácilmente de un documento a otro.

De hecho, la versión que conocemos actualmente del hipertexto proviene del interés de los científicos en compartir sus documentos y hacer referencias a otros documentos. Este interés propició la creación de la "tela de araña mundial" (*World-Wide Web*, WWW) en el Centro Europeo para la Investigación Nuclear (CERN). Tim Berners-Lee, uno de los científicos que trabajaba allí, ideó el formato HTML para representar documentos con enlaces a otros documentos. Dicho formato fue posteriormente establecido como estándar por el W3C (*World-Wide Web Consortium*, <http://www.w3c.org/>). [9].

5.3.2. Ventajas de aplicaciones web

Usar aplicaciones web ahorra dinero. Empleará mejor su tiempo por no tener que ocuparse de aprender a manejar nuevos programas, ni mantenerlos o hacer copias de seguridad de sus datos y podrá trabajar desde cualquier sitio. Será más eficiente; podrá ganar más y gastará menos. [10]

1. Ahorran costes de hardware y software
2. Fáciles de usar
3. Facilitan el trabajo colaborativo y a distancia
4. Escalables y de rápida actualización
5. Provocan menos errores y problemas
6. Los datos son más seguros

Según Kendall y Kendall “Hay muchos beneficios derivados de la implementación de una aplicación en la Web”

1. Una creciente difusión de la disponibilidad de un servicio, producto, industria, persona o grupo.
2. La posibilidad de que los usuarios accedan las 24 horas.
3. La estandarización del diseño de la interfaz.
4. La creación de un sistema que se puede extender a nivel mundial y llegar a gente en lugares remotos sin preocuparse por la zona horaria en que se encuentren [11]



5.3.3. Lenguajes y tecnologías en programación web

Actualmente existen diferentes lenguajes de programación para desarrollar en la web, estos han ido surgiendo debido a las tendencias y necesidades de las plataformas. Estas forman parte de las tecnologías que se utilizan.

5.3.4. Tipos de aplicaciones web

Existen muchos tipos de aplicaciones web estas no son realizadas precisamente para dispositivos móviles, sino que son realizadas para que funcionen y sean ejecutadas en cualquier tipo de navegador ya sea desde el móvil o equipos de escritorio.

Aplicación web estática: son básicamente informativas y están enfocadas principalmente a mostrar una información permanente, donde el navegante se limita a obtener dicha información sin poder interactuar con la página visitada.

En las páginas web estáticas no se utilizan bases de datos ni se requiere programación, Para desarrollarlas es suficiente utilizar código HTML y construirlas principalmente con hipervínculos o enlaces (links) entre las páginas que conforman el sitio.

Aplicación web dinámica: Las páginas web dinámicas son aquellas en las que la información presentada se genera a partir de una petición del usuario de la página. [12]

5.4. Sistemas Distribuidos

Los sistemas distribuidos se refieren a cualquier sistema en el que múltiples agentes autónomos, cada uno con capacidades de cómputo individual se comunican entre sí y afectan mutuamente su comportamiento. Los sistemas distribuidos hacen referencia a cualquier evento en el cual se maneja un sistema en una red de computadoras y trata de describir las tendencias hacia la funcionalidad distribuida. Los sistemas distribuidos se refieren a los servicios que provee un sistema de computación distribuido. [13]

5.4.1. Arquitectura de sistemas distribuidos

Dentro de la arquitecturas de sistemas distribuidos existen sistemas grandes de tiempo real en los que el software esta formados por varios procesos que pueden ejecutarse sobre procesadores diferentes así como un sistema multiprocesador; arquitecturas cliente-servidor, una aplicación se modela como un conjunto de servicios proporcionados por los servidores y un conjunto de clientes



que usan estos servicios; arquitectura de objetos distribuidos, los componentes fundamentales del sistemas son objetos que proporcionan una interfaz a un conjunto de servicios que ellos suministran; y computación distribuida inter organizacional el cual ha sido implementada a nivel organizacional en donde una organización tiene varios servidores y reparte su carga computacional entre ellos.

Sin embargo, será conveniente implementar la arquitectura cliente servidor debido a que se adapta a las circunstancias operativas de las empresas PYMES y MIPYMES. [14]

5.4.2. Arquitectura cliente-servidor

Un Sistema Cliente/Servidor en tres capas representa un Sistema Distribuido en el que se han separado los distintos servicios que componen el sistema. La división que normalmente se sigue para estos sistemas es la definición de tres capas lógicas (que posteriormente se convertirán en diferentes capas físicas) de la siguiente manera: En la capa más inferior se encuentra la capa de datos, en esta capa se ubican las diferentes bases de datos de las que la aplicación obtendrá y añadirá datos, en ésta capa también se encuentran los procedimientos almacenados que ayudan a simplificar los accesos, modificaciones e inserciones sobre los datos.

La siguiente capa contiene las reglas o la lógica de negocios, en esta capa se definen los componentes (entendiéndose por componente un conjunto de clases que hacen algo) que contienen la definición de las operaciones que son necesarias para que el sistema haga su trabajo, además en dichos componentes residen las operaciones que manejan los datos de la capa inferior. En éstos componentes están las reglas que dicen cómo utilizar los datos y mantienen la integridad de los mismos. Por otro lado, la capa de negocios oculta una posible distribución física de los datos.

Por último, está la capa de presentación, ésta capa se encarga únicamente de presentar los datos a los usuarios y de establecer la interface para que exista una comunicación entre los mismos usuarios y el sistema. Esta capa carece de procesamiento y se limita únicamente a mostrar los datos a los usuarios y comprobar que las peticiones de los mismos son, por lo menos, semánticamente, correctas y de esta manera evitar que se hagan peticiones a la capa de negocio que a prioridad son inviábiles debido a que incumplen algún requisito previo. [15]

En éste sentido, un sistema cliente/servidor en tres capas establece, al menos, tres capas donde los usuarios no tienen constancia sobre cómo se almacenan ni donde residen los datos, estos sólo se comunican con la capa de presentación, ésta se ocupa de procesar las peticiones de los usuarios y



transmitirlas a los componentes de la capa de negocios, en esta capa se procesará la petición modificando, pidiendo o consultando los datos necesarios, que son provistos por la capa de datos que además de proveer dichos datos a la capa superior se encarga de almacenarlos y mantenerlos correctamente. [15]

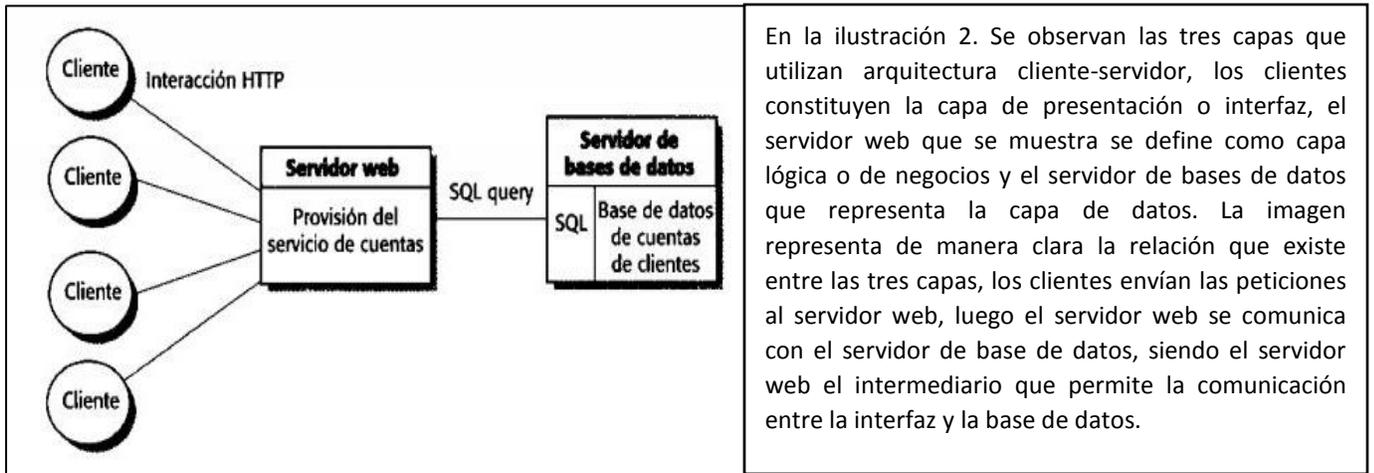


Ilustración 2. Arquitectura Cliente-Servidor de tres capas [16]

5.4.3. Arquitectura de las aplicaciones Web

Aplicaciones con arquitectura en las que los usuarios pueden utilizarlas accediendo a un servidor web a través de internet mediante un navegador. Es una aplicación de software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador.

5.4.3.1. Presentación distribuida

La presentación distribuida, en su forma más simple, es una interfaz gráfica de usuario a la que se le pueden acoplar controles de validación de datos, para evitar la validación de los mismos en el servidor. [16]

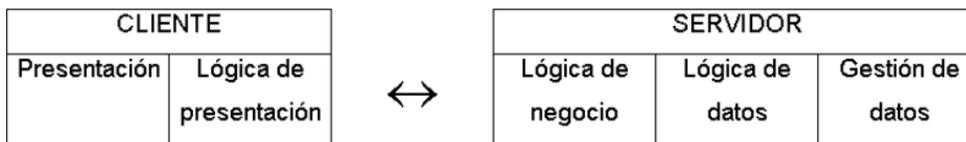


Ilustración 3. Presentación Distribuida [9]



5.4.4. Modelo en capas

Según Ian Sommerville el uso de una arquitectura de tres capas permite optimizar la transferencia de información entre el servidor web y el de la base de datos. [14]

5.4.4.1. Capa de Presentación

Lógica de presentación. Se encarga de la entrada y salida de la aplicación con el usuario. Sus principales tareas son: obtener información del usuario, enviar la información del usuario a la lógica de negocio para su procesamiento, recibir los resultados del procesamiento de la lógica de negocio y presentar estos resultados al usuario. [16]

5.4.4.2. Capa de Negocio

Lógica de negocio (o aplicación). Se encarga de gestionar los datos a nivel de procesamiento. Actúa de puente entre el usuario y los datos. Sus principales tareas son: recibir la entrada del nivel de presentación, interactuar con la lógica de datos para ejecutar las reglas de negocio (business rules) que tiene que cumplir la aplicación (facturación, cálculo de nóminas, control de inventario, etc.) y enviar el resultado del procesamiento al nivel de presentación. [12]

5.4.4.3. Capa de Datos

Lógica de datos. Se encarga de gestionar los datos a nivel de almacenamiento. Sus principales tareas son: almacenar los datos, recuperar los datos, mantener los datos y asegurar la integridad de los datos. [12]

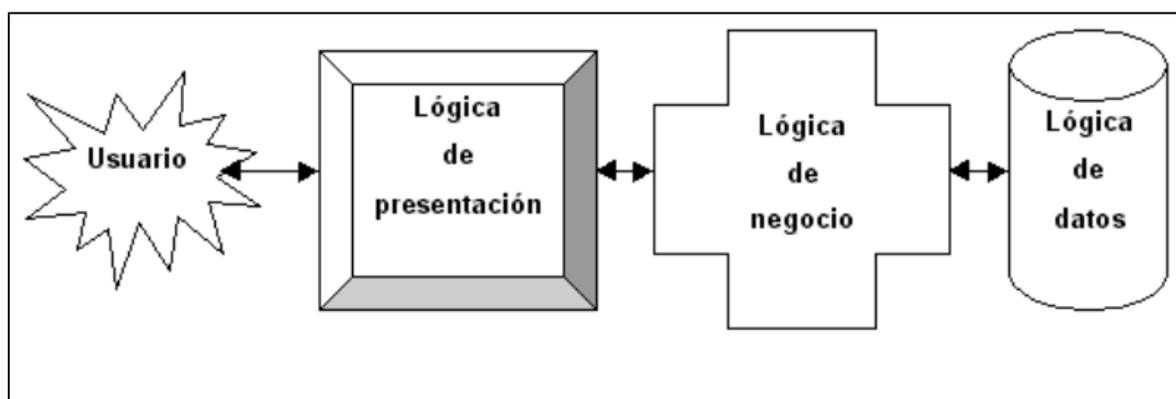


Ilustración 4. Separación de Funciones [13]



5.5. Protocolo de servicios web

Es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como internet.

5.5.1. Protocolo de Control de Transmisión (TCP/IP)

Son las siglas de Protocolo de Control de Transmisión/Protocolo de Internet un sistema de protocolos que hacen posibles servicios Telnet, FTP, E-mail, y otros entre ordenadores que no pertenecen a la misma red.

TCP/IP control de transmisión protocolo de internet representa todas las reglas de comunicación para Internet y se basa en la noción de dirección IP, es decir, en la idea de brindar una dirección IP a cada equipo de la red para poder en rutar paquetes de datos.

Servicios TCP/IP estándar

Existen distintos protocolos que permiten la comunicación mediante una red haciendo posible la transferencia de archivos, entre ellos podemos mencionar los siguientes:

- **FTP y FTP anónimo**

El Protocolo de transferencia de archivos (FTP por sus siglas en ingles File Transfer Protocol) transfiere archivos a una red remota y desde ella. El protocolo incluye el comando ftp y el daemon in.ftpd (proceso de servidor de protocolo de transferencia de archivos de Internet). FTP permite a un usuario especificar el nombre del host remoto y las opciones de comandos de transferencia de archivos en la línea de comandos del host local. [17]

- **TFTP**

El protocolo de transferencia de archivos trivial (TFTP por sus siglas en ingles Trivial File Transfer Protocol) ofrece funciones similares a ftp, pero no establece la conexión interactiva de ftp. Como consecuencia, los usuarios no pueden ver el contenido de un directorio ni cambiar directorios. Los usuarios deben conocer el nombre completo del archivo que se va a copiar. [17]



5.5.2. Protocolo de Transferencia de Hipertexto (HTTP)

Es el principal protocolo tecnológico de la red que permite enlazar y navegar por Internet. HTTPS es “Hyper Text Transfer Protocol” con una ‘S’ añadida al final, que hace referencia a “*Secure Sockets Layer*” (SSL) otro importante protocolo desarrollado para realizar transferencias de forma segura en Internet usando nuestro navegador. Básicamente, lo que ocurre es que la página Web codifica la sesión con certificado digital. De este modo, el usuario tiene ciertas garantías de que la información que envíe desde dicha página no podrá ser interceptada y utilizada por terceros. [18]

HTTP se basa en sencillas operaciones de solicitud/respuesta. Un cliente establece una conexión con un servidor y envía un mensaje con los datos de la solicitud. El servidor responde con un mensaje similar, que contiene el estado de la operación y su posible resultado. Todas las operaciones pueden adjuntar un objeto o recurso sobre el que actúan; cada objeto web (documento HTML, fichero multimedia) es conocido por su Localizador Uniforme de Recursos LUR (más conocido por la sigla URL, del inglés Uniform Resource Locator).

5.5.3. Protocolo de acceso a objetos simple (SOAP)

Microsoft define SOAP como un protocolo que define el formato XML (siglas en inglés de eXtensible Markup Language, traducido como "Lenguaje de Marcado Extensible" o "Lenguaje de Marcas Extensible") para los mensajes de intercambio en el uso de un Web Service. Para aquellos programadores que solían utilizar llamadas del tipo RPC (por sus siglas en inglés Remote Procedure Calls), SOAP también las soporta. Adicionalmente, es posible mediante SOAP definir un mensaje HTTP y este punto es de especial interés puesto que el protocolo imprescindible para Internet es HTTP. [19]

Básicamente SOAP es un paradigma de mensajería de una dirección sin estado, que puede ser utilizado para formar protocolos más complejos y completos según las necesidades de las aplicaciones que lo implementan. Puede formar y construir la capa base de una "pila de protocolos de web service", ofreciendo un framework de mensajería básica en el cual los web services se pueden construir.



En conclusión, el SOAP es el principal elemento de la infraestructura de servicios web y un factor clave en su funcionamiento, independiente de plataformas, sistemas operativos, modelos de objetos y lenguajes de programación, la interoperabilidad entre los objetos y componentes distribuidos. [19]

5.6. Etapas de Desarrollo del sistema

Estas son las etapas fundamentales para el desarrollo del software en ellas se detalla el modelo del software que se desea construir, esto con el fin de cumplir las funciones que realiza la empresa con la determinación de los requerimientos del sistema.

5.6.1. Identificación de problemas, Oportunidades y Objetivos

Según Ian Sommerville la primera fase requiere que el analista observe objetivamente lo que sucede en un negocio; el analista determina con precisión cuales son los problemas. El aprovechamiento de las oportunidades podría permitir a la empresa obtener una ventaja competitiva o establecer un estándar para la industria. La identificación de objetivos también es una parte importante de la primera fase. En primer lugar, el analista debe averiguar lo que la empresa trata de conseguir [14].

5.6.2. Determinación de los Requerimiento de Información

En la fase de determinación del requerimiento de información, el analista se esfuerza por comprender la información que necesitan los usuarios para llevar a cabo sus actividades. Esta fase es útil para que el analista confirme la idea que tiene de la organización y sus objetivos. Los implicados en esta fase son el analista y los usuarios, por lo general trabajadores y gerentes del área de operaciones.

5.6.3. Análisis de las necesidades del sistema

El analista tiene que ver con el análisis de las necesidades del sistema. De nueva cuenta, herramientas, y técnicas especiales auxilian al analista en la determinación de los requerimientos. Una de estas herramientas es el uso de diagramas de flujo de datos para graficar las entradas, los procesos y las salidas de las funciones del negocio en una forma gráfica estructurada. A partir de los diagramas de flujo de datos se desarrolla un diccionario de datos que enlista todos los datos utilizados en el sistema, así como sus respectivas especificaciones. [14]



Durante esta fase el analista de sistemas analiza también las decisiones estructuradas que se hayan tomado. Las decisiones estructuradas son aquellas en las cuales se pueden determinar las condiciones, las alternativas de condición, las acciones, las alternativas de condición, las acciones y las reglas de acción.

5.6.4. Diseño del sistema recomendado

El analista utiliza la información recopilada en las primeras fases para realizar el diseño lógico del sistema de información. El analista diseña procedimientos precisos para la captura de datos que aseguran que los datos que ingresen al sistema de información sean correctos. Además, el analista facilita la entrada eficiente de datos al sistema de información mediante técnicas adecuadas de diseño de formularios y pantallas.

La fase del diseño también incluye el diseño de archivos o bases de datos que almacenaran gran parte de los datos indispensables para los encargados de tomar las decisiones en la organización. [14]

5.6.5. Desarrollo y Documentación del software

En la quinta fase del ciclo de vida del desarrollo de sistemas, el analista trabaja de manera conjunto con los programadores para desarrollar cualquier software original necesario. Entre las técnicas estructuradas para diseñar y documentar software se encuentran los diagramas de estructuras.

Durante esta fase el analista también trabaja con los usuarios para desarrollar documentación efectiva para el software, como manuales de procedimientos, ayuda en línea y sitios Web que incluyan respuestas a preguntas frecuentes.

Los programadores desempeñan un rol clave en esta fase porque diseña, codifican y eliminan errores sintácticos de los programadores de cómputo. [14]

5.6.6. Prueba y Mantenimiento del sistema

Antes de poner el sistema en funcionamiento es necesario probarlo. Es mucho menos costoso encontrar los problemas antes que el sistema se entregue a los usuarios. Una parte de las pruebas las realizan los programadores solos, y otra la llevan a cabo de manera conjunta con los analistas de



sistemas. Primero se realiza una serie de pruebas con datos de muestra para determinar con precisión cuales son los problemas y posteriormente se realiza otra con datos reales del sistema actual.

El mantenimiento del sistema de información y su documentación empiezan en esta fase y se llevan a cabo de manera rutinaria durante toda su vida útil, Gran parte del trabajo habitual del programador consiste en el mantenimiento, y las empresas invierten enormes sumas de dinero en esta actividad. Parte del mantenimiento, como las actualizaciones de programas, se pueden realizar de manera automática a través de un sitio Web.

5.6.7. Implementación y Evaluación del sistema

Esta es la última fase del desarrollo de sistemas, y aquí el analista participa en la implementación del sistema de información. En esta fase se capacita a los usuarios en el manejo del sistema. Parte de la capacitación la imparten los fabricantes, pero la supervisión de esta es responsabilidad del analista de sistemas. Además, el analista tiene que planear una conversión gradual del sistema anterior al actual. Este proceso incluye la conversión de archivos de formatos anteriores a los nuevos, o la construcción de una base de datos, la instalación de equipo y la puesta en producción del nuevo sistema. [14]

Se menciona la evaluación como la fase final del ciclo de vida del desarrollo del sistema principalmente en aras del debate. En realidad, la evaluación se lleva a cabo durante cada una de las fases. [14]

5.7. ENTITY FRAMEWORK

Es un asignador objeto-relacional que permite a los desarrolladores de .NET trabajar con datos relacionales usando objetos específicos del dominio. Elimina la necesidad de la mayor parte del código de acceso a datos que los desarrolladores suelen tener que escribir.

Incluye el proveedor de datos de EntityClient. Este proveedor administra las conexiones, traduce las consultas de entidad en consultas específicas del origen de datos y devuelve un lector de datos que Entity Framework usa para materializar los datos de la entidad en los objetos. Cuando no se requiere la materialización de los objetos, el proveedor de EntityClient también se puede utilizar como un proveedor de datos ADO.NET estándar habilitando las aplicaciones para ejecutar consultas Entity El diagrama siguiente muestra la arquitectura de Entity Framework para el acceso a datos:



Diagrama de arquitectura de Entity Framework

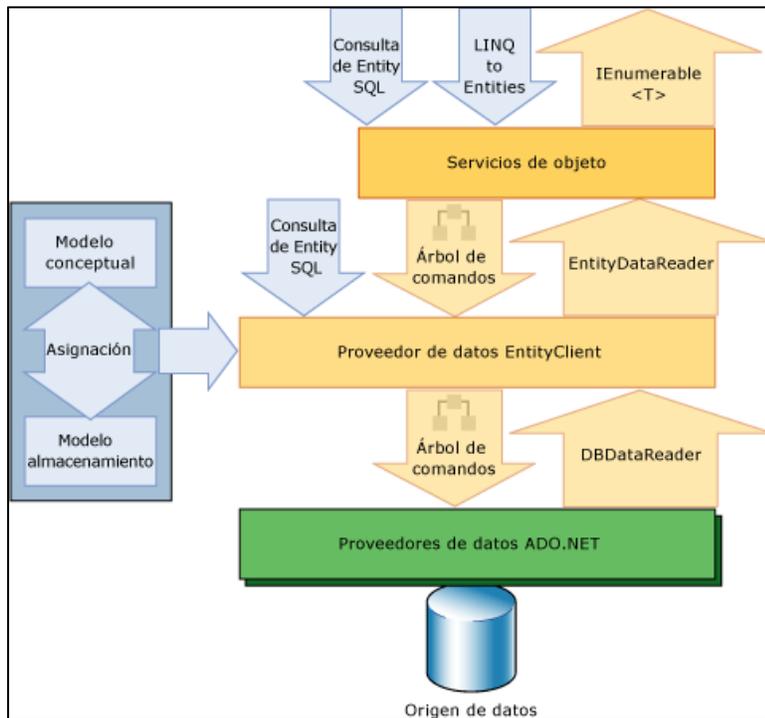


Ilustración 5. Entity Framework [20]

Las herramientas de Entity Data Model pueden generar una clase derivada de `System.Data.Objects.ObjectContext` o `System.Data.Entity.DbContext` que representa el contenedor de entidades definido en el modelo conceptual. Este contexto del objeto proporciona los medios para realizar el seguimiento de los cambios y administrar las identidades, la simultaneidad y las relaciones de los procedimientos almacenados que el programador especifica. [20]

5.7.1. Ventajas de ENTITY FRAMEWORK:

Las aplicaciones pueden funcionar en términos de un modelo conceptual más centrado en la aplicación, que incluye tipos con herencia, miembros complejos y relaciones.

Las aplicaciones están libres de dependencias de codificación rígida de un motor de datos o de un esquema de almacenamiento.

Las asignaciones entre el modelo conceptual y el esquema específico de almacenamiento pueden cambiar sin tener que cambiar el código de la aplicación.



Los programadores pueden trabajar con un modelo de objeto de aplicación coherente que se puede asignar a diversos esquemas de almacenamiento, posiblemente implementados en sistemas de administración de base de datos diferentes.

Se pueden asignar varios modelos conceptuales a un único esquema de almacenamiento.

La compatibilidad con Lenguaje Integrated Query (LINQ) proporciona validación de la sintaxis en el momento de la compilación para consultas en un modelo conceptual [21]



6. Marco Metodológico

En la siguiente sección se detallan el tipo de investigación, métodos y técnicas de recolección y procesamiento de información

6.1. Hipótesis:

Hipótesis de investigación (Hi): El sistema web de gestión del servicio que ofrece la empresa Americable S.A. responde a los objetivos descritos.

6.2. Sistematización de variables

6.2.1. Operacionalización de variables

En el siguiente cuadro se encuentran las variables identificadas por objetivos, con sus respectivos indicadores de evaluación.

OBJETIVOS ESPECÍFICOS	VARIABLES	INDICADOR	DESCRIPCIÓN
1. Determinar los requerimientos funcionales y no funcionales para la automatización de la gestión de los servicios de abonados de la empresa Americable.	Gestión de servicios	<ul style="list-style-type: none"> Ahorro de tiempo en gestión. Fácil manejo de información. Información segura. Agilidad en los procesos. 	Conjunto de actividades o funciones que se realizan al momento de solicitar un servicio
2. Diseñar procesos e interfaz de usuario de acuerdo a los requerimientos determinados para la gestión de los servicios ofrecidos por la empresa Americable.	Requerimientos funcionales y no funcionales	<ul style="list-style-type: none"> Identificación de los requerimientos funcionales y no funcionales Identificación de procesos a automatizar en la empresa. 	Conjunto de funcionalidades, que describen un proceso entrada y salida de información del software y criterios que permiten evaluar operaciones del software, describiendo lo que se hace y como lo hace el sistema
3. Aplicar la metodología de programación en capas y tecnología web para el	Arquitectura del software	Diseño del software en arquitectura distribuida haciendo uso de modelo en capas y tecnologías web.	Modelado de todos los procesos a automatizar



Desarrollo de un sistema web para la gestión del servicio de televisión por cable que ofrece la empresa Americable S.A en la ciudad de Juigalpa Chontales 2016.

funcionamiento requerido del sistema a desarrollar.			
4. Evaluar el cumplimiento de los requerimientos funcionales y no funcionales en el sistema web desarrollado a través de pruebas de validación.	Validación de requerimientos	Distintos tipos de prueba que se aplicaran al sistema para evaluar el funcionamiento	Pasos que se aplican para determinar que los requerimientos han sido cumplidos y que no existen errores en el diseño y funcionamiento

Objetivos	Dimensiones	Datos e-indicadores.
Determinar los requerimientos funcionales y no funcionales para la automatización de la gestión de los servicios de abonados de la empresa Americable.	<ul style="list-style-type: none"> • Organizacional • Tecnología • Proceso 	<ul style="list-style-type: none"> • Roles • Datos de entrada • Datos de salida. • Asociaciones de proceso. • Restricciones. • Procesos • Políticas de privacidad • Seguridad.
Diseñar procesos e interfaz de usuario de acuerdo a los requerimientos determinados para la gestión de los servicios ofrecidos por la empresa Americable.	<ul style="list-style-type: none"> • Organizacional • Tecnología • Proceso 	<ul style="list-style-type: none"> • Casos de uso • Diagramas de flujo • Diagramas E/R • Bocetos.
Aplicar el modelo de programación en capas y tecnología web para el funcionamiento requerido del sistema a desarrollar.	<ul style="list-style-type: none"> • Tecnología • Proceso 	<ul style="list-style-type: none"> • Plataforma de desarrollo • Gestor de base de datos y Lenguaje de programación. • Navegador web • Manual de usuario • Manual técnico



Desarrollo de un sistema web para la gestión del servicio de televisión por cable que ofrece la empresa Americable S.A en la ciudad de Juigalpa Chontales 2016.

