



Universidad Central de Venezuela
Facultad de Ciencias
Escuela de Computación

**Desarrollo de una Solución Web para el
Proceso de Visitas Clínicas de una
Compañía Aseguradora**

Trabajo Especial de Grado

Presentado ante la Ilustre

Universidad Central de Venezuela

Por el Br: Argenis Jesús Chirinos Correa

CI.: V-22.671.694

Para optar al título de

Licenciado en Computación

Tutor:

Prof. Franky Uzcátegui Polo

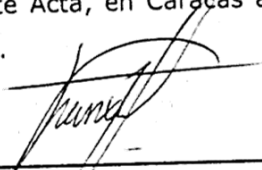
UNIVERSIDAD CENTRAL DE VENEZUELA
FACULTAD DE CIENCIAS
ESCUELA DE COMPUTACIÓN

ACTA


Quienes suscriben, miembros del jurado designado por el Consejo de la Escuela de Computación, para examinar el Trabajo Especial de Grado titulado "Desarrollo de una Solución Web para el Proceso de Visitas Clínicas de una Compañía Aseguradora" y presentado por el Bachiller: Argenis Jesús Chirinos Correa C.I.: 22.671.694, a los fines de optar al título de **Licenciado en Computación**, dejamos constancia de lo siguiente:

Leído como fue dicho trabajo, por cada uno de los miembros del jurado, se fijó el día 08 de Mayo de 2017, a las 11:00 A.M, para que el autor lo defendiera en forma pública, lo que este hizo en la Sala 01 de la Escuela de Computación, mediante una presentación oral de su contenido, luego de lo cual respondió a las preguntas formuladas. Finalizada la defensa pública del Trabajo Especial de Grado, el jurado decidió aprobar con la nota de 20 puntos.


En fe de lo cual se levanta la presente Acta, en Caracas a los ocho (08) días del mes de Mayo del año dos mil diecisiete (2017).



Prof. Franky Uzcátegui
(Tutor)



Prof. Mercy Ospina
(Jurado)



Prof. Franklin Sandoval
(Jurado)

Dedicatoria

Dedico este Trabajo Especial de Grado primero que todo al Dios Todopoderoso y a la Virgen Santísima, por darme la fortaleza espiritual para lograr todas mis metas.

A todo el grupo de profesores que día a día con el transcurrir del tiempo nos sirven de guía a nosotros los estudiantes, para lograr las metas trazadas.

A mi abuelo Manuel, quien aunque ahora está con Dios sé que le hubiese encantado verme cumplir parte de mis sueños.

A mi mami Yenis, a mi papá Argenis y a mi hermana Jennifer por apoyarme incondicionalmente.

Agradecimiento

Ante todo debo de agradecer al Dios Todopoderoso y a la Virgen Santísima en sus diferentes advocaciones, en particular a la Virgencita Del Valle por guiarme y darme la fortaleza en todo momento, para no decaer. Es difícil para mí resumir en tan solo unas líneas las palabras de agradecimiento que les tendría que dar a cada una de las personas que han colaborado de una u otra manera en la realización de tan importante meta. Lo que expreso en ésta página siempre se quedará pequeño, pues son innumerables las gracias que les tengo que dar a todas las personas que contribuyeron en mi formación como persona y profesional.

Agradezco a la Universidad Central De Venezuela, por haber sido mi segunda casa durante mis estudios universitarios. Agradezco al profesor Franky Uzcátegui por su guía y orientación en la realización de esta tesis. Agradezco a la profesora Mercy Ospina y al Profesor Franklin Sandoval por sus valiosas sugerencias; y a todos los profesores que me guiaron durante mis estudios en la universidad pues me ayudaron a cumplir una de mis metas.

A mi mamá Yenis, a mi papá Argenis y a mi hermana Jennifer por su apoyo incondicional. A mi abuelita por sus ocurrencias. A mi abuelito Manuel quién por siempre vivirá en mi corazón.

Universidad Central de Venezuela

Facultad de Ciencias

Escuela de Computación

Centro de Investigación de Sistemas de Información

**Desarrollo de una Solución Web para el Proceso de Visitas Clínicas de una
Compañía Aseguradora**

Autor: Br. Argenis Jesús Chirinos Correa. Cl.: V-22.671.694

Tutor: Prof. Franky Uzcátegui Polo

Resumen

El presente Trabajo Especial de Grado tiene como objetivo desarrollar una solución web que permita optimizar la gestión del proceso de Visitas Clínicas llevado a cabo por una compañía aseguradora. A través de esta solución se ofrece un conjunto de funcionalidades que permite al personal de la compañía envuelto en este proceso realizar sus tareas y llevar control sobre ellas de forma más rápida y eficiente.

Para el desarrollo de esta solución se utilizaron diversas tecnologías de desarrollo web como HTML, CSS, JavaScript y Node.JS. El desarrollo de la misma fue guiado a través de la Metodología de Desarrollo SCRUM aplicando adaptaciones de acuerdo a las necesidades de la misma. Se utilizó SCRUM ya que es una Metodología de Desarrollo ágil de rápida adaptación, flexible y eficaz que permite agregar valor significativo de forma rápida en todo proyecto.

Palabras Claves: Solución Web, Aplicación Web, Metodología de Desarrollo, Ágil, SCRUM, Compañía Aseguradora.

Índice

Resumen.....	iv
Índice.....	v
Índice de Ilustraciones.....	viii
Índice de Tablas.....	xi
Introducción.....	1
Problema de la Investigación	3
1.1 Situación Actual	3
1.2 Planteamiento del Problema	3
1.3 Objetivo General	11
1.4 Objetivos Específicos.....	11
1.5 Justificación	11
1.6 Alcance	12
Marco Conceptual	13
2.1 Sistema de Información	13
2.1.1 Sistemas de Procesamiento de Transacciones.....	14
2.1.2 Sistema de Información Gerencial	14
2.1.3 Sistemas de Soporte de Decisiones	14
2.1.4 Sistemas de Apoyo a Ejecutivos	15
2.2 Aplicación web	15
2.3 Arquitectura Cliente – Servidor	16
2.3.1 Separación de funciones.....	18
2.4 Tecnologías para desarrollo del cliente	19
2.4.1 HTML	20
2.4.2 CSS.....	20
2.4.3 JavaScript	21
2.4.4 jQuery	21

2.4.5 AJAX	22
2.4.6 AngularJS.....	23
2.4.7 Bootstrap.....	23
2.5 Tecnologías para el desarrollo del servidor	24
2.5.1 Bases de datos	24
2.5.2 Sistemas gestores de bases de datos	24
2.5.3 PostgreSQL