<p>Evaluar el cumplimiento de los requerimientos funcionales y no funcionales en el sistema web desarrollado a través de pruebas de validación.</p>	<ul style="list-style-type: none"> • Tecnología. • Procesos. 	<ul style="list-style-type: none"> • Estándar validación de software • Datos de prueba. • Datos reales.
---	--	--

Variable	Datos e Indicadores	Instrumento-Herramientas
Gestión de servicios	<ul style="list-style-type: none"> • Roles • Datos de entrada • Datos de salida. • Asociaciones de proceso. • Restricciones. • Procesos • Políticas de privacidad • Seguridad. 	<ul style="list-style-type: none"> • Entrevista • Guía de observación
Requerimientos funcionales y no funcionales	<ul style="list-style-type: none"> • Documento de especificación de Requerimientos. • Casos de uso • Diagramas E/R • Bocetos. 	<ul style="list-style-type: none"> • IEEE830 • Entrevista • ArgoUML • SQL Server. • Balsamick Moqup.
Arquitectura del software	<ul style="list-style-type: none"> • Plataforma de desarrollo • Gestor de base de datos y Lenguaje de programación. • Navegador web • Manual de usuario • Manual técnico 	<ul style="list-style-type: none"> • SQL SERVER 2008 • Microsoft Visual studio 2012 , (C#,HTML5, ASP.NET, bootstrap 5, CSS3) • Microsoft office 2013 • Google chrome. • Firefox
Validación de requerimientos	<ul style="list-style-type: none"> • Estándar validación de software • Datos de prueba. • Datos reales. 	<ul style="list-style-type: none"> • Estandar IEEE730. • Usuario Final.



6.3. Población y muestra

La población se ha definido como las dos empresas que venden el servicio de televisión por cable en el departamento de Chontales, las cuales son Americable S.A y Claro TV.

Se seleccionó por conveniencia la empresa Americable S.A como muestra, debido a que esta no tiene un sistema que automatice los procesos.

6.4. Tipo de investigación

Existen diversas clasificaciones que nos tratan de dar un orden acerca de los diversos tipos de Investigación que existen, claro una vez seleccionado el tema de investigación y el nivel de esta, el siguiente paso es identificar el tipo de investigación sin embargo es importante aclarar, que ninguna investigación utiliza en forma exclusiva un solo tipo de Investigación sino una combinación de tales para enriquecer el trabajo. [22]

La investigación es de campo porque está enmarcada en un sector en específico el cual es telecomunicaciones, está enfocada a la verificación de la hipótesis y es transversal porque está definido en un tiempo específico. La investigación también es de carácter descriptivo, identificando el problema que se da en la empresa Americable S.A, describiendo las causas, así como la manipulación de manera manual atribuyendo a la pérdida de información y tiempo.

6.5. Métodos y técnicas de recolección de información

Como instrumento principal en la recopilación de datos se utiliza una guía de observación identificando los procesos que se realizan al momento de contratar el servicio y entrevista realizada al propietario de la empresa, es decir a un informante clave. Formulando preguntas para obtener información detallada acerca de la gestión de los servicios, tomando datos que contribuyen a los requerimientos del software y su funcionalidad.

6.6. Técnicas y herramientas de procesamiento

No se ha utilizado ninguna herramienta de software para el procesamiento de datos debido a que el instrumento utilizado ha sido la entrevista, la información obtenida se ha organizado en una matriz elaborada en Microsoft Word.



6.7. Ciclo de vida de desarrollo de software

El Ciclo de vida que se ha utilizado es el propuesto por Ian Somerville compuesto por etapas de desarrollo del software de manera específica y detallada. Estas etapas funcionan de tal forma que el inicio de cada etapa debe esperar a la finalización de la etapa anterior. Al final de cada etapa, el modelo está diseñado para llevar a cabo una revisión final, que se encarga de determinar si el proyecto está listo para avanzar a la siguiente fase. Las etapas que componen este ciclo de vida son las siguientes:

- Identificación de problemas, Oportunidades y Objetivos
- Determinación de los Requerimientos de Información.
- Análisis de las necesidades del sistema.
- Diseño del sistema recomendado.
- Desarrollo y Documentación del software.
- Prueba y Mantenimiento del sistema.
- Implementación y Evaluación del sistema. [14]



7. Marco Administrativo

En esta etapa se establece el tiempo y costo en que se realizó el sistema.

7.1. Cronograma del documento del proyecto

El siguiente cronograma muestra las actividades y la duración de cada una de ellas para llevar a cabo el desarrollo del documento del sistema Americable S.A

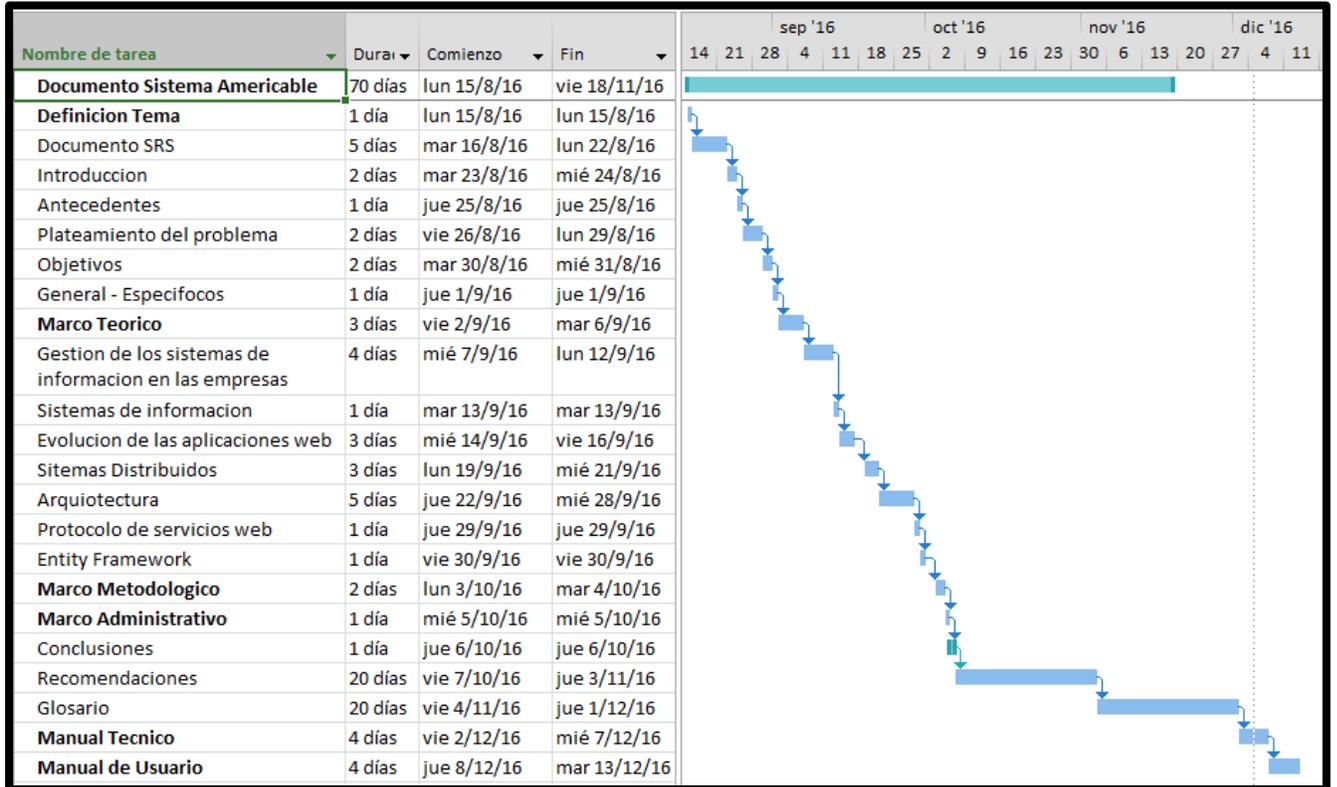


Ilustración 6. Cronograma del Documento del proyecto



7.2. Presupuesto de trabajo

En esta fase se define el costo del sistema de acuerdo al modelo de desarrollo orgánico de COCOMO establecido por Boehm en 1981.

7.2.1. Presupuesto del desarrollo del software

El Modelo Básico de COCOMO'81 estima el esfuerzo y el tiempo empleado en el desarrollo de un proyecto de software usando dos variables predictivas denominadas factores de costo (cost drivers): el tamaño del software y el modo de desarrollo. Las ecuaciones básicas son:

$$\text{Esfuerzo: } PM = A \times (KSLOC)^B$$

Dónde: **PM** es el esfuerzo estimado. Representa los meses-persona necesarios para ejecutar el proyecto.

- **KSLOC** es el tamaño del software a desarrollar en miles de líneas de código.
- **A y B** son coeficientes que varían según el Modo de Desarrollo (Orgánico, Semiacoplado, Empotrado)

$$\text{Tiempo: } TDEV = C \times (PM)^D$$

Dónde: **TDEV** representa los meses de trabajo que se necesitan para ejecutar el proyecto.

- **C y D** son coeficientes que varían según el Modo de Desarrollo (Orgánico, Semiacoplado, Empotrado)

La Tabla 1 muestra la variación de la fórmula de estimación de esfuerzo y cronograma para los tres Modos de Desarrollo:

Modo de Desarrollo	Esfuerzo	Cronograma
Orgánico	$PM=2.4 \times (KSLOC)^{1.05}$	$TDEV=2.5 \times (PM)^{0.38}$
Semiacoplado	$PM=3.0 \times (KSLOC)^{1.12}$	$TDEV=2.5 \times (PM)^{0.35}$
Empotrado	$PM=3.6 \times (KSLOC)^{1.20}$	$TDEV=2.5 \times (PM)^{0.32}$

Tabla 1: Ecuaciones del Modelo Básico de COCOMO 81. [Boehm 1981]

Tabla 1. Ecuaciones del Modelo Básico de COCOMO 81. [Boehm 1981]



- El modelo orgánico se utiliza para proyectos menores a 50 KSLOC líneas de código.
- Modelo Semi-acoplado se utiliza para proyectos intermedios menores a 300 KSLOC.
- Modelo empotrado utilizado para proyectos complejos y se engloban en un entorno de gran innovación que son mayores de 300 KSLOC.

El modo de desarrollo que se utiliza de acuerdo a la estimación de líneas de código es el orgánico debido a que el proyecto es < 50 KSLOC.

La tabla 2 muestra la cantidad de líneas de código por lenguajes de programación utilizados para el desarrollo del sistema.

Lenguaje de programación/ Lenguajes de Marcado	Líneas de Código (LDC)
CSS	2654
HTML	1795
.Net C# 2012	3258
SQL Server 2012	733
JavaScript	251
Total (LDC)	8,691
Total en KSLOC	8.691

Tabla 2. Total de líneas de código por programa

Antes de calcular el esfuerzo se debe realizar la conversión del total de líneas de código dado que la fórmula requiere obtener en KSLOC el total de líneas de código.

$$\text{KSLOC} = \text{Cantidad de líneas de código} / 1000$$

$$\text{KSLOC} = 8,691 / 1000 = 8.691$$

$$\text{Esfuerzo: PM} = A \times (\text{KSLOC})^B$$

$$\text{Esfuerzo: PM} = 2.4 \times (8.691)^{1.05}$$

$$\text{Esfuerzo: PM} = 23.239 \text{ meses-personas}$$

$$\text{Cronograma: TDEV} = C \times (\text{PM})^D$$

$$\text{Cronograma: TDEV} = 2.5 (23.239)^{0.38}$$

$$\text{Cronograma: TDEV} = 8.262$$



Una vez calculado el esfuerzo se calcula la productividad del proyecto tomando la cantidad total de líneas de códigos.

Productividad:

PR (Productividad) = Líneas de código / Esfuerzo

PR = LDC /PM = 8,691/23.239= 373.98 LDC/Mes

P (Personal Promedio) = PM / TDEV

P= 23.239/8.262=2.81 Personas

Según estas cifras será necesario un personal promedio de 2.81 personas trabajando alrededor de 8.262 meses, debido a que las personas no se pueden dividir y el tiempo es limitado el cual equivale a 1 semestre, se realiza un cálculo de costo en base a esto, conociendo que según COCOMO el tamaño del equipo desarrollador y la cantidad de tiempo son coherentes con lo obtenido.

Costo:

Cada persona trabaja en promedio 2 horas al día de lunes a jueves, teniendo en el mes un total de 32 horas por persona.

Total, horas trabajadas= (Cantidad de personas * cantidad de horas) * cantidad mes

Total horas trabajadas= (3*32) *4 =384 Horas totales del proyecto

Horas Trabajadas Por persona = 32H * 4Meses = 128 Horas

LDC totales por persona =8,691/ 3 = 2,897

LDC por hora por persona = 2,897 / 32 = 90.53

Según el modelo COCOMO se llegó a la conclusión que para el desarrollo del sistema de control de Abonados de Americable se necesita un total de tres personas trabajando 128 horas por un tiempo limitado de 4 meses.



Costo del proyecto

Para calcular el costo del proyecto se ha establecido la hora de programación con un precio de 15 \$, Los viáticos por persona tienen un costo de 40\$, electricidad 120 \$, utilerías 100 \$ y el costo mensual del internet 20\$.

Gasto	Precio Unitario	Cantidad	Precio Total
Pago a desarrolladores por hora	\$15	128 Horas	\$ 4320
Viáticos	\$40	3 Personas	\$ 120
Electricidad			\$ 120
Utilerías			\$ 100
Internet	\$ 20	4 Meses	\$ 80
Licencias de software			\$300
Precio Total Del Proyecto			\$ 5,040

Tabla 3. Presupuesto del proyecto



8. Marco de Resultados

En la etapa siguiente se describen los resultados obtenidos una vez que se realizó el sistema.

8.1. Cronograma de actividades

En la imagen se muestran las etapas de desarrollo y la cantidad de tiempo que fue necesario para la realización de sistema.

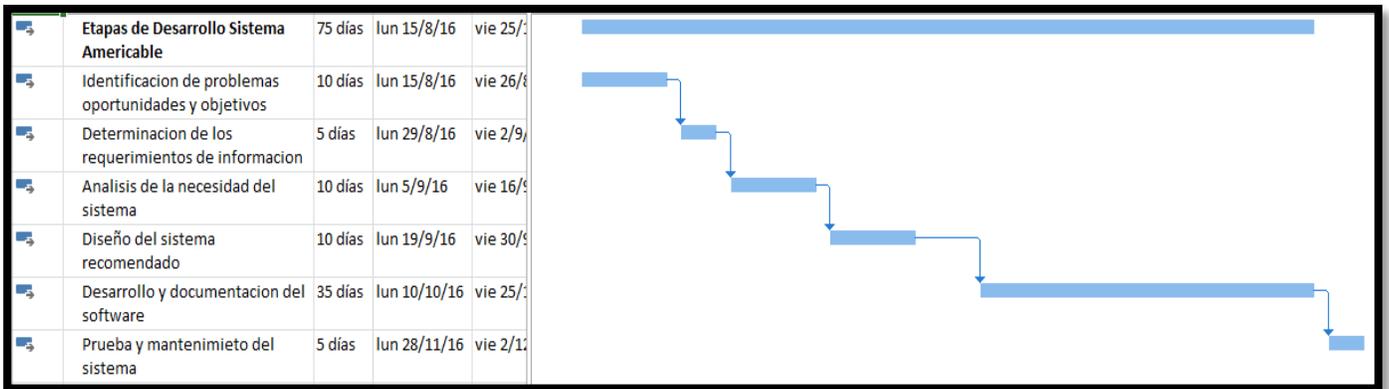


Ilustración 7. Cronograma de Actividades

8.2. Descripción de los resultados obtenidos según los objetivos

Este punto consiste en la descripción de las herramientas, métodos e instrumentos que se han sido utilizados para llevar a cabo el cumplimiento del objetivo general de la investigación dando respuesta a los objetivos específicos.

Para obtener los resultados de los objetivos se aplicaron las etapas del ciclo de vida del desarrollo de software mencionadas en la metodología. (Ver pág. 30.) Así como también otros instrumentos de evaluación para obtener la información específica del software.

Identificación de problemas, Oportunidades y Objetivos:

En esta etapa se aplicaron instrumentos de recolección de información como:

- La entrevista que ayuda a conocer mejor las necesidades de la empresa (ver Anexo 1).
- La observación para obtener información sobre cómo se realizan los procesos dentro de la empresa (ver Anexo2).



Es necesario conocer el funcionamiento del negocio para saber la problemática a la que se enfrenta y proseguir a plantear los objetivos y etapas que darán respuesta a ellos.

Determinación de los Requerimientos de Información: al iniciar la aplicación de esta etapa ha sido necesario finalizar la identificación de problemas, oportunidades y objetivos. En esta fase se aplicó el estándar SRS IEEE830 (ver pag.61, Anexo 8) como instrumento para la obtención de requerimientos funcionales y no funcionales proporcionados por el cliente. Así mismo se obtendrá los procesos ha automatizar el cual son los siguientes:

- Registro de Clientes
- Registro de Contratos del servicio de televisión por cable de cada cliente
- Registro de Empleados (Técnicos)
- Pago del servicio de televisión por cable
- Reportes de averías por los clientes
- Asignar averías a dar solución a los técnicos
- Facturación
- Informe de estado de cuenta a los clientes
- Búsquedas ágiles

De esta manera se cumple con el primer objetivo que trata sobre la determinación de los requerimientos del sistema. (Ver página 5.)

Análisis de las necesidades del sistema: Para ejecutar esta etapa es necesario haber definido con anticipación los requerimientos funcionales y no funcionales del sistema, los procesos que se necesitan automatizar en la empresa, el cual servirán de insumo para realizar diagramas de casos de usos (ver anexo 3) correspondientes a cada una de las funciones que deberá realizar el sistema según el tipo de usuario. Es importante señalar que se ha utilizado la aplicación Star UML como herramienta de apoyo al modelado de la estructura lógica del sistema.

Los diagramas de casos de uso sirven para especificar la comunicación y el comportamiento del sistema mediante la interacción de los usuarios o lo que es igual muestra la relación entre los actores y el sistema.



Diseño del sistema recomendado: Establecida la estructura lógica del sistema se realiza el diseño de la interfaz, por eso se debe aplicar anticipadamente la etapa de análisis de las necesidades para tener un enfoque claro de cómo debe funcionar y aplicarlo a la maquetación de las interfaces.

Se crearon bocetos para la vista o interfaz del sistema (ver anexo5) teniendo en cuenta la estructura lógica propuesta en los diagramas de casos de usos. Además de eso se creó la base de datos relacional que debe almacenar los datos de la empresa.

Los bocetos son una representación visual simplificada de los elementos que contendrá el sistema y han tenido como objetivo:

- Ser el punto de partida para discutir las ideas del proyecto.
- Mostrarle al cliente de forma aproximada que elementos vera en el sistema.

Las aplicaciones que han sido utilizadas como apoyo para culminar esta etapa son las siguientes:

- Mockups Balsamiq como herramienta para el diseño de bocetos.
- SQL Server como herramienta para el diseño de la base de datos.

Al finalizar con las etapas de análisis de las necesidades del sistema y Diseño del sistema recomendado se ha cumplido con el segundo objetivo de la investigación que se refiere al diseño de los procesos e interfaz de acuerdo a los requerimientos del sistema (ver página. 5.).

Desarrollo y Documentación del software: En esta fase se codificaron cada una de las interfaces realizadas en el diseño del sistema recomendado, se ha utilizado el modelo cliente-servidor y se organizó el proyecto en tres capas: presentación, negocio y lógica, datos.

Capa de presentación: es la que se comunica directamente con el usuario, se comunica únicamente con la capa de negocio.

Capa de negocio: es donde se perciben las peticiones de los usuarios y se envían las respuestas tras el proceso, también es llamada “capa lógica” porque es aquí donde se establecen todas las reglas que deben cumplirse, dicho de otra manera, sirve como intermediaria entre la capa de presentación y capa de datos en el envío y respuesta de las solicitudes.

La capa de datos es donde se recogen los datos y se encarga de acceder a los mismos, esta recibe solicitudes de almacenamiento, modificación o recuperación de información desde la capa de negocio.



Las aplicaciones que se han utilizado para la codificación del sistema son las siguientes:

- Visual estudio 2012 es el contenedor de los lenguajes de programación C# y ASP.NET que se han utilizado para la codificación del sistema. También se ha utilizado Bootstrap 4 que es un framework y CSS3 (hoja de estilo en cascada) para facilitar el diseño.

La documentación del software incluye la elaboración de un manual de usuario (Ver página 129, Anexo 11). para la comprensión del funcionamiento del sistema que busca brindar asistencia a los sujetos que usan el sistema y un manual técnico que va dirigido a un público con conocimientos sobre el área, este describe como han sido desarrollado el sistema (Ver página. 102, Anexo 10).

Culminada la etapa de desarrollo y documentación de software da respuesta al tercer objetivo (ver página 5) del trabajo de investigación que reside en aplicar la metodología de programación en capas y tecnología para el funcionamiento del sistema.

Pruebas y mantenimiento del sistema: Una vez finalizada la codificación del sistema se da inicio a las pruebas y mantenimiento del sistema. Para la ejecución de esta etapa se realizan pruebas unitarias de caja blanca y caja negra con el fin de determinar el buen funcionamiento y la validación, verificación de datos dentro del sistema, de esta manera encontrar los errores antes de instalarlo. Estas pruebas son realizadas a través del plan de verificación y validación IEEE730 (Ver página 83, Anexo 9).

El mantenimiento se debe realizar cada cierto tiempo cuando el sistema se está implementado por completo dentro de la empresa para evitar que ocurran errores inesperados.

Evaluación del sistema: La evaluación se ejecuta en cada una de las etapas de desarrollo del sistema haciendo uso de la herramienta IEEE730 que consiste en un plan de verificación y validación, define lo que es un software de alta calidad, proporciona los requisitos mínimos para el aseguramiento de la calidad del software. De igual forma se evalúa el cumplimiento de los requerimientos funcionales y no funcionales en el sistema web.

Una vez realizadas las pruebas y evaluación del sistema se puede afirmar que se da por cumplido el último objetivo de la investigación que consiste en evaluar el cumplimiento de los requerimientos del sistema web a través de pruebas de validación (ver página 5).



9. Conclusiones

El proyecto de desarrollo de un sistema web para el control del servicio de televisión por cable ofrecido por la empresa Americable S.A, ha cumplido los puntos que se describen a continuación:

- 1) Se ha logrado determinar los requerimientos funcionales y no funcionales para la automatización en la gestión de servicios a los abonados en la empresa Americable S.A. aplicando el estándar SRS IEEE830 permitiendo definir las funciones, componentes del sistema, procesos que intervienen y diseñar conforme los criterios para operar el sistema.
- 2) El diseño del sistema ha sido desarrollado de acuerdo a los requerimientos funcionales y no funcionales propuestos por el cliente permitiendo una interfaz intuitiva y de fácil acceso a los usuarios, realizando la estructura lógica mediante los diagramas de casos de usos, base de datos; y bocetos.
- 3) El sistema ha sido desarrollado en C# y ASP.NET haciendo uso de tecnología web aplicando la metodología en capas para el funcionamiento requerido del sistema.
- 4) Se ha realizado el proceso de validación de los requerimientos funcionales y no funcionales en cada una de las etapas de desarrollo del sistema a través de pruebas de validación con el uso del estandar IEEE730 que es un plan de validación y verificación de software durante la aplicación de las pruebas que se aplicaron a diferentes módulos del sistema los resultados de las pruebas y los objetivos que se pretende alcanzar:
 - Identificar la información de proyecto y los componentes del software que fueron verificados
 - Detalles de los requerimientos recomendados verificados
 - Estrategias de verificación que serán usadas
 - La disminución de riesgos
 - Validaciones de cada administradores del sistema
 - Seguridad del Sistema
 - Corregir funciones incorrectas o ausentes

El desarrollo del proyecto ha permitido la automatización del servicio de televisión por cable ofrecido por la empresa Americable, de modo que la gestión de los abonados se realiza en poco tiempo y la distancia no será un obstáculo para que el usuario reporte una avería y verifique su estado de cuentas.



10. Recomendaciones

A continuación, se describirán las recomendaciones realizadas, dirigidas para las personas que manipularan el sistema y no tienen experiencia con las nuevas tecnologías de información:

- El administrador del sistema debe realizar el proceso de registro de forma correcta llenando los campos conforme a lo requerido.
- Para registrar a un cliente el administrador deberá realizar el contrato del servicio.
- Los usuarios deben cerrar sesión después de manipular al sistema.
- El administrador deberá realizar el proceso de modificación o eliminación solamente en caso de ser necesario.
- Deben hacerse pruebas al sistema antes de ponerlo en funcionamiento.
- El mantenimiento del sistema deberá llevarse a cabo de manera rutinaria durante toda su vida útil.
- Para implementar el sistema se debe capacitar al usuario administrador y empleado sobre el manejo del sistema.
- En caso de fallas en el sistema comunicarse con los desarrolladores.
- Por ser primera vez que en la Empresa se implementará este sistema se recomienda establecer normas de uso para los usuarios que están establecidos por el administrador.
- Cuando el sistema sea implementado, se recomienda designar un personal que tenga conocimientos en programación de sitio web y manejo de base de datos, para cuando se presente problemas o se necesite la actualización de datos, este realice esta función.



11. Glosario de Términos

Automatizados: Procesos que se manejan de manera automática y que anteriormente se controlaban de manera manual.

Código: líneas de código del sistema que especifican las instrucciones que debe realizar.

Cronograma: calendario de trabajo del conjunto actividades y su duración que determinan el tiempo total en que se llevara a cabo.

Interfaz: Pantallas, interfaces o presentación del sistema que permiten la interacción directa con los usuarios.

Metodología: Conjunto de métodos, estructuras o procedimientos que se realizan para el desarrollo de la investigación.

Módulos: Distintas ventanas que posee el sistema navegable por los usuarios.

Parámetros: Variables que sirven para identificar métodos en programación, cada uno de ellos mediante su valor.

Programación: es el proceso de diseñar, codificar las interfaces y procesos tales como requerimientos funcionales y no funcionales que realiza el sistema.

Software: conjunto de aplicaciones instaladas en una computadora.

ERS: Especificación de Requerimientos de software.

E/R: Entidad Relación.

IEEE: Instituto de Ingenieros Eléctricos y Electrónicos

RF: Requerimientos Funcionales.

RNF: Requerimientos no Funcionales.

SCAA: sistema de control de abonados de Americable

Gestión: Administración de procesos del servicio.

Arquitectura: conjunto de elementos que constituyen un diseño o modelo de trabajo.



Cliente: Usuarios, computadoras y aplicaciones que se comunican con el sistema realizando peticiones.

Servidor: Aplicación en ejecución capaz de atender peticiones por los clientes y devolver una respuesta.

Servicio: Actividades, procesos que brinda la empresa a los clientes.

TICS: Tecnologías de información y comunicación.

Sistema: Aplicación, software que ejecuta de manera automática los procesos de la empresa.

Cascada: Procedimientos que se realizan de manera continua.

Estándar(es): Estructura de trabajo validada por profesionales.

Procesos: actividades que se ejecutan en la empresa.

Concurrencia: Peticiones de los clientes que se ejecutan simultáneamente.

Capas: organización jerárquica del sistema.

Herramientas: Aplicaciones, estándares, metodologías que se utilizan para el desarrollo de la investigación.

Rigurosa: Que se cumple o hace cumplir las normas establecidas en el proyecto

Incorporación: acción o efecto de incorporar las partes del sistema en uno solo

Escalables: es un término usado en tecnología para referirse a la propiedad de aumentar la capacidad de trabajo o de tamaño del sistema sin comprometer su funcionamiento y calidad.

Estandarización: es el proceso mediante el cual se realizaron las actividades es el proceso standard a través del cual se tiene que actuar o proceder dentro del proyecto

Lenguajes: procesos por los cual se escribe, se prueba, se depura y se mantiene el código fuente del sistema informático.

Distribuido: es una colección de computadoras separadas físicamente y conectadas entre sí por una red de comunicaciones; cada máquina posee sus componentes de hardware y software que el programador percibe como un solo sistema.



Remotamente: persona que esta distante o lejano en el tiempo y en el espacio pero que puede hacer uso del sistema.

Lógica: pieza fundamental en la que se basa el programador para poder entender y dar solución a un problema, para posteriormente implementarla en forma de programas

Recuperar: Backups o recuperación de información haciendo copias de seguridad, alojamiento de datos en algún dispositivo o en la nube.

Protocolo: reglas utilizadas por las computadoras para comunicarse entre sí mediante el uso de la red.

Transferencia: Enviar ficheros o paquetes de archivos haciendo uso de la red.

Red: conjunto de dispositivos y ordenadores conectados entre sí que comparten información haciendo uso de la web.

Analista: Analista de sistemas o programador en el área de informática, encargado de desarrollo de aplicaciones en lo que respecta al diseño y obtención de algoritmos.

Modelado: Analizar los diagramas que componen UML y ofrecer acercamientos a casos de usos guiados para diseñar la estructura lógica del sistema.

Plataforma: sistemas que sirven como la base para hacer funcionar determinados módulos o componentes de los sistemas. Se puede definir el sistema operativo, navegadores, .NET, como plataformas.

Requerimientos: requisitos que explican lo que debe hacer un sistema con respecto al funcionamiento.

Subsistema: Pequeñas elementos interrelacionados que en, si mismo, es un sistema, pero son parte de un sistema superior.

Telecomunicaciones: sistemas que permiten la comunicación, de forma análoga.

Implantar: instalación de un sistema en una determinada empresa o negocio.

Centralización: componentes que operan desde un mismo lugar o ubicación central desde donde se ejecutan todos los procesos de un sistema.



12. Referencias

- [1] Directv, «DIRECTV,» [En línea]. Available: <http://bit.ly/1K12VPN>.
- [2] «Claro TV,» [En línea]. Available: <http://bit.ly/2kbOvYD>. [Último acceso: 7 noviembre 2016].
- [3] J. J. M. M. (CV), «CONTRIBUCIÓN DE LOS SISTEMAS DE INFORMACIÓN A LA ECONOMÍA DE LAS EMPRESAS,» Tamaulipas, 2013.
- [4] M. Morales Martínez, «Analítica web para empresas: arte, ingenio y anticipación,» Editorial UOC, 2010.
- [5] K. Rodríguez Perojo y R. Ronda León, El web como sistema de información, vol. Volumen 14 n, La Habana: Scielo Cuba, 2007, p. 24.
- [6] H. R. Varian, «Sistemas de Información,» de *Fundamentos de diseños de bases de datos*, California, Universidad de Granada, p. 22.
- [7] S. D. ROGER PRESSMAN, «SISTEMAS DE INFORMACION,» [En línea]. Available: <http://bit.ly/1v240T2>. [Último acceso: 11 AGOSTO 2016].
- [8] INFORMATICAHOY, 2007-2016. [En línea]. Available: <http://bit.ly/1hQ5z0K>.
- [9] F. Berzal, F. J. Cortijo y J. C. Cubero, Desarrollo Profesional de Paginas Web con ASP.NET.
- [10] J. F. Muñoz. [En línea]. Available: <http://bit.ly/2kbe5xl>.
- [11] K. E. Kendall y J. E. Kendall, Analisis y Diseño de sistemas, Sexta Edicion ed., G. T. Mendoza, Ed., Naucalpan de Juarez, 2005, p. 752.
- [12] S. L. Mora, Programacion de aplicaciones web: Historia, Principios básicos y clientes web., E. C. Universitario, Ed., San Vicente: Editorial Club Universitario, 2002.
- [13] I. E. T. Aroquipa, «Curso».
- [14] I. Sommerville, Ingenieria del Software, Septima Edicion ed., M. Martin-Romo, Ed., Madrid: PEARSON EDUCACION, S.A., 2005, p. 712.
- [15] M. I. C. E. S. Rebaza, Creacion de Aplicaciones Distribuidas con PowerBuilder y EAServer, vol. I, Chimbote, 2008, p. 89.
- [16] S. L. Mora, Programacion en Internet:Clientes Web, E. C. Universitario, Ed., San Vicente: Club Universitario, 2001.
- [17] O. C. a. i. affiliates, «ORACLE,» Oracle Corporation, 2010. [En línea]. Available: <http://bit.ly/2j2YHRy>. [Último acceso: 06 Octubre 2016].



- [18] L. castro, «Ordenadores y portatiles web,» 25 septiembre 2009. [En línea]. Available: <http://abt.cm/2fE1AFq>. [Último acceso: 12 agosto 2016].
- [19] Microsoft, «Microsoft,» 2016. [En línea]. Available: <http://bit.ly/1TIIItc7>. [Último acceso: 06 Octubre 2016].
- [20] microsoft, «ENTITY FRAMEWORK,» 2015. [En línea]. Available: <http://bit.ly/2k984wK>. [Último acceso: 12 AGOSTO 2016].
- [21] C. t. h.-f. a. l. u. a. C. C. A. S.-A. 3. License, «ventajas de entity framework,» 2015. [En línea]. Available: <http://bit.ly/2jgKjD5>. [Último acceso: 24 septiembre 2016].
- [22] Deymor B. Centty Villafuerte, «eumed.net,» Servicios Academicos internacionales S.C., [En línea]. Available: <http://bit.ly/1N4mwlu>. [Último acceso: 1 Febrero 2017].
- [23] A. Mocholí, «yeePLY,» 15 septiembre 2015. [En línea]. Available: <https://www.yeePLY.com/blog/6-tipos-desarrollo-de-aplicaciones-web/>. [Último acceso: 30 septiembre 2016].
- [24] Calderon Neyra, «monografias.com,» Año 2002. [En línea]. Available: <http://www.monografias.com/trabajos16/sistemas-distribuidos/sistemas-distribuidos.shtml>. [Último acceso: 30 septiembre 2016].
- [25] D. C. T. Dantas, «UNIVERSIDAD FEDERAL DEL RIO DE JANEIRO,» 17 octubre 2007. [En línea]. Available: http://www.gta.ufrj.br/grad/07_2/daniel/. [Último acceso: 30 septiembre 2016].
- [26] R. E. Ingeniería de software. Un enfoque practico pressman, «modelo de desarrollo de software,» PRESSMAN, Raño , 2002. [En línea]. Available: http://moodle2.unid.edu.mx/dts_cursos_md/lic/IEL/SI/AM/06/Modelos.pdf. [Último acceso: 30 septiembre 2016].
- [27] «univercidad del cauca,» Colombia, [En línea]. Available: fceca.unicauca.edu.co/old/siconceptosbasicos.htm. [Último acceso: 12 AGOSTO 2016].
- [28] «WIKIPEDIA,» CLARA BLANCHAR, 20 7 2014. [En línea]. Available: <https://es.wikipedia.org/wiki/Empresa>. [Último acceso: 12 AGOSTO 2016].
- [29] R. Lapiedra Alcamí, . C. Devece Carañana y . J. Guiral Herrando, «Introducción a la gestión de sistemas de información en la empresa,» Primera Edicion ed., Universitat Jaume I. Servei de Comunicació i Publicacions, 2011, p. 72.
- [30] U. P. E. Libertador, «Manual de Trabajos de Grado,» 2006. [En línea]. Available: <https://espaciovirtual.wordpress.com/2007/08/11/101-terminos-de-investigacion-cientifica/>. [Último acceso: 04 10 2016].

ANEXOS

Índice de Anexos

Anexo 1. Resultados de la entrevista	1
Anexo 2. Guía de observación	2
Anexo 3. Casos de Uso	3
Anexo 4. Diagrama de la Base de Datos	4
Anexo 5. Bocetos	5
Anexo 6. Mapa de navegación SCAA	10
Anexo 7. Diccionario de Datos	11

Índice de Tablas de los Anexos

Tabla 4. Resultados de la entrevista	1
Tabla 5. Guía de observación	2
Tabla 6. Tb Usuario.....	11
Tabla 7. Tb Pago	11
Tabla 8. Tb Contrato	11
Tabla 9. Tb Averías.....	12
Tabla 10. Tb Soportes	12

Índice de Ilustraciones de los Anexos

Ilustración 8. Caso de uso Cliente	3
Ilustración 9. Caso de uso administrador	3
Ilustración 10. Caso de uso Empleado/Técnico	4
Ilustración 11. Base de datos del sistema.....	4
Ilustración 12. Pantalla de inicio del sistema	5
Ilustración 13. Menú de inicio cliente y Estado de cuenta.....	5
Ilustración 14. Reporte de Avería realizada por el cliente	6
Ilustración 15. Menu administrador	6
Ilustración 16. Interfaz de Registro de contrato	7
Ilustración 17. Registro de Empleados/Técnicos	8
Ilustración 18. Registro de pagos	9
Ilustración 19. Soporte de averías/Soporte Técnico.....	9
Ilustración 20. Mapa de Navegación del sistema.....	10

Instrumentos de recolección de información.

Anexo1. Resultados de la entrevista

N°	Pregunta	Respuesta
1	¿Cuál es su cargo dentro de la empresa Americable?	Administrador
2	¿Qué tipos de servicios ofrecen?	Solo TV por cable
3	¿Existe algún sistema que controle los abonados?	No
4	¿Qué datos solicita para una crear un nuevo cliente?	Nombre, dirección, Fecha de inicio, numero de cedula
5	¿Cómo verifica que clientes están pendientes de pago?	Mediante Cobrador
6	¿Cómo se comunica el cliente con Uds. para reportar averías?	Mediante llamadas telefónicas
7	¿Cuánto tiempo se tarda en promedio al momento de crear nuevas líneas?	-----
8	¿Qué necesita el cliente para poder hacerse de una cuenta? (realizan alguna verificación, de que tipo)	Nada
9	¿Cuánto meses debe acumular el cliente para realizarle el corte del servicio?	3 Meses
10	¿Necesitan realmente de un software que controle las cuentas de los clientes?	Si
11	¿Cómo realizan el proceso de cobranza? (da recibo, o genera factura algún software)	Visita del Cobrador
12	¿Cuántos tipos de contratos existen? (contrato de acuerdo a la cantidad de canales o meses pagados por adelantados)	Solo un tipo de contrato
13	¿Se paga aparte por la conexión de la línea o va incluida en la mensualidad?	Por aparte la conexión
14	¿Se cobra la reconexión en caso de corte?	No
15	¿Se paga por la cantidad de tv que tiene el cliente?	No
16	¿Número de Clientes en la empresa?	Promedio
17	¿Cuántos TV incluye el contrato? ¿Se cobran adicionales?	Los que desee el cliente.

Tabla 4. Resultados de la entrevista

Anexo2. Guía de observación

N°	Acciones evaluadas	Si	No	Observación
1	Existencia de sistema que controle los abonados		X	Es necesario la implementación de un sistema para la automatización de los procesos de gestión y de esta manera permitir una ventaja competitiva en el negocio.
2	Verificación de existencia para nuevas conexiones		X	No se verifica si antes tuvo el servicio instalado
3	Datos solicitados para la apertura de una nueva conexión	X		Nombre - Apellidos -Dirección
4	Verificación de cliente para realizar pago	X		Se verifica mediante la búsqueda en libros
5	La empresa cuenta con plan ante posibles pérdidas de información de los clientes		X	Si la empresa pierde de alguna manera la información de los no tiene manera de recuperar los datos
6	Se tiene control de acceso al área donde se encuentra la información de los clientes		X	Los datos de los clientes son manipulados en donde encuentran espacio para hacerlo
7	Verificación de los clientes retrasados de pagos	X		Se verifica mediante la búsqueda en el libro de pagos
8	Se reparan las averías en tiempo y forma	X		Se envía un técnico a solucionar el problema
9	El pago del servicio se realiza mediante cobranza a casa de los clientes y mediante visita de los clientes a la empresa	X		Cuando el pago se realiza mediante visita a los clientes se pega la copia del recibo al libro de pagos.

Tabla 5. Guía de observación

Modelado del Sistema.

Anexo3. Casos de Uso.

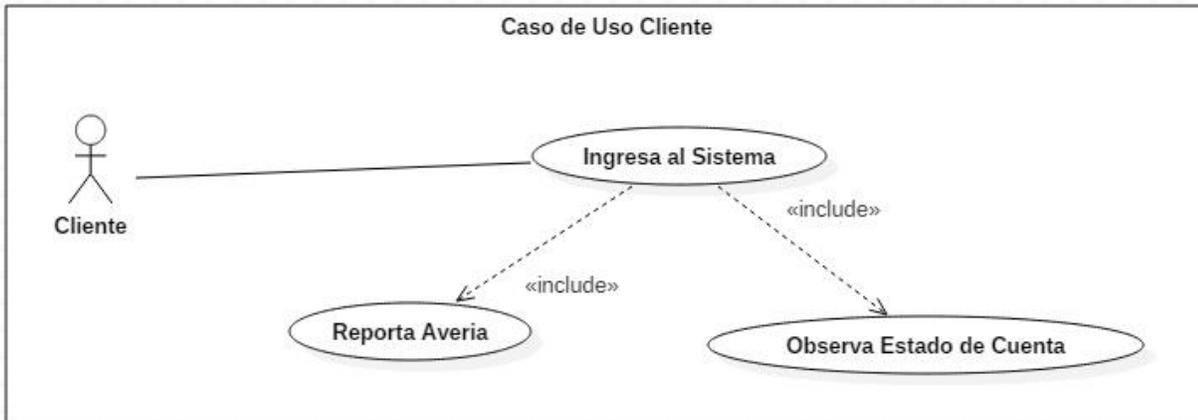


Ilustración 8. Caso de uso Cliente

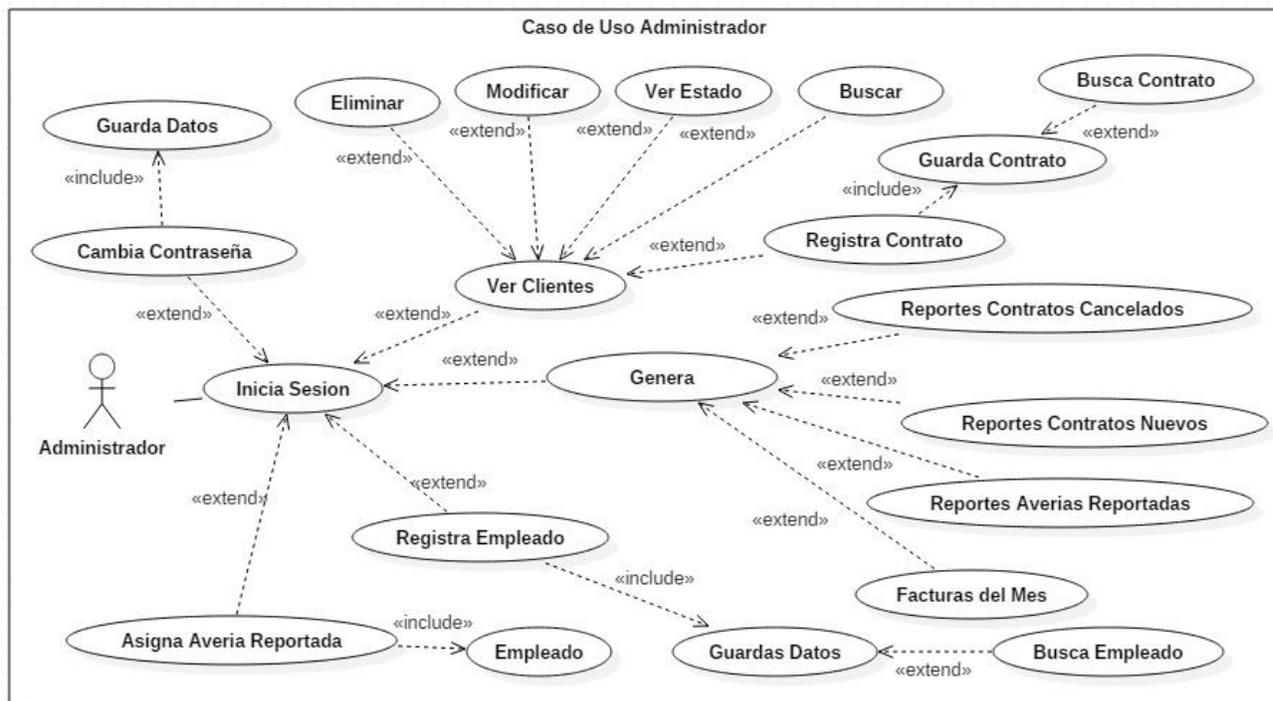


Ilustración 9. Caso de uso administrador

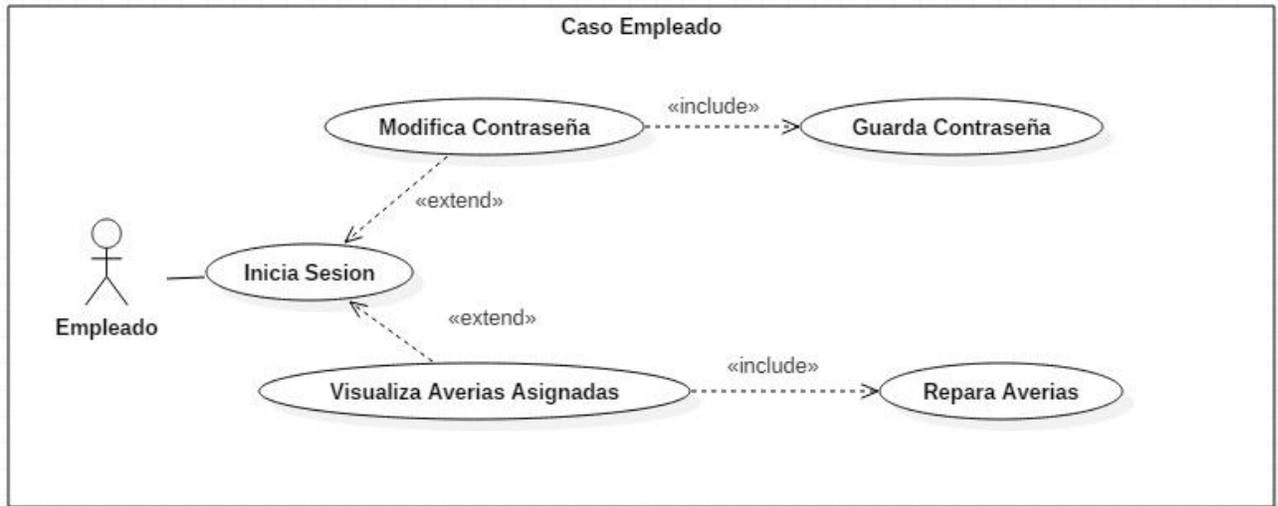


Ilustración 10. Caso de uso Empleado/Técnico

Anexo4. Diagrama de la Base de Datos.

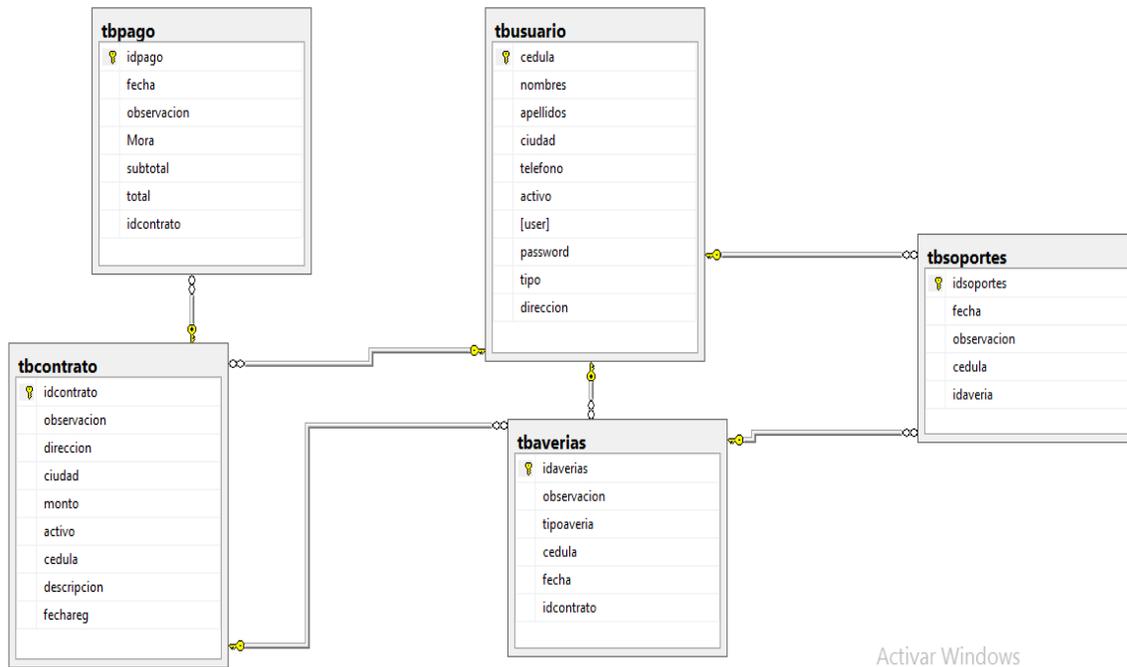


Ilustración 11. Base de datos del sistema

Diseño de Interfaces

Anexo5. Bocetos.

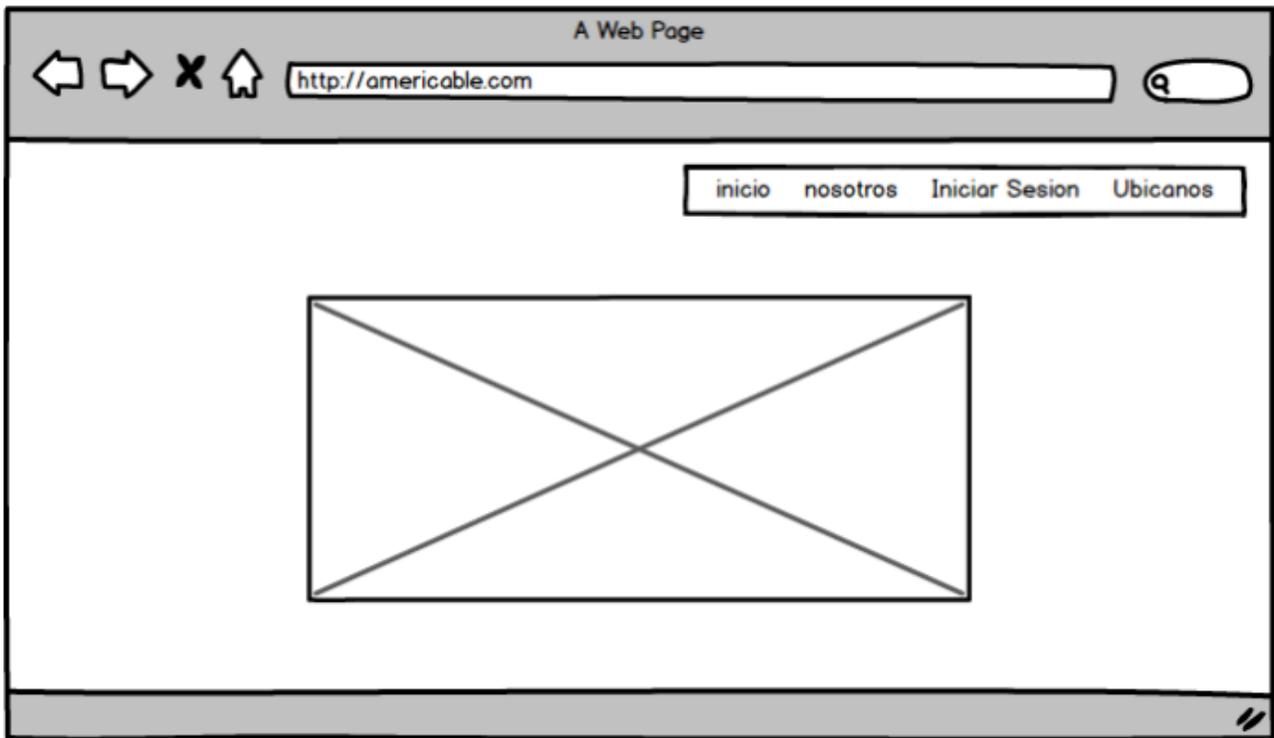


Ilustración 12. Pantalla de inicio del sistema

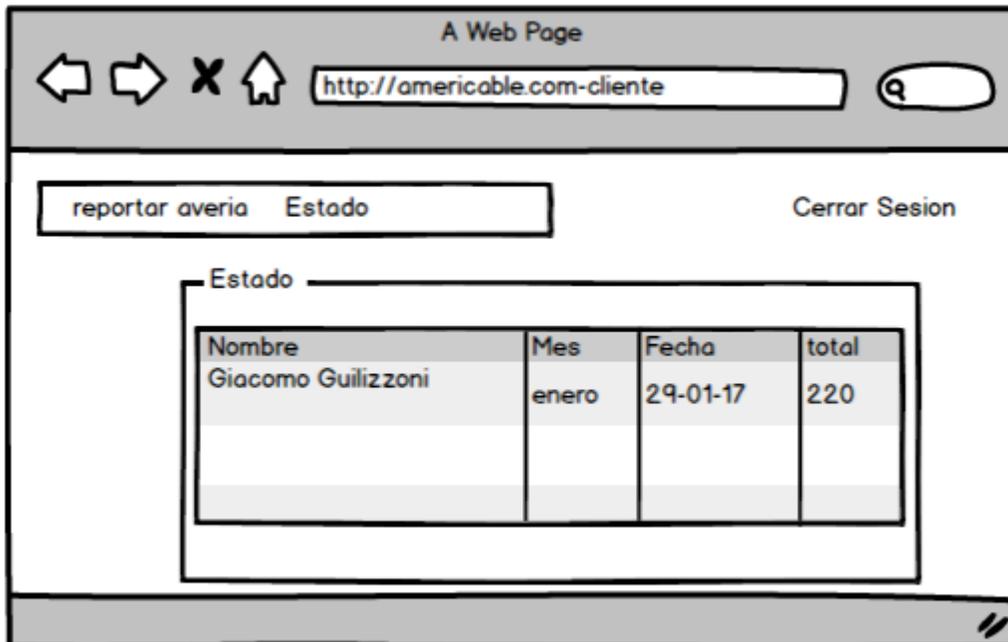


Ilustración 13. Menú de inicio cliente y Estado de cuenta

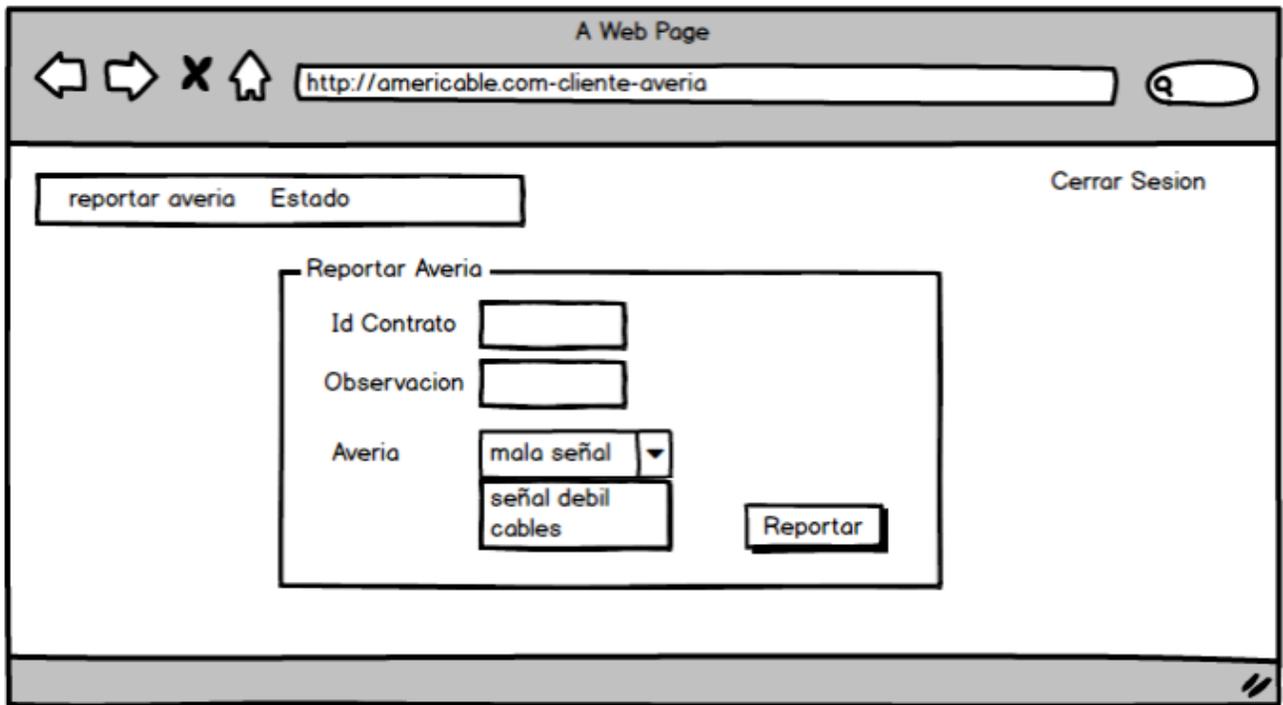


Ilustración 14. Reporte de Avería realizada por el cliente

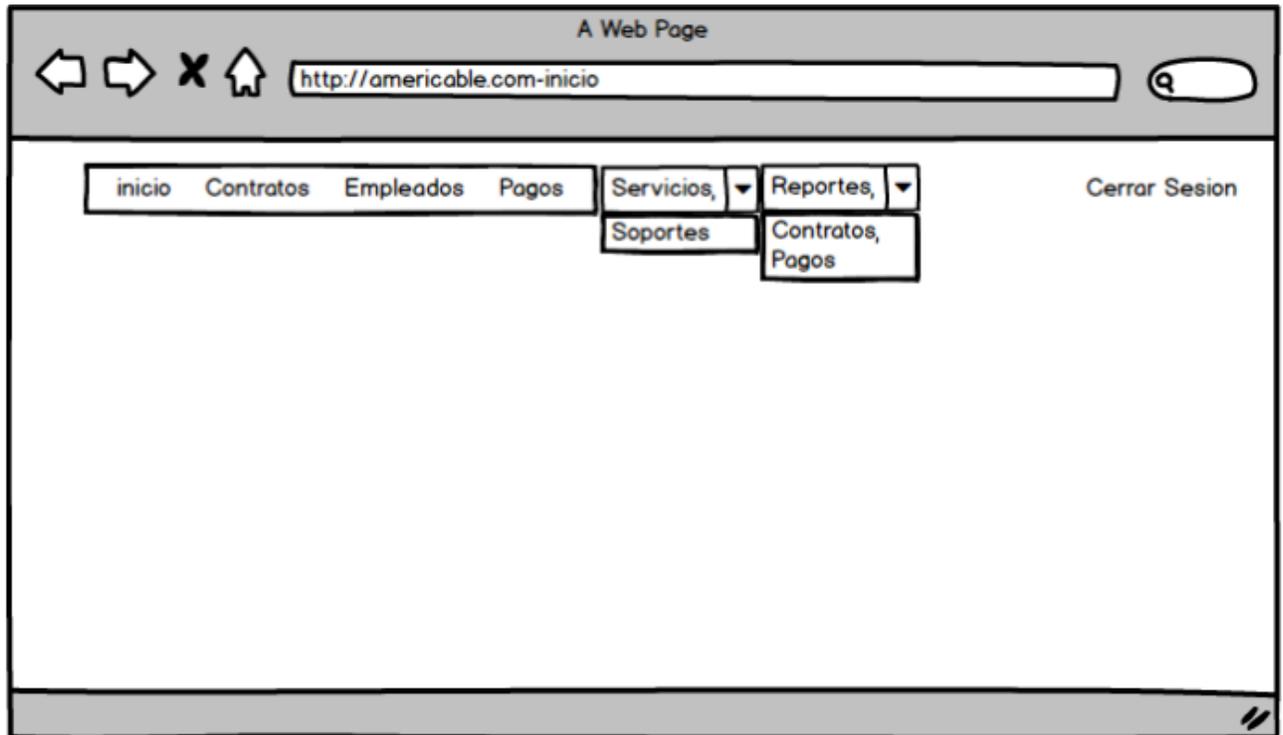


Ilustración 15. Menú administrador

A Web Page

http://americable.com-contratos

inicio Contratos Empleados Pagos Servicios Reportes Cerrar Sesión

Registrar nuevo contrato

cedula cliente

nombre cliente

apellidos cliente

direccion cliente

ciudad cliente

telefono cliente

Servicio Contratado

servicio

ciudad

direccion

observacion

costo

Guardar

Q buscar

Nombre	monto	telefono	cedula
Giacomo Guilizzoni	220	2512-2564	XXXX
Marco Botton	220	8545-2565	xxxx
Mariah Maclachlan	220	2512-5862	XXXX
Valerie Liberty	220	2512-9852	xxxx

Ilustración 16. Interfaz de Registro de contrato

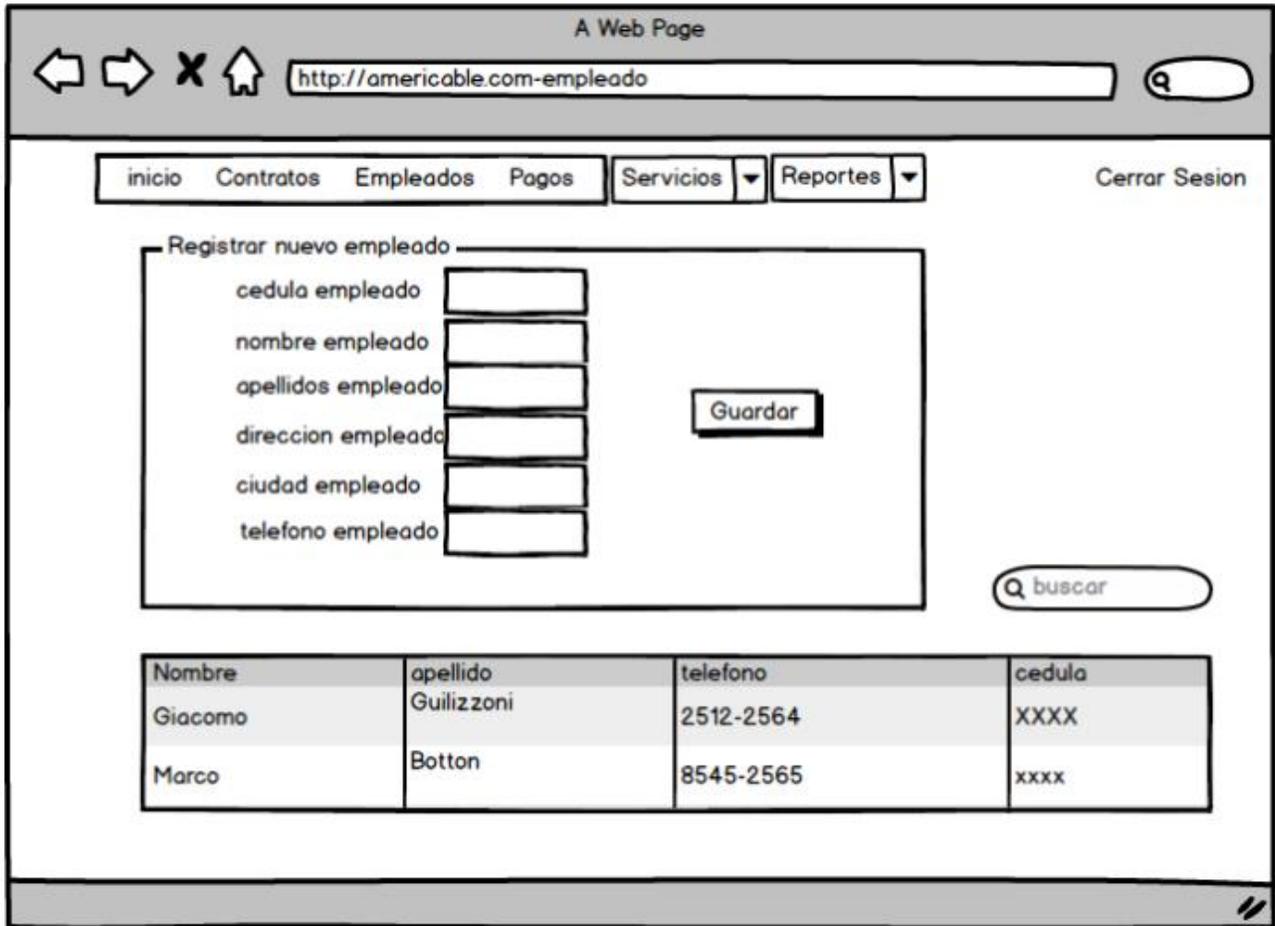


Ilustración 17. Registro de Empleados/Técnicos

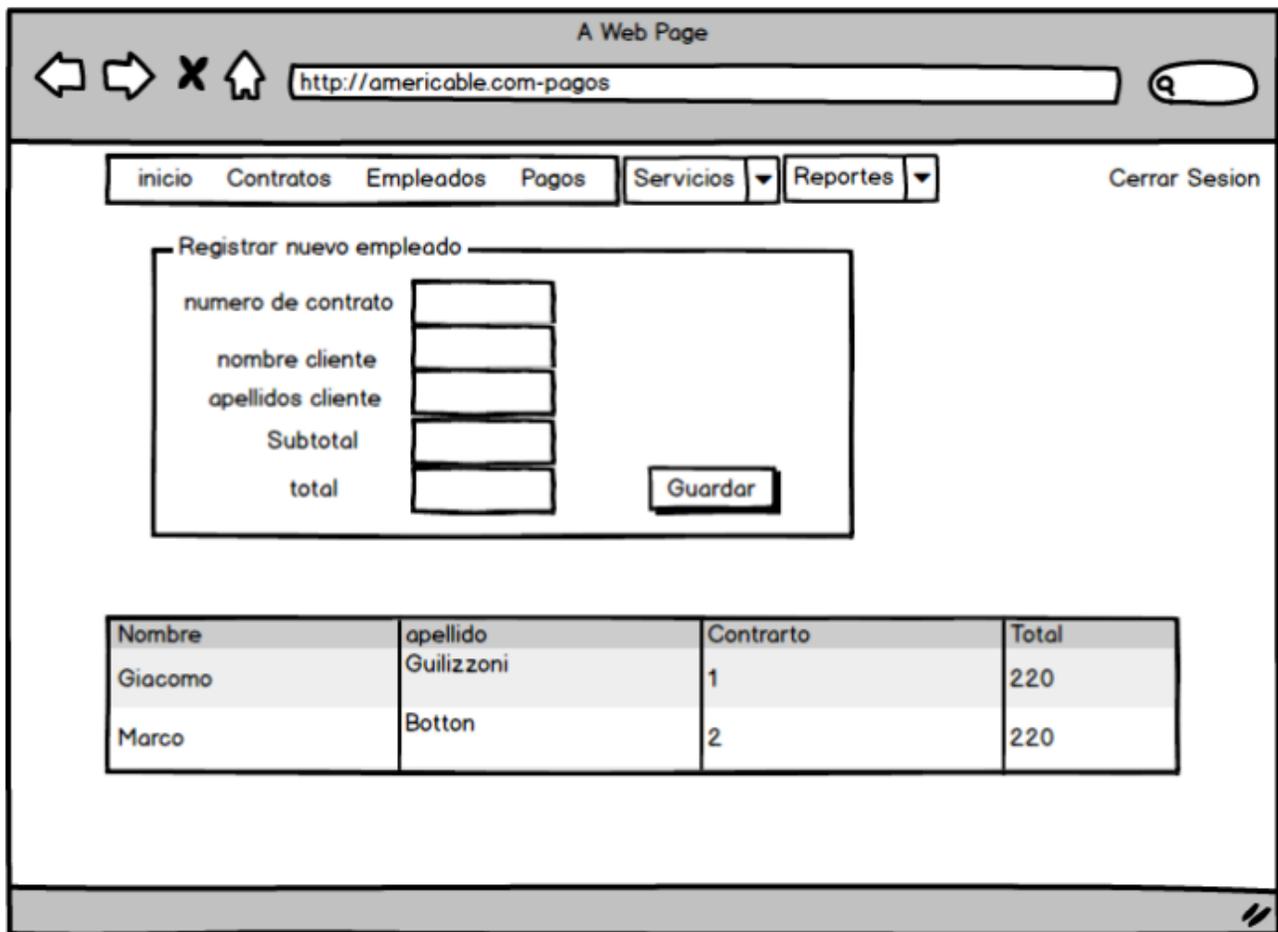


Ilustración 18. Registro de pagos

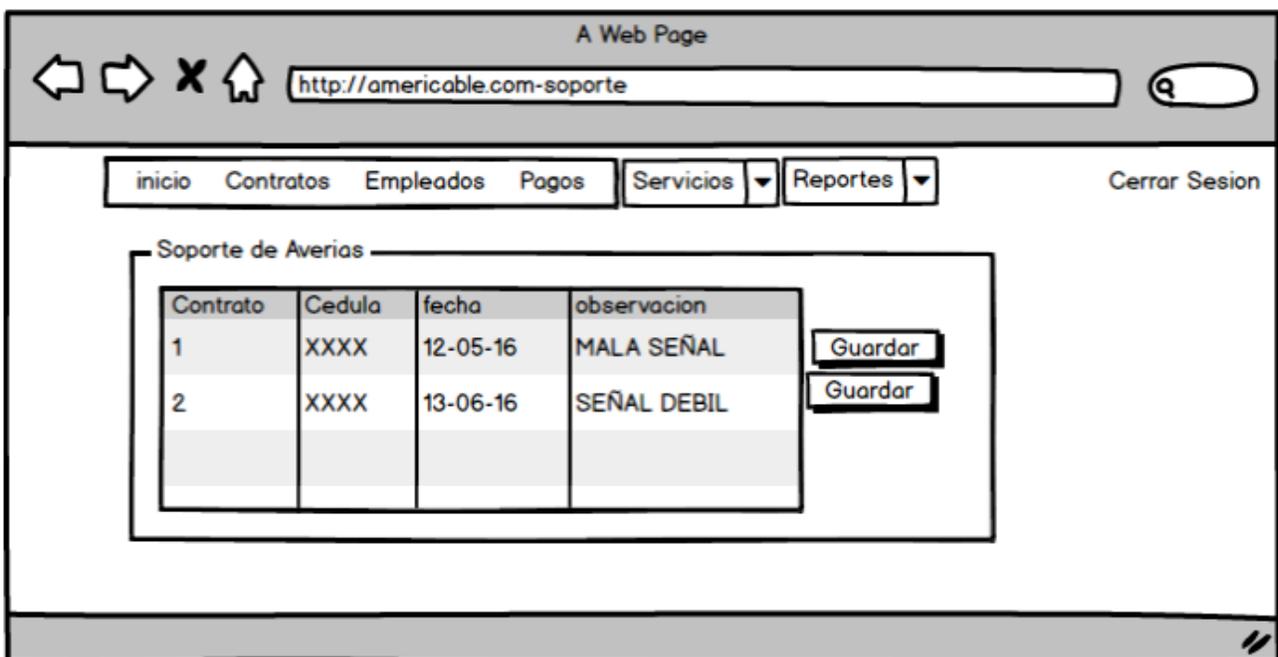


Ilustración 19. Soporte de averías/Soporte Técnico

Desarrollo, Documentación y codificación.

Anexo6. Mapa de navegación SCAA

En la siguiente figura se muestran las funciones que realiza el sistema según el tipo de usuario que ingresa al sistema:

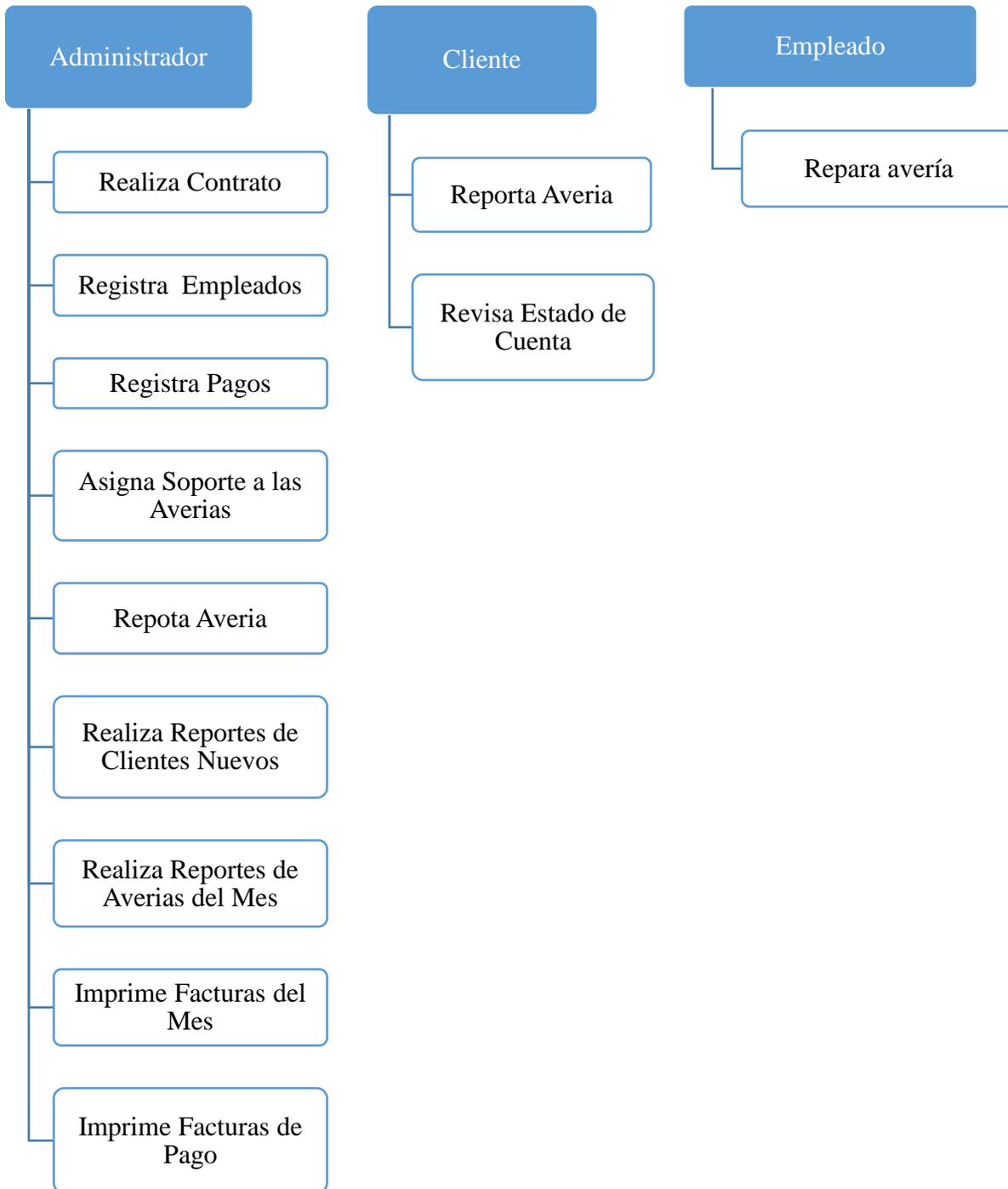


Ilustración 20. Mapa de Navegación del sistema

Anexo7. Diccionario de Datos.

tbusuario						
Campo	Tipo	longitud	PK	FK	NN	AI
cedula	Varchar	16	✓		✓	
nombres	Varchar	30			✓	
Apellidos	Varchar	20			✓	
Ciudad	Varchar	20			✓	
Teléfono	Nchar	9			✓	
Activo	Bit	-			✓	
User	Nchar	10			✓	
Password	Nchar	20			✓	
Tipo	Smallint	-			✓	
Dirección	Text				✓	

Tabla 6. Tb Usuario

tbpago						
Campo	Tipo	longitud	PK	FK	NN	AI
idpago	Int	-	✓		✓	✓
Fecha	date	-			✓	
Observación	Text	-			✓	
Mora	Float	-			✓	
Subtotal	Float	-			✓	
Total	float	-			✓	
idcontrado	int	-		✓	✓	

Tabla 7. Tb Pago

Tbcontrato						
Campo	Tipo	longitud	PK	FK	NN	AI
idcontrato	Int	-	✓		✓	✓
Observación	Text	-				
Dirección	text	-			✓	
Ciudad	Varchar	40			✓	
Monto	float	-			✓	
Activo	bit	-			✓	
Cedula	Varchar	16		✓	✓	
Descripción	Text	-			✓	
Fecha reg	datetime	-			✓	
Dirección	Text	-			✓	

Tabla 8. Tb Contrato

tbaverias						
Campo	Tipo	longitud	PK	FK	NN	AI
Idaverias	Int	-	✓		✓	✓
Observación	text	-			✓	
tipoaveria	text	-			✓	
Cedula	Varchar	16		✓	✓	
Fecha	datetime	-			✓	
idcontrato	int	-		✓	✓	

Tabla 9. Tb Averias

tbsoportes						
Campo	Tipo	longitud	PK	FK	NN	AI
Idsoportes	Int	-	✓		✓	✓
Fecha	datetime	-			✓	
Observación	text	-				
Cedula	Varchar	16		✓	✓	
Idaveria	Int	-		✓	✓	

Tabla 10. Tb Soportes

Documento de especificación de requerimientos de software IEEE 830 (SRS)

Proyecto: Desarrollo de un sistema web para la gestión del servicio de televisión por cable que ofrece la empresa Americable S.A, utilizando arquitectura distribuida de tipo cliente – servidor, en la ciudad de Juigalpa Chontales durante el segundo semestre del año 2016.

Índice

Introducción	1
Propósito.....	1
Alcance.....	1
Personal involucrado	2
Definiciones, acrónimos y abreviaturas	3
Referencias	3
Resumen	4
Descripción General	4
Perspectiva del producto	4
Funcionalidad del producto.....	4
Características de los usuarios.....	5
Restricciones	5
Suposiciones y dependencias	6
Evolución previsible del sistema.....	6
Requisitos específicos	7
Requisitos comunes de las interfaces	15
Requisitos funcionales.....	16
Requisitos No funcionales.....	19

1. Introducción

Este documento tiene como fin especificar los requisitos del software para el Sistema De Control De Abonados De Americable S.A (SCAA), describiendo cada una de las funcionalidades tales como los requerimientos funcionales y no funcionales, restricciones y herramientas de desarrollo que se aplican en el desarrollo de este, así mismo las características de los usuarios que manipularan la información y definiciones, acrónimos que contiene el documento.

Esta especificación se ha estructurado con estándar IEEE 830 que es un formato para la especificación de requisitos de software ANSI, IEEE830, 1998.

1.1. Propósito

El propósito del documento es definir cada una de las funcionalidades del sistema tales como requisitos funcionales y no funcionales para el desarrollo del sistema de información web que permitirá controlar abonados de la empresa Americable y que mediante este instrumento los desarrolladores puedan guiarse y entender de manera técnica cada uno de los procesos del sistema.

Este será utilizado por estudiantes de ingeniería en sistemas, programadores, analistas de sistema y desarrolladores web.

1.2. Alcance

La especificación de requisitos va dirigida al usuario del sistema que tiene como objetivo desarrollar sistemas de control de abonados para las empresas y para profundizar en la automatización de cada uno de los procesos de esta, la cual tiene como objetivo principal controlar la información de los servicios que brinda la empresa de manera administrativa como instalación y reparación de averías y datos de los clientes (abonados).

1.3. Personal involucrado

Nombre	Weslin Aldemaro Vargas Martínez
Rol	Programador
Categoría profesional	Si
Responsabilidades	Realización de la base de datos
Información de contacto	Barrio larga espada del monumento 3 c al lago.
Aprobación	Analista de Sistema

Nombre	Evert Josué Ortiz Morales
Rol	Programador
Categoría profesional	Si
Responsabilidades	Crear las interfaces de Comunicación
Información de contacto	Zona D De Enitel 2c. al Norte.
Aprobación	Analista de Sistema

Nombre	Ariel Emmanuel Solano Ortega
Rol	Analista de sistema
Categoría profesional	Si
Responsabilidades	Desarrollar el software
Información de contacto	Del BDF 6 c al sur.
Aprobación	Analista de Sistema

1.4. Definiciones, acrónimos y abreviaturas

Nombre	Descripción
ASP	Páginas de Servidor Activo
BD	Base de Datos
HTML	Lenguaje de Marcado de Hipertexto
IDE	Entorno de Desarrollo Integrado
IEEE	Instituto de Ingenieros Eléctricos y Electrónicos
ISO	Organización de Estándares Internacional
LAN	Red de Área Local
RF	Requerimientos Funcionales
RNF	Requerimientos No Funcionales
SGBD	Sistema Gestor De Base De Datos
SOAP	Protocolo De Acceso Simple A Objetos
XML	Lenguaje de Etiquetado Extensible

1.5. Referencias

Referencia	Título	Ruta	Fecha	Autor
SRS (IEEE 830)	Requerimientos_Norma_ieee830	www.dis.unal.edu.co	26-03-2009	José Luis
Ingeniería de software	Un enfoque práctico.	www.freelibros.org/	2010	Roger S. Pressman

1.6.Resumen

El presente documento se divide en tres secciones. En la primera sección se detalla una introducción al mismo describiendo de manera clara la especificación de los recursos del sistema.

En la segunda sección del documento se describe de manera general el sistema, cada una de las funciones que debe realizar, los datos asociados a estas, restricciones, supuestos y dependencias que involucran el desarrollo de este sin mencionar tantos detalles.

En la tercera y última sección se detallan las funcionalidades del sistema, los requisitos que debe satisfacer el sistema, tales como los requisitos de hardware y software para la implementación de este.

2. Descripción General

2.1.Perspectiva del producto

Este sistema es totalmente independiente de otro ya que fue creado para el control de abonados de Americable, este trata de ahorrar tiempo al cliente y así mismo él pueda realizar sus solicitudes desde donde quiera que se encuentre a través de internet, teniendo fácil acceso y sin límites geográficos.

2.2.Funcionalidad del producto

El sistema funcionara de manera intuitiva y sencilla, de modo que el usuario cliente tan solo requiere tener acceso a internet para entrar al sitio web, deberá contratar el servicio para obtener su usuario dentro del sistema y de esta manera podrá reportar una avería y verificar su estado de cuenta. El SCAA no permitirá realizar pagos en línea y se tendrá acceso haciendo uso de navegadores web desde cualquier dispositivo electrónico, Desde el punto de vista del administrador del sistema, realizara registros de los contratos directamente al sistema y así mismo controla los pagos del servicio y gestiona reportes de averias.

2.3.Características de los usuarios

Tipo de usuario	Administrador
Formación	TSU en informática.
Habilidades	Programación web, bases de datos, soporte técnico, redes, soporte técnico del equipo de cómputo.
Actividades	Registro de contrato de clientes, registro de empleo, reporte de avería, visualización de averías, impresión de facturas, búsqueda de clientes.

Tipo de usuario	Empleado
Formación	Conocimientos básicos de manejo de software
Habilidades	Soporte técnico a sistemas operativos, redes locales y equipo de cómputo.
Actividades	Control y manejo del sistema.

Tipo de usuario	Cliente
Formación	Niveles Básicos de Navegación Web
Habilidades	Manejar una computadora con conexión web
Actividades	Reportar averías y visualización de estado de cuenta

2.4.Restricciones

- Interfaz diseñada para ser utilizada a través de internet con los siguientes navegadores:
 - Google Chrome
 - Mozilla Firefox
 - Internet Explorer
- Los servicios deben de ser capaces de atender consultas de manera concurrente.
- Capacidad de almacenar hasta 10,000 clientes en la base de datos.
- Realizado usando el modelo Cliente-Servidor.
- Diseño e implementación sencilla independiente de la plataforma o del lenguaje de programación.
- Sistema operativo de Windows 7 o superior, Linux.

- Computadora con Memoria RAM de 2GB o más, procesador Dual Core o con mayor capacidad, Disco duro mayor o igual a 500GB.

2.5.Suposiciones y dependencias

- 2.5.1.** Visual Studio es un conjunto completo de herramientas de desarrollo para la generación de aplicaciones web ASP.NET, Servicios Web XML, aplicaciones de escritorio y aplicaciones móviles.
- 2.5.2.** C# (pronunciado si Sharp en inglés) es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma. NET, que después fue aprobado como un estándar por la ECMA (ECMA-334) e ISO (ISO/IEC 23270)
- 2.5.3.** SQL (Structured Query Lenguaje) es un lenguaje de programación diseñado para almacenar, manipular y recuperar datos almacenados en bases de datos relacionales. La primera encarnación de SQL apareció en 1974, cuando un grupo de IBM desarrolló el primer prototipo de una base de datos relacional.
- 2.5.4.** Bootstrap es un framework creado por Twitter que incluye varias herramientas de software libre para el diseño y desarrollo web.
- 2.5.5.** CSS (Cascading Style Sheets, u Hojas de Estilo en Cascada) es la tecnología desarrollada por el World Wide Web Consortium (W3C) con el fin de separar la estructura de la presentación.
- 2.5.6.** HTML5 es la última versión de HTML. El término representa dos conceptos diferentes: Se trata de una nueva versión de HTML, con nuevos elementos, atributos y comportamientos. Contiene un conjunto más amplio de tecnologías que permite a los sitios Web y a las aplicaciones ser más diversas y de gran alcance.
- 2.5.7.** Navegador de Internet (Google Chrome, Mozilla Firefox, Internet Explorer, Safari)
- 2.5.8.** El sistema requiere conexión a internet.
- 2.5.9.**

2.6.Evolución previsible del sistema

Es posible implementar el sistema de manera responsiva para que los clientes puedan realizar sus solicitudes de manera accesible desde dispositivos móviles.

3. Requisitos específicos

Requerimientos Funcionales

Número de requisito	RF01
Nombre de requisito	Autenticación de usuario
Descripción	Los usuarios deben identificarse para acceder a los módulos que les corresponden de acuerdo a su nivel de accesibilidad.
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Tabla en la base de datos
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RF02
Nombre de requisito	Registro de usuarios Empleados
Descripción	Los Clientes y Empleados deberán estar registrados para tener acceso al sistema. El sistema permitirá al administrador registrar al Empleado o técnico proporcionando datos como: Nombre, Apellido, Dirección, N° de Cedula, correo electrónico.
	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Empresa Americable.
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RF03
Nombre de requisito	Estado de cuenta.
Descripción	El cliente tendrá acceso a visualizar su estado de cuenta. El sistema proporciona al cliente una tabla para visualizar los meses del año que ha pagado el servicio y meses pendientes.
	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Empresa Americable.
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RF04
Nombre de requisito	Informe de mora al cliente.
Descripción	Una vez que el cliente entre en mora el sistema enviara automáticamente un informe de facturas pendientes impreso en el recibo del mes actual.
	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Empresa Americable.
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RF05
Nombre de requisito	Visualizar moras de los clientes.
Descripción	El usuario administrador podrá verificar que clientes están en mora en caso de solicitarlo de manera personal.
	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Empresa Americable.
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RF06
Nombre de requisito	Cambiar Nombre de usuario y Contraseña.
Descripción	Todos los usuarios tendrán acceso a cambiar el nombre de usuario y contraseña cuando lo deseen.
	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Empresa Americable.
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RF07
Nombre de requisito	Registrar contrato.
Descripción	El cliente deberá solicitar al administrador el contrato del servicio. El proceso de registro del cliente se dará únicamente en caso de realizar el contrato del servicio. El sistema permitirá al administrador realizar el contrato llenando los campos correspondientes al registro del cliente y datos del contrato.
	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Empresa Americable.
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RF08
Nombre de requisito	Búsquedas
Descripción	El sistema permitirá al administrador realizar búsquedas por cliente. Estas búsquedas son realizadas por cada uno de los campos de las tablas en la base de datos.
	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Empresa Americable.
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RF09
Nombre de requisito	Reparación de averías
Descripción	Gestión de solicitud de reparación de averías: El sistema proporcionara la opción de reportar averías para que el cliente brinde los datos correspondientes para asistir a la solicitud.
	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Empresa Americable.
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RF10
Nombre de requisito	Eliminar usuario(cliente, empleado)
Descripción	El sistema permitirá al usuario administrador eliminar los datos de los clientes que dan de baja a los servicios o que no cumplen con las cláusulas del contrato. El sistema permitirá al administrador eliminar a los usuarios técnicos que no laboran en la empresa. El sistema permitirá al administrador eliminar las averías que han sido asistidas y corregidas por el empleado técnico de la empresa.
	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Empresa Americable.
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RF11
Nombre de requisito	Gestionar Reportes
Descripción	<p>Permite al administrador imprimir reportes de los pagos de los clientes.</p> <p>El sistema Generara automáticamente reportes de los contratos realizados cada mes, de esta misma manera generara reportes de los contratos cancelados.</p> <p>Las facturas de los clientes serán generadas e impresas cada mes de manera automática.</p> <p>Las averías realizadas mensualmente por los clientes serán reportadas automáticamente.</p>
	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Empresa Americable.
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RF12
Nombre de requisito	Asignación de Averías
Descripción	<p>Permite al administrador asignar empleados técnicos que asistan a las solicitudes de averías de los clientes enviando un reporte a sus cuentas en el sistema.</p>
	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Empresa Americable.
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RF13
Nombre de requisito	Generar Recibo de pago.
Descripción	Gestión de pago del servicio: El sistema permite al administrador registrar el pago del cliente e imprimir recibo.
	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Empresa Americable.
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RF14
Nombre de requisito	Estado de Averías reportadas
Descripción	Visualización de averías al técnico: El Técnico tendrá acceso a visualizar las averías a dar solución asignadas por el administrador. Una vez solucionada la avería deberá reportar al sistema.
	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Empresa Americable.
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RF15
Nombre de requisito	Modificar datos de los usuarios
Descripción	El sistema permite al administrador editar datos erróneos en los registros realizados así como registro de datos personales de los usuarios en caso de ser necesario.
	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Empresa Americable.
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Requerimientos No Funcionales

Número de requisito	RNF01
Nombre de requisito	Interfaz del sistema.
Descripción	El sistema tendrá una interfaz de uso sencilla e intuitiva.
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Analista del Sistema.
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RNF02
Nombre de requisito	Ayuda
Descripción	La interfaz del sistema deberá proporcionar un icono de ayuda que muestre el manual de usuario para facilitar el trabajo en cuanto al manejo.
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Analista del Sistema.
Prioridad del requisito	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> Baja/ Opcional Alta/Eencial Media/Deseado

Número de requisito	RNF03
Nombre de requisito	Fiabilidad
Descripción	El sistema realiza las operaciones lógicas de manera rápida a las solicitudes de los clientes.
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Analista del Sistema.
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RNF04
Nombre de requisito	Desempeño
Descripción	El sistema deberá permitir que la información almacenada o registros realizados puedan ser consultados y actualizados permanentemente y de manera concurrente sin que afecte el tiempo de respuesta.
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Analista del Sistema.
Prioridad del requisito	<input checked="" type="checkbox"/> <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional Alta/Esencial

Número de requisito	RNF05
Nombre de requisito	Confiabilidad continua del sistema
Descripción	El sistema debe estar disponible siempre, funcionar las 24 horas del día los 7 días de la semana permitiendo la carga de datos.
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Analista del Sistema.
Prioridad del requisito	<input checked="" type="checkbox"/> <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional Alta/Esencial

Número de requisito	RNF06
Nombre de requisito	Portabilidad en el acceso.
Descripción	Los distintos usuarios podrán acceder al sistema desde cualquier ordenador, dispositivo móvil sin importar el sistema operativo o plataforma que se esté utilizando; realizar sus consultas y solicitudes siempre y cuando tenga acceso a internet.
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Analista del Sistema.
Prioridad del requisito	<input checked="" type="checkbox"/> <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Número de requisito	RNF07
Nombre de requisito	Seguridad
Descripción	El sistema garantiza a los usuarios seguridad como uso de variable de sesión en cuanto a la información y datos que se controlan y gestionan así como las contraseñas. Un nivel adecuado de seguridad proporcionado por entity framework en cuanto a la base de datos para evitar posibles ataques.
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Analista del Sistema.
Prioridad del requisito	<input checked="" type="checkbox"/> <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional Alta/Esencial

Número de requisito	RNF08
Nombre de requisito	Funcionalidad operable.
Descripción	El sistema realizara las funciones de manera correcta y aceptable permitiendo suprimir errores en la transición de datos.
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Analista del Sistema.
Prioridad del requisito	<input checked="" type="checkbox"/> <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional Alta/Esencial

3.1. Requisitos comunes de los interfaces

Los requisitos comunes de las interfaces describen las entradas y salidas del sistema, de manera que el sistema posee distintas interfaces para su funcionamiento.

3.1.1. Interfaces de usuario

La interfaz con el usuario consistirá en un menú que lleva como elementos las distintas ventanas; botones, tablas, listas y campos de textos. Dentro de las ventanas existen: Solicitud de un segundo TV, solicitud de averías, Estado de cuentas. Esta interfaz deberá ser diseñada de acuerdo a las especificaciones propuestas y será visualizada a través de navegadores de internet.

3.1.2. Interfaces de hardware

Sera necesario tener equipos de cómputos en perfecto estado cumpliendo con las siguientes características: Los requisitos recomendados para la navegación del sistema en el hardware son no menores de 2 GB de RAM, con un disco duro de 500GB de memoria al menos, un mouse, un teclado para la entrada de datos y adaptadores de red.

3.1.3. Interfaces de software

Es de vital importancia el uso de otros productos de software independientes pero que integran el sistema. Es necesario utilizar:

- Sistema Operativo Windows XP, 7 o superior. El propósito de este es organizar y administrar las funciones del ordenador y establecer la comunicación ordenador-cliente o usuario.
- Explorador: Google Chrome, Mozilla Firefox o Internet Explorer en las versiones antes mencionadas. Se hace uso de un explorador para acceder al sistema ya que es un sistema que opera en la web.
- SGBD SQL server 2012 para gestionar la información en la base de datos.

3.1.4. Interfaces de comunicación

El sistema es independiente de otros sistemas. Existe la comunicación entre las capas del sistema, los servidores, clientes y aplicaciones se comunican entre sí, utilizando protocolos TCP/IP en internet, haciendo uso de socket, y otros servicios de tipo SOAP que permiten la iteración entre las distintas capas.

3.2.Requisitos funcionales

3.2.1. Requisito funcional 1

Autenticación de usuario: Los usuarios deben identificarse para acceder a los módulos que les corresponden de acuerdo a su nivel de accesibilidad.

3.2.2. Requisito Funcional 2

Registro de usuarios Empleados: Los Clientes y Empleados deberán estar registrados para tener acceso al sistema.

El sistema permitirá al administrador registrar al Empleado o técnico proporcionando datos como: Nombre, Apellido, Dirección, N° de Cedula, correo electrónico.

3.2.3. Requisito Funcional 3

Estado de cuenta: El cliente tendrá acceso a visualizar su estado de cuenta.

El sistema proporciona al cliente una tabla para visualizar los meses del año que ha pagado el servicio y meses pendientes.

3.2.4. Requisito Funcional 4

Informe de Mora al cliente: Una vez que el cliente entre en mora el sistema enviara automáticamente un informe de facturas pendientes impreso en el recibo del mes actual.

3.2.5. Requisito Funcional 5

Visualizar moras de los clientes: El usuario administrador podrá verificar que clientes están en mora en caso de solicitarlo de manera personal.

3.2.6. Requisito Funcional 6

Cambiar Nombre de Usuario y Contraseña: Todos los usuarios tendrán acceso a cambiar el nombre de usuario y contraseña cuando lo deseen.

3.2.7. Requisito Funcional 7

Registrar Contrato: El cliente deberá solicitar al administrador el contrato del servicio. El proceso de registro del cliente se dará únicamente en caso de realizar el contrato del servicio. El sistema permitirá al administrador realizar el contrato llenando los campos correspondientes al registro del cliente y datos del contrato.

3.2.8. Requisito Funcional 8

Búsquedas: El sistema permitirá al administrador realizar búsquedas por cliente. Estas búsquedas son realizadas por cada uno de los campos de las tablas en la base de datos.

3.2.9. Requisito Funcional 9

Reparación de averías: Gestión de solicitud de reparación de averías: El sistema proporcionara la opción de reportar averías para que el cliente brinde los datos correspondientes para asistir a la solicitud.

3.2.10. Requisito Funcional 10

Eliminar:

- El sistema permitirá al usuario administrador eliminar los datos de los clientes que dan de baja a los servicios o que no cumplen con las cláusulas del contrato.
- El sistema permitirá al administrador eliminar a los usuarios técnicos que no laboran en la empresa.
- El sistema permitirá al administrador eliminar las averías que han sido asistidas y corregidas por el empleado técnico de la empresa.

3.2.11. Requisito Funcional 11

Gestionar Reportes:

- Permite al administrador imprimir reportes de los pagos de los clientes.
- El sistema Generara automáticamente reportes de los contratos realizados cada mes, de esta misma manera generara reportes de los contratos cancelados.
- Las facturas de los clientes serán generadas e impresas cada mes de manera automática.
- Las averías realizadas mensualmente por los clientes serán reportadas automáticamente.

3.2.12. Requisito Funcional 12

Asignación de Averías: Permite al administrador asignar empleados técnicos que asistan a las solicitudes de averías de los clientes enviando un reporte a sus cuentas en el sistema

3.2.13. Requisito Funcional 13

Generar Recibo de pago.

Gestión de pago del servicio: El sistema permite al administrador registrar el pago del cliente e imprimir recibo.

3.2.14. Requisito Funcional 14

Estado de Averías Reportadas

Visualización de averías al técnico: El Técnico tendrá acceso a visualizar las averías a dar solución asignadas por el administrador. Una vez solucionada la avería deberá reportar al sistema.

3.2.15. Requisito Funcional 15

Modificar: El sistema permite al administrador editar datos erróneos en los registros realizados así como registro de datos personales de los usuarios en caso de ser necesario.

3.3. Requisitos No funcionales

3.3.1. Requisitos de rendimiento

Garantizar que el diseño de las consultas y otros procesos no afecten la conexión con múltiples usuarios que soliciten el servicio. Siendo capaz de procesar el 75% de las transacciones cada 3 segundos ya que su tiempo de respuesta es de forma rápida y segura para guardar la información, teniendo en cuenta que la información que puede guardarse no debe sobrepasar los límites requeridos. Hay que tener en cuenta que la velocidad depende de la red y de las terminales en que se está ejecutando.

3.3.2. Seguridad

Evitar medidas contra ataques DDOS (Ataque de negación de servicios) habilitando opciones de logs para llevar un control adecuado de las conexiones que existen con dichos router, uso correcto de variables de sesión evitando usar variables superglobal, se hará uso de SSL (Secure Socket Layer), y asignaciones de roles distintos para el usuario administrador y el usuario final o cliente mediante la autenticación haciendo uso de nombre de usuario y contraseña. Dado que su funcionalidad depende de la iteración de las capas del sistema y que se utilizaran servicios, el diseño UI trabajara de forma remota permitiendo la integridad de la base de datos. También se evitan infiltraciones en la base de datos mediante la autenticación de usuario en SQL server.

3.3.3. Fiabilidad

Las funciones que ejecuta el sistema las realiza de forma correcta bajo condiciones operativas conforme a los requerimientos funcionales y no funcionales, de modo que cada una de las capas del sistemas interactúan exitosamente entre si y cada uno de sus procesos son altamente funcionales evitando errores.

3.3.4. Disponibilidad

Se pretende que el sistema funcione las 24 horas del día, el sistema estaría disponible en un 100% de tiempo de servicio para los usuarios finales siempre y cuando tenga acceso a internet.

3.3.5. Mantenibilidad

El mantenimiento se clasificara en cuatro tipos de actividades:

- Adaptable – modificar el sistema para hacer frente a cambios en el ambiente del software (SGBD, OS)
- Perfectivo – implementar nuevos, o cambiar requerimientos de usuario referentes a mejoras funcionales para el software
- Correctivo, diagnosticar y corregir errores, posiblemente los encontrados por los usuarios
- Preventiva – aumentar la capacidad de mantenimiento de software o fiabilidad para evitar problemas en el futuro

Estas actividades se aplicaran para reducir los costos del ciclo de vida del software. Deben ser ejecutadas por un desarrollador de sistema especialista en el área aplicándolas cada que sea necesario. Debido a que cada una de ellas se aplica en tiempos independientes no es posible especificar el rango de tiempo.

3.3.6. Portabilidad

Los componentes del sistema son independientes del servidor ya que este es funcional en cualquier servidor u ordenador en el cual se aloje, el sistema en si puede ser utilizable en la plataforma de .Net, el código fuente puede ser reutilizable sin necesidad de volver a codificar nuevo código, es funcional en cualquier sistema operativo compatible con .Net y que posea un navegador web. Además de esto es un sistema responsivo funcional en cualquier dispositivo móvil.

PLAN DE VERIFICACIÓN Y VALIDACIÓN IEEE 730

Sistema SCAA

Índice

Historia de revisiones	1
Introducción.....	2
Propósito.....	3
Punto de partida	4
Alcance	5
Identificación del proyecto	6
Estrategia de Evolución	6
Requerimientos para verificar	7
Estrategia de Verificación	7

1. Historia de revisiones

Fecha	Versión	Descripción	Autor
12/11/2016	1	Presentación del plan de verificación y validación de pruebas	Evert Ortiz Morales
18/01/2017	1.0	Descripción de pruebas	<ul style="list-style-type: none">• Weslin Aldemaro Vargas Martínez• Ariel Solano Ortega

2. Introducción

La validación y verificación de software es un conjunto de procedimientos, actividades, técnicas y herramientas que se utilizan, para el desarrollo y asegurar que un producto de software cumpla con los requerimientos planteados por el usuario final.

El desarrollo del sistema SCAA (Sistema para el control de abonados de Americable), está formado por un conjunto de reglas, que facilitan el desarrollo, el esfuerzo de la validación y verificación del software, para verificar que tan eficiente y seguro es el sistema.

Este plan está orientado a las áreas de proceso que tiene el sistema SCAA, tanto en el ámbito de desarrollo (Validación y Verificación) como de la de adquisición (Validación y Verificación de la adquisición).

A continuación se describen en detalle los procesos de validación y verificación de las pruebas, definiendo estrategias y tipos de pruebas, según el SCAA.

3. Propósito

Encontrar errores y defectos que puedan existir en el uso del sistema a fin de corregirlos. Verificar que los validadores de datos funcionen y limiten el ingreso de información, para que se restrinjan los datos que no estén permitidos. Las características de seguridad relacionadas con el ingreso no autorizado de usuarios, de manera que se puedan realizar modificaciones donde no sean permitidas. Con la realización de este Plan se pretenden alcanzar los siguientes objetivos:

- Identificar la información de proyecto y los componentes de software que deben ser verificados
- Detallar los requerimientos recomendados para verificar.
- Describir las estrategias de verificación que serán usadas.
- Especificar los entregables del proyecto de verificación.
- Detectar y corregir los defectos tan pronto como sea posible en el ciclo de vida del software.
- Disminuir los riesgos.
- Optimizar la calidad y fiabilidad del software.

Los objetivos que tienen la V y V. La **validación** tiene por objetivo determinar la corrección del producto final con respecto a las necesidades de los usuarios finales. La **verificación** tiene por objetivo demostrar la consistencia y corrección del sistema.

4. Punto de partida

El inicio a la verificación y validación del software, se da a partir de la creación del sistema web SCAA, realizada durante el segundo semestre del año 2016, por medio del cual se hará una verificación y validación en entorno de funcionamiento, visibilidad, entre otros al sistema.

Se tomaran a verificar y validar los módulos que contiene el sistema como usuario (registro de clientes) usuario administrador (dar de baja usuario, reportar avería, registrar empleado), se desarrolla con ayuda de los programas Visual Studio 2012 y SQL Server 2012 estos son los programas utilizados para el desarrollado del sistema web.

5. Alcance

En cuanto a la verificación del sistema web SCAA se aplicaran pruebas unitarias, pruebas de validación, pruebas de caja negra y pruebas de caja blanca, con el fin de determinar el buen funcionamiento del mismo.

Características que serán objeto de verificación:

- **Consistencia:** vigilar que la información sea coherente.
- **Precisión:** corrección de la sintaxis.
- **Compleitud:** Se han implementado las situaciones demandadas según requerimientos funcionales.

6. Identificación del proyecto

El documento que se utilizó para elaborar el Plan de Verificación y Validación es el siguiente:

- ✓ Plan de verificación y validación 1.0 del estándar IEEE 730

7. Estrategia de Evolución

El plan de verificación y validación fue realizado por el grupo de trabajo del desarrollo del sistema (SCAA).

Contiene los siguientes campos:

Responsables de monitorear el Plan de Verificación y Validación.

- Br. Evert Ortiz Morales.
- Br. Weslin Aldemaro Vargas Martínez.
- Br. Ariel Solano Ortega.

Como serán evaluados y aprobados los cambios al Plan.

Los cambios del plan se efectuarán de acuerdo a cada resultado de las pruebas que se realicen en el sistema y según los requerimientos de cada prueba que se utilice.

Como serán realizados y comunicados los cambios al Plan.

Se conocerán y se discutirán con el grupo de trabajo para sus posteriores cambios en el sistema y documento de validación-verificación.

8. **Requerimientos para verificar**

Requerimientos no funcionales a verificar:

El sistema de SCAA, puede ejecutarse y administrarse en cualquier PC, mediante el servicio de internet.

- Seguridad del Sistema
- Rendimiento del sistema
- Fiabilidad
- Portabilidad

Requerimientos funcionales a verificar:

- Roles de los Usuarios/Administrador.
- La codificación del sistema.
- Procesos que realizara el Administrador.
- Procesos que realizara el usuario.
- Base de Datos.

9. **Estrategia de Verificación**

Tipos de pruebas

Las pruebas realizadas en este documento para el sistema SCAA se basan en el marco de buenas prácticas del estándar IEEE 730 de Validación y Verificación de Software.

- **Prueba unitaria**

Descripción

Consiste en probar cada módulo independientemente uno del otro, la prueba de unist test se realizó en los procesos más importantes del sistema (registro de clientes, reportar avería,

registrar empleado) para detectar todos los errores de códigos que se presentan en las entradas, salidas y procesamiento de la información.

Objetivo de la prueba unitaria

- Evidenciar que cada módulo que conforma el SCAA, entendido como una unidad funcional, está correctamente codificado.
- Demostrar que los datos introducidos son iguales a los almacenados en la base de datos.

Técnica

Verificar la correcta codificación de los procesos de SCAA.

Realizar revisiones a las ventanas que serán de más uso para el usuario

Resultados obtenidos

Los resultados arrojados en esta prueba se muestran en las tablas siguientes:

Parámetros Evaluados

Modulo	Funcionamiento	Modulo funcional/No funcional	Conexión con la bd	Datos arrojados por el sistema	Fallas encontradas en el modulo	Fallas reencontradas en la bd	Resultados de la prueba	Fecha de ejecución
Registro de empleados	El módulo de empleados se encuentra funcionando de acuerdo a los requerimientos establecidos.	Total-funcional	La conexión del módulo con la base de datos se encuentra correcta en los datos de entrada y salida.	Los datos introducidos en el formulario son exactos a los almacenados en la base de datos.	No se encontraron fallas en el módulo	No se encontraron fallas en los campos que componen la tabla empleado.	La prueba unitaria en el módulo de empleados fue exitosa, ya que no se encontraron defectos de código.	18-01-2017
Registro de contrato	El módulo de registro de contrato se encuentra funcionando de acuerdo a lo acordado en los requerimientos.	Funcional	Se encuentra con buena conexión a la base de datos.	Los datos llenados en el formulario son los mismos datos almacenados en la base de datos.	No se encontraron fallas	No se encontraron fallas	Exitosa	19-01-2017
Asignación de averías	Este módulo se encuentra funcionando correctamente con lo establecido en los requerimientos	Funcional	Buena conexión	Los datos ingresados son los mismos almacenados en la bd	No hay fallas	No hay fallas	No se encontraron errores	20-01-2017
Autenticación de usuario/ Administrador	El modulo está en buen funcionamiento	Funcional	Buena conexión	Los campos están validados y con restricción	No se encontraron fallas	No presenta defectos o fallas	Exitosa la prueba ya que todo funciona correctamente	20-01-2017
Editar cuentas	Buen funcionamiento de este modulo	Funcional	Buena conexión	Campos actualizados en la bd	Cero fallas	No hay defectos	Exitosa	21-01-2017
Realizar búsquedas	Funciona correctamente	Funcional	Buena conexión	Muestra datos almacenados	No hay fallas	Cero fallas	Exitosa	21-01-2017
Eliminar cuentas	El modulo dar de baja funciona correctamente	Funcional	Conexión buena	Se actualizan los datos en la bd	Cero fallas	Cero fallas	Exitosa	22-01-2017

Verificación de prueba unitaria

(Modulo registro de empleado)

Datos	Verificación			Observación
	Si	No	No aplica	
Guarda valores duplicados		X		Ninguna
Existen diálogos de guardado de información		X		Ninguna
Guarda datos ingresados	X			Ninguna
Existen alertas de campos no llenados	X			Ninguna
Los datos ingresados coinciden con los almacenados en la BD	X			Ninguna
El módulo de empleados es independiente de otro modulo	X			Ninguna
Los campos del módulo empleados se encuentran validados	X			Ninguna

Se hicieron 7 interrogantes de prueba unitaria en checklist, las que resultaron 2 de manera negativa y 5 positivas.

Prueba de Validación

Descripción de la prueba de validación

En esta prueba de validación se debe asegurar que el software cumple las expectativas definidas por el cliente.

La validación va más allá de comprobar si el sistema está acorde con su especificación, para probar que el software hace lo que el usuario final espera a diferencia de lo que se ha especificado.

Esta prueba se realizó basándose en el marco de buenas prácticas IEEE 730, aplicadas al módulo de Registro de contrato del sistema SCAA.

Objetivo de la prueba

Comprobar que SCAA cumple con los requerimientos establecidos.

Técnica

Verificar que cada campo del formulario de contrato se encuentre de acuerdo a los campos establecidos en la base de datos.

Resultado de la prueba de validación:

Lista de verificación del módulo registro de contrato

Datos de verificación	Verificación			Observaciones
	Si	No	No aplica	
Permite guardar datos duplicados		X		
Almacena los datos de llenado	X			
Envía diálogos erróneos de llenado	X			
Verifica la longitud de los datos en cada campo del formulario		X		
El formulario cumple con los requisitos funcionales	X			
Los datos de entrada y salida son correctos	X			
Los campos del formulario se encuentran validados para el usuario	X			

Este módulo cuenta con 7 interrogantes para validar el registro de contrato resultando 2 negativas y 5 positivas.

Pruebas de Caja Negra

En que consiste la prueba de Caja Negra

- Se centran en los requisitos funcionales del software.
- Comprueba que la funcionalidad del programa o sistema es completamente operativa.
- Las entradas y las salidas sean correctas.
- Verifica que la integridad de la información interna se mantenga.

Esta prueba se aplicó a los módulos de validación de usuarios, registro de empleados y estados de cuenta. Donde se comprobó la funcionalidad de los módulos, para verificar los datos de entrada y salida del SCAA.

Objetivo de la prueba

- Detectar funciones incorrectas o ausentes.
- Detectar errores en los procesos e interfaz del sistema.
- Eliminar los errores de estructura de datos o acceso a BD externas.
- Corregir errores de rendimiento.

Técnica

- Tabla de condiciones

Resultados de la prueba

Test ID	Descripción	Pasos	Resultados esperados	Status
T1 Validación de usuario	Probar el botón “Ingresar” con entradas validas	Introducir valores validos como: <ul style="list-style-type: none"> • usuario • contraseña 	La información se valida correctamente	Pass ”pasar”
T2 Validación de usuario	Probar el botón “Ingresar” con entradas validas	Introducir datos no validos: <ul style="list-style-type: none"> • usuario • contraseña 	Los datos no se guardan porque no coinciden con los almacenados en la BD	Fail “fallar”

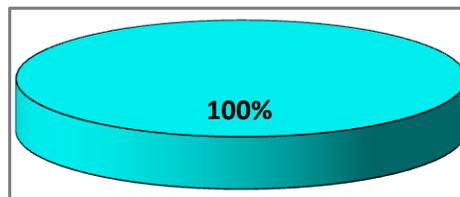
Test ID	Descripción	Pasos	Resultados esperados	Status
T3 Registro de Empleados	Botón “Guardar” con entradas validas	Introducir datos validos del empleado: <ul style="list-style-type: none"> • Nombre • Cedula 	Los datos son guardados correctamente	Pass ”pasar”
T4 Registro de Empleados	Botón “Guardar” con entradas validas	Introducir datos no validos del empleado: <ul style="list-style-type: none"> • Nombre • Cedula 	Los datos no se guardan debido a que el sistema pide datos validados	Fail ”fallar”

Test ID	Descripción	Pasos	Resultados esperados	Status
T5 Estado de la cuenta.	Botón “buscar” para visualizar la información de cliente	Se introducen datos validos: <ul style="list-style-type: none"> • Nombre • Numero de contrato 	Los datos se guardan y se visualiza el estado de cuenta del cliente	Pass “pasar”
T6 Estado de la cuenta.	Botón “buscar” para visualizar la información de cliente	Se introducen datos inválidos: <ul style="list-style-type: none"> • Nombre • Numero de contrato 	El sistema no muestra nada, porque solo clientes que estén registrados son los que tendrán un estado de cuenta	Fail “fallar”

Test ID	Descripción	Pasos	Resultados esperados	Status
T7 Informe de mora	Una vez que el cliente entre en mora el sistema enviara automáticamente un informe de facturas pendientes impreso en el recibo del mes actual	Ingresar datos validos del cliente datos personales: <ul style="list-style-type: none"> • Nombre • Dirección • Teléfono 	Solo el cliente podrá ver las facturas pendientes impreso en el recibo del mes actual	Pass “pasar”
T8 Informe de mora	Cuando el cliente se encuentre en mora el sistema enviara un informe de facturas pendientes impreso en el recibo del mes actual	Digite datos no validos del cliente: <ul style="list-style-type: none"> • Nombre • Dirección • Teléfono 	El sistema no puede generar ningún informe ya que los datos del cliente tienen que ser válidos y deben estar almacenados en la BD del sistema	Fail “fallar”

Test ID	Descripción	Pasos	Resultados esperados	Status
T9 Registrar un cliente	Botón “Guardar” con datos validos del cliente	Introducir datos validos: “Cédula”, “Nombres”, “Apellidos”, “dirección”	Los datos han sido guardados exitosamente	Pass “pasar”
T10 Registrar un cliente	Botón “Guardar” con datos inválidos del cliente	Introducir datos no validos: “Cédula”, “Nombres”, “Apellidos”, “dirección”	El sistema no le permite guardar datos que no estén validados	Fail “fallar”

Se realizaron 10 pruebas de caja negra utilizando la técnica de condiciones: resultando 10 (pass) siendo exitosos los resultados obtenidos en esta prueba comprobando que la funcionalidad del SCAA.



■ Pruebas de caja negra SCAA

Pruebas de caja blanca

En que consiste la prueba de caja blanca

Se centra en los detalles del software, por lo que su diseño está fuertemente ligado al código fuente. El testeador escoge distintos valores de entrada para examinar cada uno de los posibles flujos de ejecución del programa y cerciorarse de que se devuelven los valores de salida adecuados.

Esta prueba se aplicó al “Login” del sistema SCAA, donde se comprobará que los usuarios que están establecidos son los únicos que tendrán acceso a los módulos del sistema según corresponda a cada uno.

Objetivos de la prueba

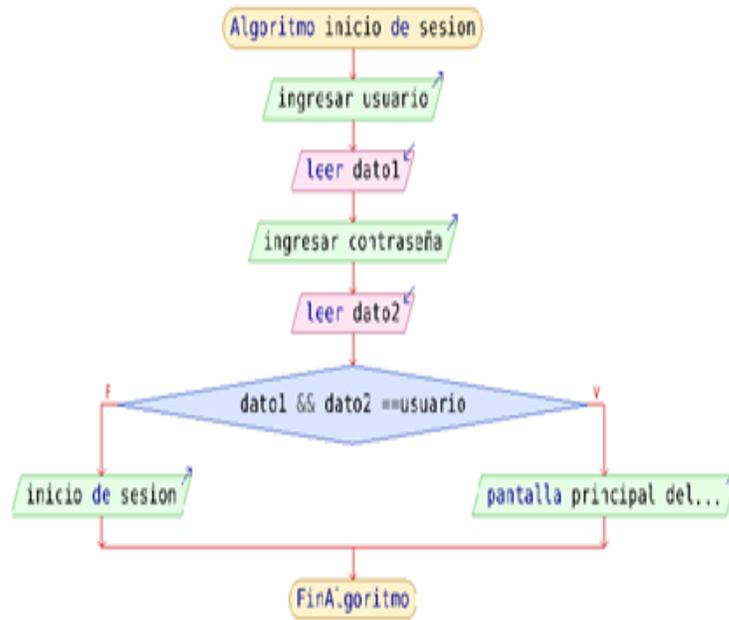
- Comprobar los flujos de ejecución dentro de cada unidad.
- Ejercitar las decisiones lógicas verdaderas y falsas

Técnica

Responsabilidad de decisión y condición, requiere que cada condición de cada división se evalúe cuando es verdadera y cuando es falsa al menos una vez, y que cada decisión se evalúe cuando es falsa y cuando es verdadera.

Resultados de la prueba

Datos de entrada	Procesos	Salida
Validación de usuario	<ul style="list-style-type: none">• Abrir BD• Leer la consulta (select * from tipo usuario) para verificar si coinciden con los datos que están registrados.	Ingresar a la sección a la cual corresponde el tipo de usuario.



La prueba de caja blanca consistió en la técnica de cobertura decisión y condición realizada con la ayuda de un diagrama de flujo que permitió ver los diferentes caminos que se pueden presentar en el inicio de sesión del sistema. Siendo ejecutada satisfactoriamente.

Anexo10. Manual Técnico

Manual técnico del Sistema De Control De Abonados Americable S.A.

Índice

Introducción	1
Creando base de datos	2
Creando Procesos Servidor	3
Creando Procesos Cliente.....	13

a) Introducción

En el siguiente manual se explica los procesos y métodos con los cuales ha sido elaborado **SCAA (Sistema de Control de Abonados de Americable S.A)**.

En el manual se puede encontrar los procesos que se realizaron para la creación del servidor utilizando servicios web, con cada uno de los web métodos en donde se levantan los servicios para luego ser llamados por el cliente y así usar los servicios web.

SCAA está realizado en .Net con C# 2012 y SQL Server 2012 como herramientas de programación y base de datos, además se utiliza Bootstrap como herramienta de diseño en la parte del cliente para darle colorido y diseño.

Se realiza bajo el esquema ENTITY FRAMEWORK que proporciona .net para sistemas orientados a objetos, ya que proporciona mayor seguridad y es más fácil para posteriores actualizaciones, además de ser una de las plataformas más actuales que se utilizan para crear sistemas web.

b) Creando base de datos

La base de datos está creada en SQL Server 2012 consta de 5 tablas:

1. **Tbpagos**
2. **Tbusuario**
3. **Tbcontrato**
4. **Tbaverias**
5. **Tbsoportes**

Tbpagos en esta tabla se almacenan los pagos del servicio realizados por los clientes la cual contiene una fecha que el día que se realiza el pago del servicio, observación si la hay, mora en caso de deber más de un mes del servicio, un subtotal y un total de lo adeudado por el cliente.

Tbusuario en esta tabla se almacenan los tres tipos de usuario los cuales son: administrador, empleado y cliente, se almacenan en la misma tabla, pero se diferencia por medio del tipo de usuario (0,1,2).

Tbcontrato es la tabla contenedora del contrato del servicio realizado por el cliente.

Tbaverias es la tabla en la cual se almacenan las averías reportadas por los clientes.

Tbsoportes en la tabla se almacenan las averías que han sido reparadas.

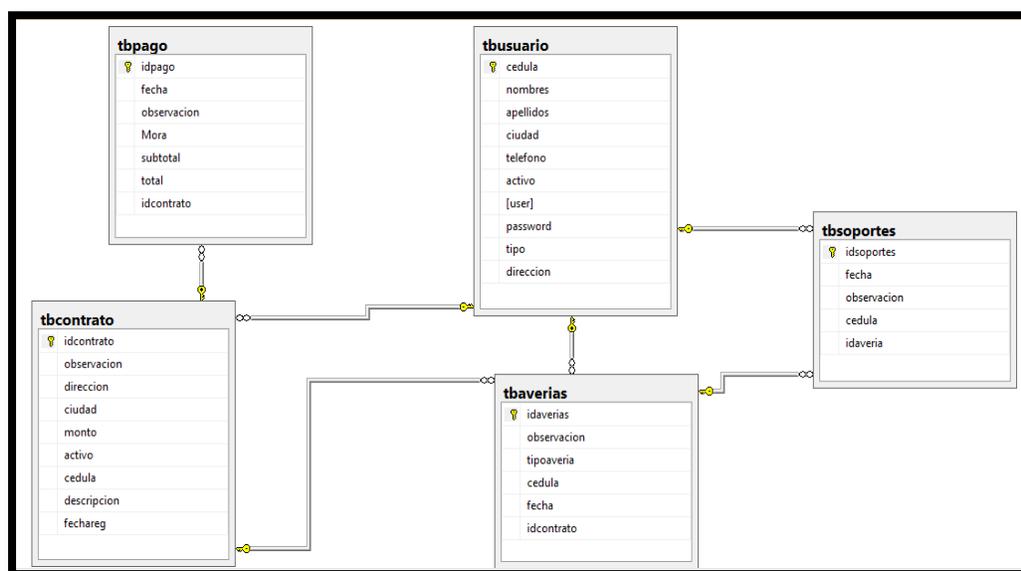


Fig.1

En la figura 1 podemos observar la base de datos con sus debidas relaciones.

c) Creando Procesos Servidor

Para realizar el servidor primeramente se realizó el archivo AmericableBD.edmx o mapeo como también se le conoce, el cual contiene los elementos CRUD (**Crear, Leer, Actualizar, Borrar**) en la base de datos, en la Fig. 2 se observa el objeto creado en C# para realizar la comunicación directa con la base de datos.

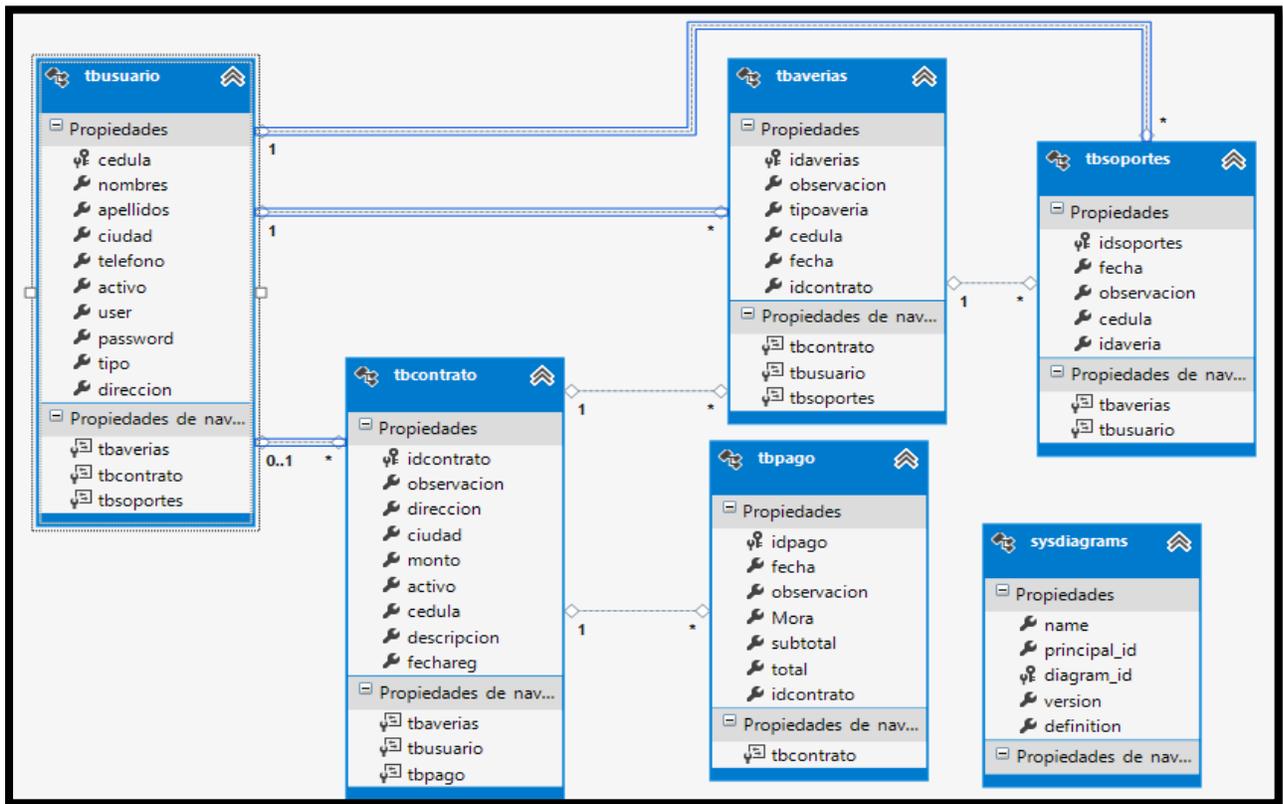


Fig.2

En la figura 3 se observa el método web para registrar un usuario a la base de datos, se declara el método el cual contiene cada uno de los parámetros que se necesita para obtener el usuario a registrarse. Se procede a llamar **AmericableBDEntidad** la cual contiene los elementos **CRUD** dentro del archivo **AmericableBD.edmx** luego se procede a llamar la tabla que se necesita utilizar con cada uno de los parámetros declarados primeramente en el método, una vez obtenidos los parámetros que necesitan se mandan a insertar a la base de datos **bd.tbusuario.Add()**; luego devuelve la respuesta si los datos se agregaron correctamente.

```
[WebMethod]
public int agregarusuario(string cedula, string nombre, string apellidos, string direccion, string ciudad, string telefono, string user, string password, Int16 tipo, Boolean activo)
{
    try
    {
        using (AmericableBDEntidad bd = new AmericableBDEntidad())
        {
            tbusuario t = new tbusuario()
            {
                cedula = cedula,
                nombres = nombre,
                apellidos = apellidos,
                direccion = direccion,
                ciudad = ciudad,
                telefono = telefono,
                user = user,
                password = password,
                tipo = tipo,
                activo = activo
            };
            bd.tbusuario.Add(t);
            return bd.SaveChanges();
        }
    }
    catch
    {
        return 0;
    }
}
```

Fig.3

Servicio web actualizar usuario Fig. 4, para actualizar un usuario primero se comprueba si el contrato existe, para luego guardar los cambios realizados:

```
[WebMethod]
public int actualizarusuario (string cedula, string nombre, string apellidos, string direccion, string ciudad, string telefono, string user, string password)
{
    try
    {
        using (AmericableBDEntidad bd = new AmericableBDEntidad())
        {
            var resultado = bd.tbusuario.Find(cedula);

            if(apellidos != "")
                resultado.apellidos = apellidos;
            if (ciudad!="")
                resultado.ciudad = ciudad;
            if (direccion!="")
                resultado.direccion = direccion;
            if (nombre != "")
                resultado.nombres = nombre;
            if (password!="")
                resultado.password = password;
            if (telefono!="")
                resultado.telefono = telefono;
            if (user!="")
                resultado.user = user;
            return bd.SaveChanges();
        }
    }
    catch
    {
        return 0;
    }
}
```

Fig.4

Servicio web eliminar usuario Fig. 5, para proceder a eliminar el usuario se necesita como dato esencial su número de cedula, se valida la existencia de la cedula con los datos del cliente y luego son eliminados.

```
[WebMethod]
public int eliminarusuario(string cedula)
{
    try
    {
        using (AmericableBDEntidad bd = new AmericableBDEntidad())
        {
            var resultado = bd.tbusuario.Find(cedula);
            bd.tbusuario.Remove (resultado);
            return bd.SaveChanges();
        }
    }
    catch
    {
        return 0;
    }
}
```

Fig.5

Servicio web agregar pago Fig. 6, para realizar el pago se necesita el método web con la fecha que realizara el pago, observación si la hay, mora si existe y el total a pagar, todos estos datos son necesarios para realizar el pago de la factura del mes.

```
[WebMethod]
public int agregarpagos(DateTime fecha, string observacion, float mora, float subtotal, float total, int idcontrato)
{
    try
    {
        using (AmericableBDEntidad bd = new AmericableBDEntidad())
        {
            tbpago t = new tbpago()
            {
                fecha = fecha,
                idcontrato = idcontrato,
                Mora = mora,
                observacion = observacion,
                subtotal = subtotal,
                total = total
            };
            bd.tbpago.Add(t);
            return bd.SaveChanges();
        }
    }
    catch
    {
        return 0;
    }
}
```

Fig. 6

Servicio web listar pago y eliminar pago: para listar los pagos se realiza la clase correspondiente en la cual se muestra el id del contrato, el id del pago, la mora y la observación, luego ser buscado se retorna la lista de los pagos de los clientes.

```
public List<listpagos> listarpago()
{
    using (AmericableBDEntidad bd = new AmericableBDEntidad())
    {
        List<listpagos> lista = new List<listpagos>();
        var resultado = bd.listarpagos();
        foreach (var fila in resultado)
        {
            lista.Add(new listpagos { fecha = fila.fecha, idcontrato = fila.idcontrato, idpago = fila.idpago, mora = fila.mora, observacion = fila.observacion });
        }
        return lista;
    }
}

[WebMethod]
public int eliminarpago(int idpago)
{
    try
    {
        using (AmericableBDEntidad bd = new AmericableBDEntidad())
        {
            var resultado = bd.tbpago.Find(idpago);
            bd.tbpago.Remove(resultado);
            return bd.SaveChanges();
        }
    }
    catch
    {
        return 0;
    }
}
```

Fig. 7

Servicio web listar averías Fig. 8: para listar las averías que el cliente puede reportar se muestran los tipos de averías, una vez que es reportada se asigna un empleado de soporte técnico para que dé solución a la avería.

```
public List<tipoaverias> listaraverias()
{
    List<tipoaverias> t = new List<tipoaverias>();
    string vnombempleado = "";
    string vcedulaempleado = "";
    int vidsoporte=0;
    string vobsensoporte = "";
    using (AmericableBDEntidad db = new AmericableBDEntidad())
    {
        var resultado = db.listaraverias();

        foreach (var fila in resultado)
        {
            var resultado2 = db.listarsoportest(fila.idaverias);

            foreach (var fila2 in resultado2){
                vnombempleado = fila2.nombreE;
                vcedulaempleado = fila2.cedula;
                vidsoporte = fila2.idsoportes;
                vobsensoporte = fila2.observacion;
            }
            t.Add(new tipoaverias { cedula = fila.cedula, descripcion = fila.tipoaveria, fecha = fila.fecha, idcontrato
        }
        return t;
    }
}
```

Fig. 8

Servicio web comprobar usuario Fig. 9: para comprobar el usuario que inicia sesión se listan los usuarios existentes del sistema, luego que se muestran los resultados se valida si está activo para permitir el inicio de sesión.

```
[WebMethod]
public List<listauser> comprobarusuario(string vuser)
{
    List<listauser> t = new List<listauser>();
    using (AmericableBDEntidad db = new AmericableBDEntidad())
    {
        var resultado = db.comprobarusuario(vuser);
        foreach (var fila in resultado)
            t.Add(new listauser { activo = fila.activo, apellidos = fila.apellidos, cedula = fila.cedula, ciudad=fila.c:

        return t;
    }
}

[WebMethod]
public List<listauser> sesionusuario(string vuser, string vpassword)
{
    List<listauser> t = new List<listauser>();
    using (AmericableBDEntidad db = new AmericableBDEntidad())
    {
        var resultado = db.listarusuarios(vuser, vpassword);
        foreach (var fila in resultado)
            t.Add(new listauser { activo = fila.activo, apellidos = fila.apellidos, cedula = fila.cedula, ciudad = fila

        return t;
    }
}
```

Fig. 9

Servicio web comprobar contrato, listar empleado Fig.10: para comprobar si el contrato existe solo se busca por medio del id para verificar si el servicio está activo, luego se listan los resultados y se muestran de estar activo.

```
[WebMethod]
public Boolean comprobarcontrato(string idcontrato)
{
    using (AmericableBDEntidad db = new AmericableBDEntidad())
    {
        var resultado = db.comprobarcontrato(int.Parse(idcontrato));
        foreach (var fila in resultado)
            return true;

        return false;
    }
}

[WebMethod]
public List<listempleadoLista> listareempleados()
{
    List<listempleadoLista> lista = new List<listempleadoLista>();
    using (AmericableBDEntidad bd = new AmericableBDEntidad())
    {
        var resultado = bd.listareempleados();
        foreach (var fila in resultado)
        {
            lista.Add(new listempleadoLista { cedula = fila.cedula, nombre = fila.Nombre });
        }

        return lista;
    }
}
```

Fig.10

Servicio web listar contrato, listar datos de los contratos: para listar los contratos se realiza la lista de los contratos y esta muestra la cedula, dirección, descripción y demás datos del cliente que realizo el contrato y se listan.

```
[WebMethod]
public List<listarcontrato> listarcontratos()
{
    List<listarcontrato> lista = new List<listarcontrato>();
    using (AmericableBDEntidad bd = new AmericableBDEntidad())
    {
        var resultado = bd.listarcontrato();
        foreach (var fila in resultado)
        {
            lista.Add(new listarcontrato { cedula = fila.cedula, descripcion = fila.descripcion, direccion1 = fila.direc
        }
        }
    }
}

[WebMethod]
public List<datoscontrato> listardatoscontrato(int idcontrato)
{
    List<datoscontrato> lista = new List<datoscontrato>();
    using (AmericableBDEntidad bd = new AmericableBDEntidad())
    {
        var resultado = bd.cargarcontratodatos(idcontrato);
        foreach (var fila in resultado)
        {
            lista.Add(new datoscontrato { apellidos = fila.apellidos, cedula = fila.cedula, descripcion = fila.descripcion
        }
    }
    return lista;
}
```

Fig.11

El método web listar contrato, lista los contratos que se han guardado. Para listar los pagos de los contratos se necesita el id del contrato luego se realiza una búsqueda del contrato, los pagos realizados a este y se lista.

Mostrar pago por contrato: acá se listan los datos de los empleados registrados en el sistema.

```
[WebMethod]
public List<listempleado> listarepleadocompleto()
{
    List<listempleado> lista = new List<listempleado>();
    using(AmericableBDEntidad bd = new AmericableBDEntidad())
    {
        var resultado = bd.listarepleadocompleto();
        foreach ( var fila in resultado)
        {
            lista.Add(new listempleado { apellido = fila.apellidos, cedula = fila.cedula, ciudad = fila.ciudad, direccion = fila.direccion});
        }
        return lista;
    }
}

[WebMethod]
public List<listpagoscliente> listarpagoporcontrato(string idcontrato)
{
    List<listpagoscliente> lista = new List<listpagoscliente>();
    using (AmericableBDEntidad bd = new AmericableBDEntidad())
    {
        var resultado = bd.listarpagosclienteContrato(int.Parse(idcontrato));
        foreach (var fila in resultado)
        {
            lista.Add( new listpagoscliente{ apellidos = fila.apellidos, cedula=fila.cedula, fecha = fila.fecha, idcontrato = fila.idcontrato});
        }
        return lista;
    }
}
```

Fig.

12

En la Fig.13 se muestran todas las clases de las cuales se hacen uso los servicios web.

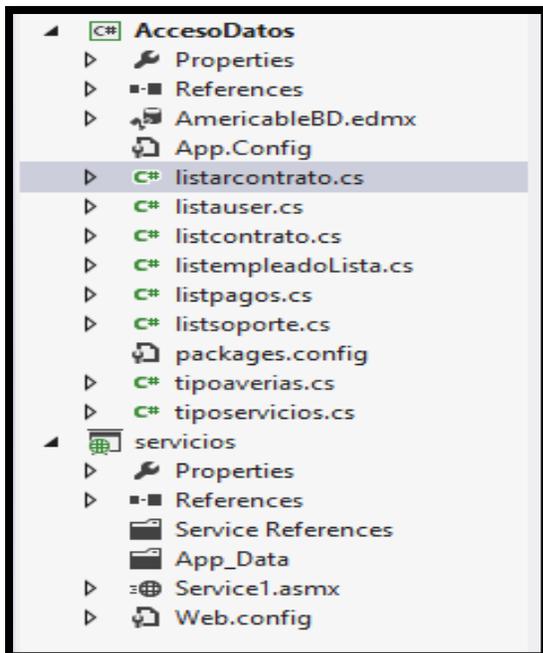


Fig.13

La Fig.14 muestra la clase listar contrato que se utiliza para listar los contratos vigentes en el sistema.

```
namespace AccesoDatos
{
    public class listarcontrato
    {
        public int idcontrato { get; set; }
        public string nombre { get; set; }
        public string descripcion { get; set; }
        public string direccion1 { get; set; }
        public string direccion2 { get; set; }
        public double monto { get; set; }
        public string telefono { get; set; }
        public string cedula { get; set; }
    }
}
```

Fig.14

En la fig.15 es utilizada para mostrar la lista de los usuarios del sistema.

```
namespace AccesoDatos
{
    public class listauser
    {
        public string cedula{ get; set; }
        public string nombres { get; set; }
        public string apellidos { get; set; }
        public string direccion { get; set; }
        public string ciudad { get; set; }
        public string telefono { get; set; }
        public string user { get; set; }
        public string password { get; set; }
        public Int16 tipo { get; set; }
        public Boolean activo { get; set; }
    }
}
```

Fig.15

En la Fig.16 se muestra la clase listar contratos en la que se listan los contratos registrados en el sistema y datos del contrato que muestra los datos del contrato cuando son solicitados.

```
namespace AccesoDatos
{
    public class listcontrato
    {
        public int idcontrato { get; set; }
        public string observacion { get; set; }
        public string direccion { get; set; }
        public string ciudad { get; set; }
        public float monto { get; set; }
        public Boolean activo { get; set; }
        public string cedula { get; set; }
        public string descripcion { get; set; }
        public DateTime fecha { get; set; }
    }

    public class datoscontrato
    {
        public string cedula { get; set; }
        public string descripcion { get; set; }
        public string direccion { get; set; }
        public double monto { get; set; }
        public string nombres { get; set; }
        public string apellidos { get; set; }
    }
}
```

Fig.16

La fig.17 muestra la clase listar empleado que busca los empleados registrados en el sistema y los muestra

```
namespace AccesoDatos
{
    public class listempleadolista
    {
        public string nombre { get; set; }
        public string cedula { get; set; }
    }

    public class listempleado
    {
        public string cedula { get; set; }
        public string nombre { get; set; }
        public string apellido { get; set; }
        public string direccion { get; set; }
        public string ciudad { get; set; }
        public string telefono { get; set; }
    }
}
```

Fig.17

En la Fig.18 se muestra la clase listar pagos, en esta clase se listan los pagos de los clientes y son mostrados en el estado de cuenta del cliente.

```
namespace AccesoDatos
{
    public class listpagos
    {
        public int idpago { get; set; }
        public DateTime fecha { get; set; }
        public string observacion { get; set; }
        public double mora { get; set; }
        public double subtotal { get; set; }
        public double total { get; set; }
        public int idcontrato { get; set; }
    }

    public class listpagoscliente
    {
        public int idpago { get; set; }
        public DateTime fecha { get; set; }
        public string observacion { get; set; }
        public double mora { get; set; }
        public double subtotal { get; set; }
        public double total { get; set; }
        public int idcontrato { get; set; }
        public string cedula { get; set; }
        public string nombres { get; set; }
        public string apellidos { get; set; }
    }
}
```

Fig.18

En la Fig.19 se convocan los tipos de averías que son reportados por los clientes.

```
namespace AccesoDatos
{
    public class tipoaverias
    {
        public int id { get; set; }
        public string descripcion { get; set; }
        public string observación { get; set; }
        public string cedula { get; set; }
        public DateTime fecha { get; set; }
        public int idcontrato { get; set; }
        public string nombempleado { get; set; }
        public string cedulaempleado { get; set; }
        public int idsoporte { get; set; }
        public string observacionsoporte { get; set; }
    }
}
```

Fig.19

d) Creando Procesos Cliente

Menú principal del sistema SCAA realizado en HTML con CSS y utilizando el framework de Bootstrap para dar estilos al sistema.

Para ingresar al sistema debe haber un usuario para esto se han creado tres tipos de usuario:

- Administrador inicia sesión con el valor tipo = 0
- Empleado inicia sesión con el valor tipo =1
- Cliente inicia sesión con el tipo de valor =2

Se realizó en una página maestra (main:Master) luego se llamaron cada una de las paginas hijas realizadas las que contienen las acciones y métodos para la funcionalidad del sistema.

Una vez iniciada la sesión según el tipo de usuario se procede a mostrar el menú que corresponde a cada usuario:

En la Fig.20 se muestra el código html del menú de usuario administrador

```
<div class="navbar-wrapper">
<div class="container-fluid">
<nav class="navbar">
<div class="container">
<div class="navbar-header">
<button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#navbar" aria-expanded="false" aria-controls="navbar">
<span class="sr-only">Toggle navigation</span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
</button>
</div>
<div id="navbar" class="navbar-collapse collapse">
<ul class="nav navbar-nav">
<li class="active"><a href="WebForm1.aspx" class="">Inicio</a></li>
<li><a href="contrato.aspx" runat="server" id="btncontratos">Contratos</a></li>
<li><a href="empleados.aspx" runat="server" id="btnempleados">Empleados</a></li>
<li><a href="pagos.aspx" class="" runat="server" id="btnpagos">Pagos</a></li>
<li class="dropdown">
<a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true" aria-expanded="false">Servicios<span class="caret"></span></a>
<ul class="dropdown-menu">
<li class="dropdown">
<a href="tipoaverias.aspx" runat="server" id="btnaveria">Reportar averias</a>
</li>
<li><a href="soporte.aspx" runat="server" id="btnsoporte">Soportes</a></li>
</ul>
</li>
<li class="dropdown" runat="server" id="btnreportes"><a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true" aria-expanded="false">Reportes<span class="caret"></span></a>
<ul class="dropdown-menu">
<li><a href="#">Contratos</a></li>
<li><a href="reportpagos.aspx">Pagos</a></li>
</ul>
</li>
</ul>
</div>
</div>
</div>
</div>
```

Fig.20

En la Fig. 21 se muestran todas las carpetas de menú y las páginas .aspx que se crearon para la funcionalidad del sistema.

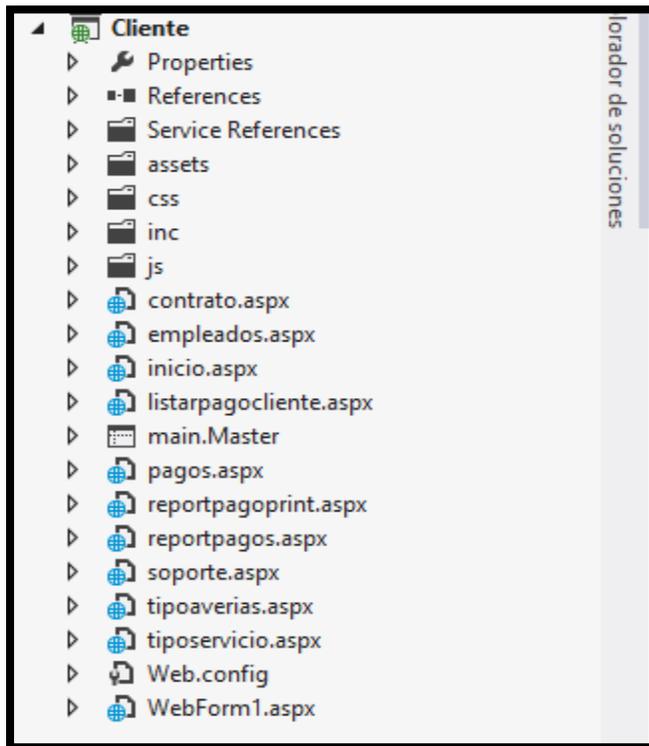


Fig. 21

En la carpeta Service Reference se encuentra la dirección del servidor con el cual se comunica el cliente y es donde se encuentran los servicios web con los realiza la conexión para su debido funcionamiento.

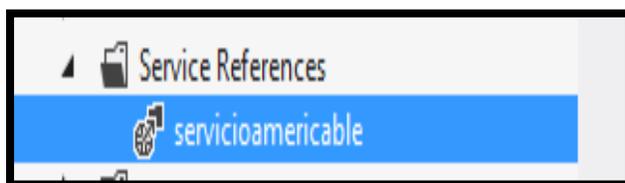


Fig. 22

En la carpeta assets se pueden encontrar las imágenes que se utilizan para dar él diseño al sistema.



Fig. 23

En la carpeta css se encuentran las librerías de bootstrap y los estilos que se realizaron para dar el estilo y diseño al sistema.

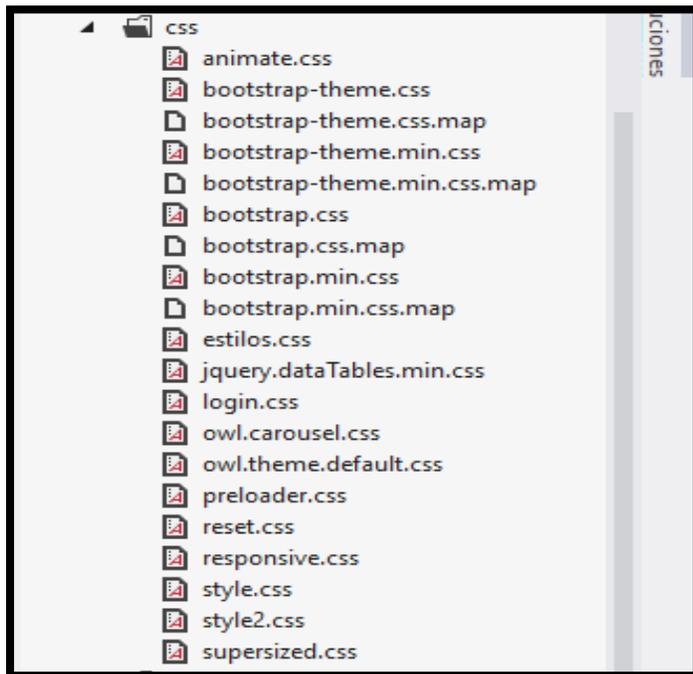


Fig. 24

En la carpeta inc se encuentran las acciones que se realizan en el sistema como los son cerrar sesión, iniciar sesión, y acciones de eliminar o modificar algún archivo creado.

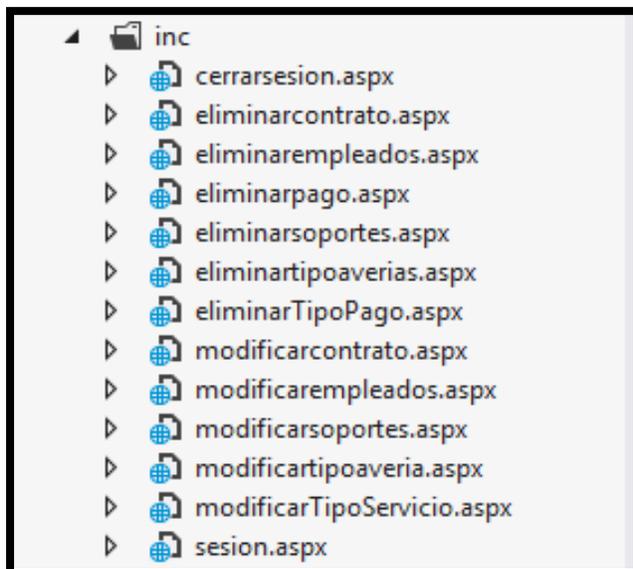


Fig. 25

Para el inicio de sesión se utilizan variables de sesión las cuales son utilizadas para el ingreso del tipo de usuario al sistema, para esto se listan los usuarios del sistema en los cuales se comparan los datos proporcionados por el cliente para comprobar que estén dentro del sistema y así darle acceso al sistema enviándolo a la página que le corresponde según el tipo de usuario (Administrador, Cliente, Empleado).

```
string vuser = (Request.Form["txtusername"].ToString());
string pass = (Request.Form["txtpassword"].ToString());

Cliente.servicioamericable.Service1SoapClient ser = new Cliente.servicioamericable.Service1SoapClient();

var resultado = ser.sesionusuario(vuser, pass);

if (resultado.Count != 0)
{
    foreach (var datos in resultado)
    {
        if (datos.activo == false)
        {
            Session["name"] = datos.nombres;
            Session["apell"] = datos.apellidos;
            Session["ced"] = datos.cedula;
            Session["tip"] = datos.tipo;
            Session["user"] = datos.user;
            Session["activo"] = datos.activo;
            if (datos.tipo == 2)
            {
                Response.Redirect("../webform1.aspx");
            }
            else if ((datos.tipo == 0))
            {
                Response.Redirect("../listarpagocliente.aspx");
            }
        }
        else
        {
            Response.Redirect("../inicio.aspx");
        }
    }
}
```

Fig. 26

Para realizar el cierre de sesión solo se realiza el abandono de sesión y se envía a la página principal del sistema.

```
namespace Cliente.inc
{
    public partial class cerrarsesion : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            Session.Abandon();
            Response.Redirect("../inicio.aspx");
        }
    }
}
```

Fig. 27

Para guardar un contrato se solicitan los datos del cliente que son requeridos obligatoriamente para realizar el contrato:

```
<div class="panel-footer">
  <form action="" method="post">

  <div class="col-md-12">
    <div class="row">
      <div class="col-md-6">
        <label for="basic-url1" style="font-size: 18px">Datos del Cliente:</label>
        <div class="input-group">
          <span class="input-group-addon" id="basic-addon3" style="font-size: 20px">(*) Cédula del Cliente</span>
          <asp:TextBox ID="txtcedula" runat="server" CssClass="form-control" aria-describedby="basic-addon3" placeholder="Ingrese la cédula del cliente" Font-Size=
        </div>
        <br />
        <div class="input-group">
          <span class="input-group-addon" id="basic-addon3" style="font-size: 20px">(*) Nombre del Cliente</span>
          <asp:TextBox ID="txtnombre" runat="server" CssClass="form-control" aria-describedby="basic-addon3" placeholder="Ingrese el nombre del cliente" Font-Size=
        </div>
        <br />
        <div class="input-group">
          <span class="input-group-addon" id="basic-addon3" style="font-size: 20px">(*) Apellidos del Cliente</span>
          <asp:TextBox ID="txtapellidos" runat="server" CssClass="form-control" aria-describedby="basic-addon3" placeholder="Ingrese el apellido del cliente" Font-Size=
        </div>
        <br />
        <div class="input-group">
          <span class="input-group-addon" id="basic-addon3" style="font-size: 20px">(*) Dirección del Cliente</span>
          <asp:TextBox ID="txtdireccion" runat="server" CssClass="form-control" aria-describedby="basic-addon3" placeholder="Ingrese la dirección del cliente" Font-Size=
        </div>
        <br />
        <div class="input-group">
          <span class="input-group-addon" id="basic-addon3" style="font-size: 20px">(*) Ciudad del cliente</span>
          <asp:TextBox ID="txtciudad" runat="server" CssClass="form-control" placeholder="Ciudad del cliente" Font-Size="18pt" aria-describedby="basic-addon3"></asp
        </div>
        <br />
      </div>
    </div>
  </div>
</form>
</div>
```

Fig.28

Luego se guardan y se muestran para verificar que el contrato se guardó correctamente:

Para mostrar los datos se utilizan tablas llamando los datos directamente desde la base de datos.

Se capturan cada uno de los datos solicitados al cliente, ya que es obligatorio llenar cada uno de los campos y se selecciona el servicio para luego ser guardados.

```
protected void Page_Load(object sender, EventArgs e)
{
    txtcedula.Focus();
}
protected void guardarusuario(object sender, EventArgs e)
{
    string vcedula = txtcedula.Text.ToString();
    string vnombres = txtnombre.Text.ToString();
    string vapellidos = txtapellidos.Text.ToString();
    string vdireccion = txtdireccion.Text.ToString();
    string vciudad = txtciudad.Text.ToString();
    string vtelefono = txttelf.Text.ToString();
    string vuser = generarusuario();
    string vpassword = generarpassword();
    Int16 vtipo = 0;
    Boolean vactivo = false;
    string vdescripcion = Request.Form["cbservicios"].ToString();
    string vciudadC = txtCiudad2.Text.ToString();
    string vdireccion2 = txtdireccion2.Text.ToString();
    string vobservacion = txtobservacion.Text.ToString();
    string vmonto = txtmonto.Text.ToString();
    DateTime vfechaReg = DateTime.Now;

    if ((vcedula == "") || (vdireccion2=="") || (vciudadC=="") || (vmonto=="") || (vnombres == "") || (vapellidos == "") || (vdireccion == "") || (vciudad == "") || (vtele
    {
        mensajerellene.Visible = true;
        return;
    }
    Cliente.servicioamercable.Service1SoapClient serv = new Cliente.servicioamercable.Service1SoapClient();
    serv.agregarusuario(vcedula, vnombres, vapellidos, vdireccion, vciudad, vtelefono, vuser, vpassword, vtipo, vactivo);
    serv.agregarcontrato(vobservacion, vdireccion2, vciudadC, float.Parse(vmonto), vactivo, vcedula, vdescripcion, vfechaReg);

    Response.Redirect("contrato.aspx?guardado=OK");
}
```

Fig.29

El usuario y la contraseña son generados automáticamente mediante la función Random como se muestra en la Fig.23, Fig.24, que nos permite generar una combinación aleatoria de caracteres y números en cada uno de los métodos.

```
private string generarusuario()
{
    int num;
    string cadena;
    Random aleatorio = new Random();
    do
    {
        cadena = "";
        for (int i = 1; i < 6; i++)
        {
            num = aleatorio.Next(65, 90);
            cadena += Convert.ToChar(num);
        }
        cadena += aleatorio.Next(100, 200);
    } while (verificarusuario(cadena) != false);
    return cadena;
}
```

Fig.30

```
private string generarpassword()
{
    int num = 0;
    int num2 = 0;
    string cadena = "";
    Random aleatorio = new Random();
    num = aleatorio.Next(3, 6);
    for (int i = 1; i < num; i++)
    {
        num2 = aleatorio.Next(65, 90);
        cadena += Convert.ToChar(num2);
    }
    num = aleatorio.Next(3, 5);
    for (int i = 1; i < num; i++)
    {
        num2 = aleatorio.Next(1, 100);
        cadena += num2;
    }
    return cadena;
}
```

Fig.31

La Fig.32 muestra la tabla de presentación de datos de contratos realizados.

```

<!-- Tabla para listar los datos de contratos -->
<div class="panel panel-info" style="padding: 0 18px 0 18px; line-height: 17px;" >
  <div class="table-responsive">
    <table id="contratos" class="table table-bordered panel-primary">
      <thead>
        <tr>
          <th>#</th>
          <th>Acciones</th>
          <th># contrato</th>
          <th>Nombre del Cliente</th>
          <th>Descripción</th>
          <th>Dirección del Cliente</th>
          <th>Dirección del Servicio</th>
          <th>Monto</th>
          <th>Telefono</th>
          <th>Cédula</th>
          <th>Acciones</th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <td><Response.Write(n); </td>
          <td>
            <a href="#" class="btn btn-primary btn-sm" data-toggle="modal" data-target="#<Response.Write(filea.idcontrato); </td>
            <a href="inc/eliminarcontrato.aspx?id=<Response.Write(filea.idcontrato); </td>
            <td><Response.Write(filea.idcontrato); </td>

```

Fig.32

Para registrar un empleado se registran los datos en la base de datos cada uno de estos datos debe ser llenado obligatoriamente:

```

<h3><span class="label label-default">Registro de Empleados</span></h3>
</div>
<div class="panel-footer">
  <form action="" method="post">
    <label for="basic-url" style="font-size: 18px">Datos del empleado:</label>
    <div class="input-group">
      <span class="input-group-addon" id="basic-addon3" style="font-size: 20px">(*) Cédula del empleado</span>
      <asp:TextBox ID="txtcedula" runat="server" CssClass="form-control" aria-describedby="basic-addon3" placeholder="Ingrese la cédula del empleado" Font-Size="18pt"></asp:Text
    </div>
    <br />
    <div class="input-group">
      <span class="input-group-addon" id="basic-addon3" style="font-size: 20px">(*) Nombre del empleado</span>
      <asp:TextBox ID="txtnombre" runat="server" CssClass="form-control" aria-describedby="basic-addon3" placeholder="Ingrese el nombre del empleado" Font-Size="18pt"></asp:Text
    </div>
    <br />
    <div class="input-group">
      <span class="input-group-addon" id="basic-addon3" style="font-size: 20px">(*) Apellidos del empleado</span>
      <asp:TextBox ID="txtapellidos" runat="server" CssClass="form-control" aria-describedby="basic-addon3" placeholder="Ingrese el apellido del empleado" Font-Size="18pt"></asp:Text
    </div>
    <br />
    <div class="input-group">
      <span class="input-group-addon" id="basic-addon3" style="font-size: 20px">(*) Dirección del empleado</span>
      <asp:TextBox ID="txtdireccion" runat="server" CssClass="form-control" aria-describedby="basic-addon3" placeholder="Ingrese la dirección del empleado" Font-Size="18pt"></asp:Text
    </div>
    <br />
    <div class="input-group">
      <span class="input-group-addon" id="basic-addon3" style="font-size: 20px">(*) Ciudad del empleado</span>
      <asp:TextBox ID="txtciudad" runat="server" CssClass="form-control" placeholder="Ciudad del empleado" Font-Size="18pt" aria-describedby="basic-addon3"></asp:Text
    </div>
    <br />

```

Fig.33

Una vez que se obtienen los datos del empleado a registrarse se validan los campos vacíos y se guardan los datos como se muestra en la Fig.34

```

public partial class empleados : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        txtcedula.Focus();
    }

    protected void guardarusuario(object sender, EventArgs e)
    {
        string vcedula = txtcedula.Text.ToString();
        string vnombres = txtnombre.Text.ToString();
        string vapellidos = txtapellidos.Text.ToString();
        string vdireccion = txtdireccion.Text.ToString();
        string vciudad = txtciudad.Text.ToString();
        string vtelefono = txttelf.Text.ToString();
        string vuser = generarusuario();
        string vpassword = generarpassword();

        if ((vcedula == "") || (vnombres=="") || (vapellidos=="")|| (vdireccion == "") || (vciudad=="")|| (vtelefono == ""))
        {
            mensajerellene.Visible = true;
            return;
        }

        Int16 vtipo = 1; //tipo de usuario empleado
        Boolean vactivo = false;

        DateTime vfechaereg = DateTime.Now;

        servicioamericable.Service1SoapClient serv = new servicioamericable.Service1SoapClient();
        serv.agregarusuario(vcedula, vnombres, vapellidos, vdireccion, vciudad, vtelefono, vuser, vpassword, vtipo, vactivo);
        Response.Redirect("empleados.aspx");
        mensaje.Visible = true;
    }
}

```

Fig.34

Una vez registrado el empleado se muestra como se observa en la Fig. 35:

```

<!-- Tabla de lista de datos -->
<div class="panel panel-info" style="padding: 0 18px 0 18px; line-height: 17px;" >
<div class="table-responsive">
    <table id="empleados" class="table table-bordered panel-primary">
        <thead>
            <tr>
                <th>#</th>
                <th>Cédula</th>
                <th>Nombres</th>
                <th>Apellidos</th>
                <th>Dirección</th>
                <th>Ciudad</th>
                <th>Telefono</th>
                <th>Acciones</th>
            </tr>
        </thead>
        <tbody>
            <%
                var ser = new Cliente.servicioamericable.Service1SoapClient();
                int n = 1;
                foreach (var fila in ser.listarepleadocompleto())
            >
                <tr>
                    <td><%Response.Write(n); %></td>
                    <td><%Response.Write(fila.cedula); %></td>
                    <td><%Response.Write(fila.nombre); %></td>
                    <td><%Response.Write(fila.apellido); %></td>
                    <td><%Response.Write(fila.direccion);%></td>
                    <td><%Response.Write(fila.ciudad); %></td>
                    <td><%Response.Write(fila.telefono); %></td>
                <td>
            <%
                </tbody>
            </div>
        </div>

```

Fig.35

Se llama el método para guardar usuario el cual guardara cada uno de los datos que son requeridos, el usuario y contraseña es igualmente generado por medio del Random:

```
protected void guardarusuario(object sender, EventArgs e)
{
    string vcedula = txtcedula.Text.ToString();
    string vnombres = txtnombre.Text.ToString();
    string vapellidos = txtapellidos.Text.ToString();
    string vdireccion = txtdireccion.Text.ToString();
    string vciudad = txtciudad.Text.ToString();
    string vtelefono = txttelf.Text.ToString();
    string vuser = generarusuario();
    string vpassword = generarpassword();

    if ((vcedula == "") || (vnombres=="") || (vapellidos=="")|| (vdireccion == "") || (vciudad=="") || (vtelefono == ""))
    {
        mensajerellene.Visible = true;
        return;
    }
    Int16 vtipo = 1; //tipo de usuario empleado
    Boolean vactivo = false;

    DateTime vfechaereg = DateTime.Now;

    servicioamericable.Service1SoapClient serv = new servicioamericable.Service1SoapClient();
    serv.agregarusuario(vcedula, vnombres, vapellidos, vdireccion, vciudad, vtelefono, vuser, vpassword, vtipo, vactivo);
    Response.Redirect("empleados.aspx");
    mensaje.Visible = true;
}
}
```

Fig.36

Para realizar el pago se debe tener el número de contrato, luego el sistema llama el contrato con los datos del cliente y muestra el total a pagar de la factura:

```
<div class="panel-body">
<h3><span class="label label-default">Registro de Pagos</span></h3>
</div>
<div class="panel-footer">
<form method="post">

<div class="input-group">
<span class="input-group-addon" id="basic-addon3" style="font-size: 20px">(*) Número del Contrato</span>
<asp:TextBox ID="txtcontrato" runat="server" CssClass="form-control" aria-describedby="basic-addon3" placeholder="Ingrese su número de contrato" Font-Size="18pt"></asp:Text
</div>
<button type="button" class="btn btn-block" runat="server" onclick="comprobar">Comprobar</button>
</div>
<br />

<div class="input-group">
<span class="input-group-addon" id="basic-addon3" style="font-size: 20px">Mora</span>
<asp:TextBox required="" ID="txtmora" runat="server" CssClass="form-control" aria-describedby="basic-addon3" placeholder="Ingrese la mora" Font-Size="18pt"></asp:Text
</div>
<br />

<div class="input-group">
<span class="input-group-addon" id="basic-addon3" style="font-size: 20px">Observaciones</span>
<asp:TextBox ID="txtobservacion" runat="server" CssClass="form-control" aria-describedby="basic-addon3" placeholder="Ingrese cualquier observación" Font-Size="18pt"></asp:Text
</div>
<br />
<label for="basic-url1" style="font-size: 18px">Datos del Cliente:</label>
<div class="input-group">
<span class="input-group-addon" id="basic-addon3" style="font-size: 20px">Nombre del Cliente</span>
<asp:Label ID="lblnombre" runat="server" CssClass="form-control" aria-describedby="basic-addon3" Font-Size="18pt"></asp:Label>
</div>
<br />
<div class="input-group">
<span class="input-group-addon" id="basic-addon3" style="font-size: 20px">Dirección del Cliente</span>
<asp:Label ID="lbldir" runat="server" CssClass="form-control" aria-describedby="basic-addon3" Font-Size="18pt"></asp:Label>
</div>
</div>
```

Fig.37

En la fig.38 se llama el método guardar pagos que contiene la observación, mora, subtotal y total del contrato y se comprueban los datos del dueño del contrato:

```

namespace Cliente
{
    public partial class pagos : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            txtmora.Enabled = false;
            txtobservacion.Enabled = false;
            btnguardar.Visible = false;
        }
        protected void guardarpagos(object sender, EventArgs e)
        {
            servicioamericable.Service1SoapClient serv = new servicioamericable.Service1SoapClient();
            serv.agregarpagos(DateTime.Now, txtobservacion.Text, float.Parse(txtmora.Text), float.Parse(txtsub.Text), (float.Parse(txtsub.Text) + float.Parse(txtmora.Text)), int.Pa
        }
        protected void comprobar(object sender, EventArgs e)
        {
            servicioamericable.Service1SoapClient serv = new servicioamericable.Service1SoapClient();
            foreach (var fila in serv.listardatoscontrato(Int16.Parse(txtcontrato.Text.Trim())))
            {
                lblnombre.Text = fila.nombres + " " + fila.apellidos;
                lbldescripcion.Text = fila.descripcion;
                lbldir.Text = fila.direccion;
                txtsub.Text = fila.monto.ToString();
            }
            txtmora.Enabled = true;
            txtobservacion.Enabled = true;
            txtmora.Focus();
            btnguardar.Visible = true;
        }
    }
}

```

Fig.38

Una vez que se muestra el contrato se realiza el pago, se guarda el pago realizado y se imprime la debida factura:

```

<h3><span class="label label-default">Historial de Pagos</span></h3>
</div>
<table class="table table-bordered panel-primary">
  <thead>
    <tr>
      <th>#</th>
      <th>Fecha</th>
      <th>Observación</th>
      <th>Mora</th>
      <th>Subtotal</th>
      <th>Total</th>
      <th>idcontrato</th>
    </tr>
  </thead>
  <tbody>
    <%
    var ser = new Cliente.servicioamericable.Service1SoapClient();
    string cedula = Session["ced"].ToString();
    int n = 1;
    foreach (var fila in ser.listarpagocliente(cedula))
    {
    <tr>
      <td><%Response.Write(n); %></td>
      <td><%Response.Write(fila.fecha); %></td>
      <td><%Response.Write(fila.observacion); %></td>
      <td><%Response.Write(fila.mora); %></td>
      <td><%Response.Write(fila.subtotal); %></td>
      <td><%Response.Write(fila.total); %></td>
      <td><%Response.Write(fila.idcontrato); %></td>
    </tr>
    <% n++;
    } %>
  </tbody>

```

Fig.39

Cuando el cliente reporta una avería se muestra en soporte para ser asignada a un empleado y luego ser solucionada:

```

<h3><span class="label label-default">Soporte técnico</span></h3>
</div>
<div class="table-responsive">
<table class="table table-bordered panel-primary">
<thead>
<tr>
<th>#</th>
<th># contrato</th>
<th>Descripción</th>
<th>Cédula</th>
<th>Empleado asignado</th>
<th>Fecha de registro</th>
<th>Observación</th>
<th>Acciones</th>
</tr>
</thead>
<tbody>
<%
var ser = new Cliente.servicioamericable.ServiceSoapClient();
int n = 1;
foreach (var fila in ser.Listaraverias())
{
<tr>
<td><Response.Write(n); %></td>
<td><Response.Write(fila.idcontrato); %></td>
<td><Response.Write(fila.descripcion); %></td>
<td><Response.Write(fila.cedula); %></td>
<td><Response.Write(fila.nombempleado); %></td>
<td><Response.Write(fila.fecha.ToShortDateString()); %></td>
<td><Response.Write(fila.observacion); %></td>
<td>
<a href="#" class="btn btn-primary btn-sm" data-toggle="modal" data-target="#"><span class="glyphicon glyphicon-pencil"></span></a>
<a href="inc/eliminarsoportes.aspx?id=<Response.Write(fila.id); %>&ids=<Response.Write(fila.idsoporte); %>" class="btn btn-danger btn-sm"><span class="glyphicon glyphicon-trash"

```

Fig.40

Una vez que se muestra la avería se le asigna un empleado para que le dé solución al problema reportado:

```

<div class="input-group">
<span class="input-group-addon" id="basic-addon3" style="font-size: 20px">Empleado:</span>
<select name="cbempleado" class="form-control">
<optgroup label="Seleccione el empleado">
<%
foreach (var fila2 in ser.listareempleados())
{
<option value="<Response.Write(fila2.cedula.ToString()); %>" <%if (fila.cedulaempleado == fi
<% %>
</optgroup>
</select>
</div>
<br />
<div class="input-group">
<span class="input-group-addon" id="basic-addon3" style="font-size: 20px">Observación:</span>
<input name="mobservacion" class="form-control" style="font-size: 18px" value="<Response.Write(fila.
</div>
<br />
<input name="mid" style="display: none" value="<Response.Write(fila.id); %>" />
<input name="midsorte" style="display: none" value="<Response.Write(fila.idsoporte); %>" />
<br />
<button type="submit" class="btn btn-primary" value="Guardar">Guardar</button>
</form>
</div>

```

Fig.41

Para seleccionar el tipo de avería se le muestran las averías que puede reportar el cliente y una observación si el cliente lo requiere:

```

<div class="panel panel-primary">
  <div class="panel-body">
    <h3> <span class="label label-default">Reporte su avería</span></h3>
  </div>
  <div runat="server" id="mensajerellene" class="alert alert-danger" role="alert" visible="false">Por favor rellene los campos obligatorios (*)</div>
  <div runat="server" id="mensajecontratoinvalido" class="alert alert-warning" role="alert" visible="false">El número de contrato es inválido</div>
  <div class="panel-footer">
    <form action="" method="post" >
      <label for="basic-url" style="font-size:18px">Descripción:</label>
      <div class="input-group">
        <span class="input-group-addon" id="basic-addon3" style="font-size:20px">(*) Número de Contrato:</span>
        <asp:TextBox ID="txtcontrato" runat="server" CssClass="form-control" aria-describedby="basic-addon3" placeholder="Ingrese el número de contrato" Font-Size="18pt">
        </div>
      <br />
      <div class="input-group">
        <span class="input-group-addon" id="basic-addon3" style="font-size:20px">(*) Tipo de avería:</span>
        <select name="cbaveria" class="form-control">
          <option value="" disabled selected>Seleccione un avería</option>
          <option value="Mala Señal">Mala Señal</option>
          <option value="Señal Interrumpida">Señal Interrumpida</option>
          <option value="Cable dañado">Cable dañado</option>
        </select>
      </div>
      <br />
      <div class="input-group">
        <span class="input-group-addon" id="basic-addon3" style="font-size:20px">Observación:</span>
        <asp:TextBox ID="textobservacion" runat="server" CssClass="form-control" aria-describedby="basic-addon3" placeholder="Ingrese la observación" Font-Size="18pt">
        </div>
      <br />
      <button type="button" id="btnguardar" class="btn btn-primary" runat="server" onclick="guardar_tipo" >Guardar</button>
    </form>
  </div>
</div>

```

Fig.42

Se captura por medio del id del contrato de la persona que realizo la avería y se guarda la avería una vez llenado los campos es guardada y se asigna un empleado para que dé solución:

```

public partial class tipoaverias : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }

    protected void guardar_tipo(object sender, EventArgs e)
    {
        string descripcion = Request.Form["cbaveria"].ToString();
        string observacion = textobservacion.Text.ToString();
        string idcontrato = txtcontrato.Text.ToString();
        string cedula = Session["ced"].ToString();

        if ((descripcion == "") || (idcontrato == ""))
        {
            mensajerellene.Visible = true;
            return;
        }
        Cliente.servicioamericable.Service1SoapClient serv = new Cliente.servicioamericable.Service1SoapClient();

        if (serv.comprobarcontrato(idcontrato) == false)
        {
            mensajecontratoinvalido.Visible = true;
            return;
        }
        DateTime fecha = DateTime.Now;
        serv.Agregaraveria(descripcion,observacion, cedula, fecha, Convert.ToInt32(idcontrato));
        Response.Redirect("tipoaverias.aspx");
    }
}

```

Fig.43

En la fig.44 se muestra el código para realizar la impresión del pago de la factura del cliente

```
<%
int idcontrato = int.Parse(Request.QueryString["id"].ToString());
string nombres = "";
string apellidos = "";
var serv = new Cliente.servicioamericable.Service1SoapClient();
foreach (var fila in serv.listarpagoporcontrato(idcontrato.ToString()))
{
    nombres = fila.nombres;
    apellidos = fila.apellidos;
}
%>
<div class="panel panel-primary" style="font-size:large;">
<div class="panel-body">
<div id="imprimir">
<b>Reporte de Pago </b>
<br />
<br />
<p>Número de contrato: <b><%Response.Write(idcontrato); %></b></p>
<p>
Nombre del Cliente: <b><%Response.Write(nombres); %></b><p>
Apellidos del Cliente: <b><%Response.Write(apellidos); %></b></p>
<br />
<b>Detalles de pago</b><br />
<br />
<table class="table table-bordered panel-primary">
<thead>
<tr>
<th>Fecha</th>
<th>Mora</th>
<th>Subtotal</th>
<th>Total</th>
<th>Observación</th>
</tr>
</thead>
```

Fig.44

Anexo11. Manual de Usuario

Manual de usuario del sistema de control de abonados de Americable (SCAA)

Índice

Introducción.....	1
Guía de uso.....	2

Introducción

El presente documento está dirigido para mostrar la funcionalidad y operaciones que realiza el sistema de control de abonados de Americable, para esto es necesario tener las herramientas adecuadas que nos permitan hacer un seguimiento de los procesos y tareas para realizar las acciones necesarias en las gestiones requeridas.

Acá se establecen los pasos específicos a seguir los cuales indican el buen funcionamiento del sistema, esto con el fin de facilitar el manejo y accesibilidad a las personas que lo manipularan.

Para la manipulación del sistema es obligatorio tener una cuenta como administrador, cliente o empleado, después de obtener la cuenta solo se requiere es un computador con su navegador y acceso a internet para de esta manera acceder al sistema sin ninguna dificultad.

Guía de uso

Para acceder al sistema se requiere estar registrado como usuario

- ✓ Administrador: para obtener una cuenta como administrador es necesario crear el usuario y contraseña desde la base de datos por medidas de seguridad y para controlar mejor quien tiene los mayores privilegios en el sistema.
- ✓ Cliente: para obtener un usuario de tipo cliente es necesario contratar el servicio de televisión por cable ofrecido por la empresa.
- ✓ Empleado: debe obtener su usuario mediante el registro realizado por el administrador como usuario empleado.

A continuación, se presenta la pantalla de inicio una vez que ha cargado la página principal del sistema se presenta el menú para el acceso al sistema.

La Fig.1 muestra la pantalla de inicio de sesión del sistema que detecta automáticamente que tipo de usuario es y de esta manera lo dirige al menú indicado.

Se muestran las acciones a realizar

★ AMERICABLE ★

INICIO UBICANOS NOSOTROS INICIAR SESION

INICIAR SESION

LAMANOS
+(505) 2512 - 2322

ESCRIBENOS
americable@americable.com

VISITANOS
Juigalpa Chontales, Nicaragua

Usuario

Contraseña

Iniciar

Fig.1

La Fig.2 muestra la ubicación de la empresa en el mapa a través de la aplicación google maps esto con el fin de facilitar el acceso a la empresa si el cliente no conoce su ubicación.

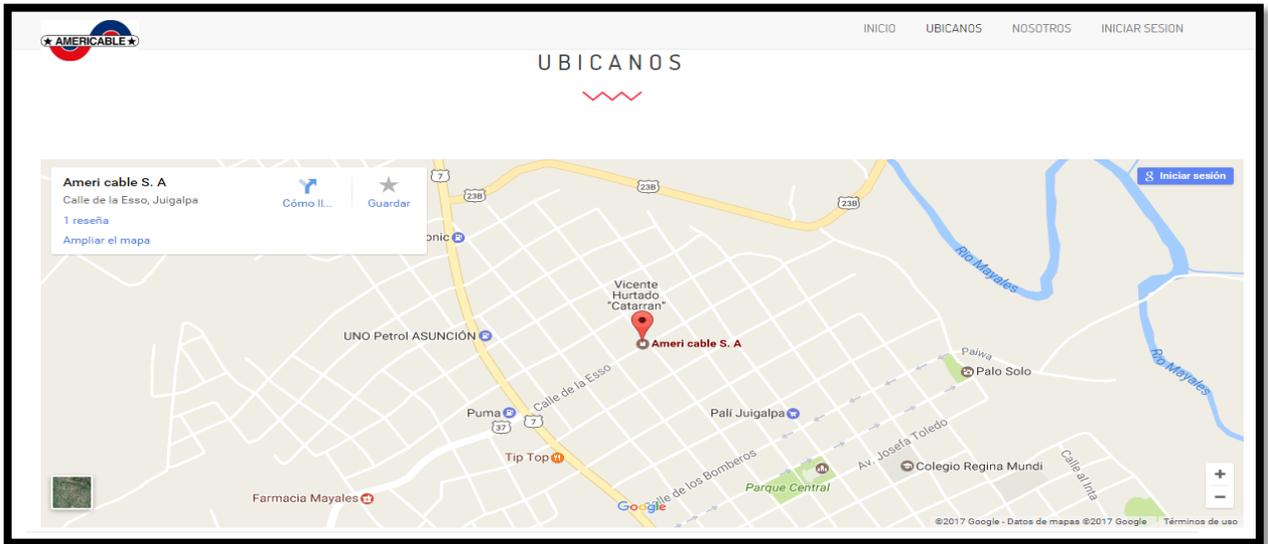


Fig.2

La Fig.3 muestra la misión, visión y valores éticos de la empresa.



Fig.3

La Fig.4 muestra la pantalla una vez que se inició de sesión como cliente.

The screenshot shows a web interface for a client. At the top, there is a navigation bar with 'Inicio', 'Servicios', 'ariel', and 'Cerrar sesión'. Below this is a 'Reportar averías' button. A callout box labeled 'Menu de acciones como cliente' points to the 'Servicios' dropdown menu. Below the button is a 'Historial de Pagos' section with a callout box labeled 'Historial de pagos realizados' pointing to a table. The table has the following data:

#	Fecha	Observación	Mora	Subtotal	Total	idcontrato
1	12/12/2016 12:00:00 a. m.	retraso	30	350	380	1
2	12/12/2016 12:00:00 a. m.	ninguna	0	350	350	1

Fig.4

Al dar clic en reportar averías se muestra el menú de la Fig.5

The screenshot shows the 'Reporte su avería' form. A callout box labeled 'Las acciones que se muestran es para reportar una avería con el número de contrato y el tipo de avería que tiene el servicio además de una observación si la hay' points to the form fields. The form includes a 'Descripción:' label, a text input for '(*) Número de Contrato:' with the placeholder 'Ingrese el número de contrato', a dropdown for '(*) Tipo de avería:' with options 'Seleccione un avería', 'Mala Señal', 'Señal Interrumpida', and 'Cable dañado', and a text input for 'Observación:' with the placeholder 'Ing...'. A 'Guardar' button is at the bottom left.

Fig.5

La Fig.6 muestra el menú de acceso para el usuario administrador en donde se pueden realizar las acciones:

- ✓ Realizar contrato
- ✓ Registrar empleado
- ✓ Registrar pagos
- ✓ Cerrar sesión



Fig.6

La Fig.7 muestra el sub menú

- ✓ Reportar averías
- ✓ Soporte



Fig.7

La Fig.8 muestra el sub menú

- ✓ reportes de contratos
- ✓ pagos realizados



Fig.8

Al dar clic en la pestaña contratos se cargara la ventana que se observa en la Fig.9 que pertenece a la realización del contrato de servicio de televisión por cable con los datos del cliente debidamente verificados con la cedula de identidad que es el único requisito para contratar el servicio.

Registro de nuevos contratos

Datos del Cliente:

(*) Cédula del Cliente

(*) Nombre del Cliente

(*) Apellidos del Cliente

(*) Dirección del Cliente

(*) Ciudad del cliente

(*) Telefono

Datos del servicio contratado

(*) Descripción del servicio

(*) Ciudad

(*) Dirección

Observación

(*) Monto

Fig.9

Al dar clic en guardar el contrato se guarda en la base de datos del sistema

Cuando los datos del contrato se han guardado en la base de datos, se muestran como en la Fig.10 para verificar que el contrato se guardó correctamente solo se da clic en búsqueda y se ingresa el número de cedula de esta manera aparecerán los datos que hemos guardado.

El sistema tiene dos acciones las cuales son modificar contratos que corresponde al símbolo azul con la figura de un lápiz y borrar que corresponde al símbolo rojo con la figura de un bote de basura de esta manera se puede actualizar los datos del cliente al modificarlos o eliminarlo si el cliente no desea continuar con el servicio.

Mostrar 10 entradas Busqueda:

#	Acciones	# contrato	Nombre del Cliente	Descripción	Dirección del Cliente	Dirección del Servicio	Monto	Telefono	Cédula	Acciones
1	 	1	Ariel Solano	cable	ninguna	La misma	350	8959-6566	125-080286-0002T	 
2	 	2	JUAN MARTIN AGUIRRE	CABLE	NINGUNA	regina 2 c. al sur	200	5898-5554	122-081198-6662D	 
3	 	3	pedro gonzalez	cable	ninguan	ninguna	350	8958-8958	124-080286-0002M	 
4	 	4	maria gonzalez	cable	ninguna	ninguna	200	5854-5858	112-081078-0002D	 
5	 	5	weslin vargas Martinez	Tv Cable	barrio la picurria	de colegio regina mundi 3c al sur	200	87250616	121-180994-0000L	 
6	 	6	miriam tellez	Tv Cable	el manguito	contigo al pali	230	25128547	121-250565-0000G	 

Mostrando 1 a 6 de 6 entradas Anterior Siguiente

Fig.10

La Fig.11 muestra la venta de registro de un empleado para ello se solicita la cédula de la persona a registrar como empleado y se guardan sus datos para crear su usuario y contraseña, luego de dar clic en guardar los datos registrados se muestran con las acciones de modificar los datos del empleado o borrarlos si la persona ha dejado de trabajar para la empresa.

Registro de Empleados

Datos del empleado:

(*) Cédula del empleado

(*) Nombre del empleado

(*) Apellidos del empleado

(*) Dirección del empleado

(*) Ciudad del empleado

(*) Telefono

Mostrar 10 entradas Busqueda:

#	Cédula	Nombres	Apellidos	Dirección	Ciudad	Telefono	Acciones
1	102-011094-0005D	Frach José	Martinez Martinez	regina 7 c. al sur	juigalpa	5821-2548	<input type="button" value="✎"/> <input type="button" value="✖"/>
2	121-080289-0000D	Juan	Martinez	Cruz roja 2 c. al sur	Juigalpa	5854-5854	<input type="button" value="✎"/> <input type="button" value="✖"/>

Mostrando 1 a 2 de 2 entradas Anterior Siguiente

Fig.11

Para realizar un Pago se solicita como requisito el número del contrato del cliente como se muestra en la Fig.12, una vez que se obtiene el número se comprueban los datos del dueño del contrato para ellos se muestra el nombre dirección y el servicio con su respectivo precio, cuando los datos se comprueban se procede a realizar el registro del pago y se guarda en el sistema.

Registro de Pagos

(*) Número del Contrato

Mora

Observaciones

Datos del Cliente:

Nombre del Cliente

Dirección del Cliente

Descripción del servicio

Subtotal

Fig.12

Cuando se ha registrado el pago se procede a mostrar como se observa en la Fig.13

#	Fecha	Observación	Mora	Subtotal	Total	idcontrato	Acciones
1	7/12/2016 12:00:00 a. m.		20	350	370	3	
2	7/12/2016 12:00:00 a. m.		60	200	260	2	
3	12/12/2016 12:00:00 a. m.	retraso	30	350	380	1	
4	12/12/2016 12:00:00 a. m.	ninguna	0	350	350	1	
5	12/12/2016 12:00:00 a. m.	retraso	50	230	280	6	

Mostrando 1 a 5 de 5 entradas

Anterior 1 Siguiente

Fig.13

La Fig. 14 muestra un reporte de pago registrado en el sistema y listo para ser impreso como comprobante de que ya se realizó el pago respectivo del mes.

Reporte de Pago

Número de contrato: 6
 Nombre del Cliente: miriam
 Apellidos del Cliente: tellez

Detalles de pago

Fecha	Mora	Subtotal	Total	Observación
12/12/2016	50	230	280	retraso

Imprimir

Fig.14

La Fig.15 muestra la factura de pago lista para imprimirse y para ser entregada al cliente.

Imprimir

Total: 1 hoja de papel

Destino Microsoft Print to PDF

Páginas Todo
 p. ej. 1-5, 8, 11-13

Diseño

Color

+ Más opciones

Imprimir utilizando el cuadro de diálogo del sistema (Ctrl+Shift+P)

Reporte de Pago

Número de contrato: 6
 Nombre del Cliente: miriam
 Apellidos del Cliente: tellez

Detalles de pago

Fecha	Mora	Subtotal	Total	Observación
12/12/2016	50	230	280	retraso

Fig.15

Al dar clic en la ventana servicios muestra el sub menú de soporte técnico que al dar clic cargara la ventana que se muestra en la Fig.16

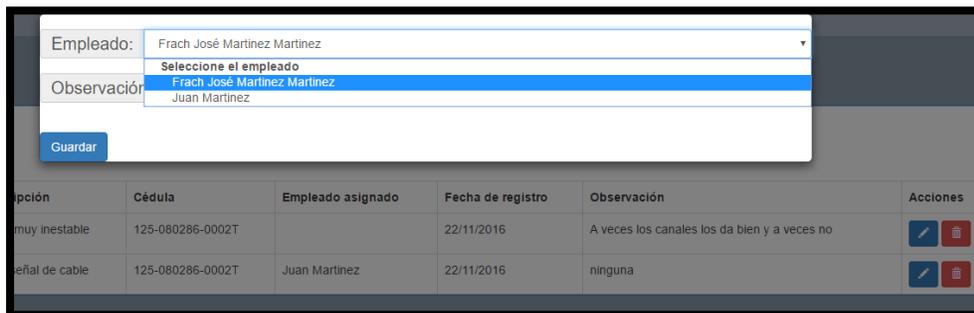


The screenshot shows a web application interface with a navigation menu at the top containing 'Inicio', 'Contratos', 'Empleados', 'Pagos', 'Servicios', and 'Reportes'. A 'Cerrar sesión' button is also visible. The main content area is titled 'Soporte técnico' and displays a table with the following data:

#	# contrato	Descripción	Cédula	Empleado asignado	Fecha de registro	Observación	Acciones
1	2	Señal muy inestable	125-080286-0002T		22/11/2016	A veces los canales los da bien y a veces no	 
2	3	Mala señal de cable	125-080286-0002T	Juan Martinez	22/11/2016	ninguna	 

Fig.16

Al dar clic en el botón azul de editar carga la Fig.17 que muestra la asignación de un empleado para dar solución a la avería reportada por el cliente.



The screenshot shows a modal form for editing a record. It features a dropdown menu for 'Empleado:' with the current selection 'Frach José Martínez Martínez'. Below it, a list of options is shown: 'Seleccione el empleado', 'Frach José Martínez Martínez', and 'Juan Martínez'. A 'Guardar' button is located at the bottom left of the modal. The background shows a dimmed version of the table from Fig.16.

Fig.17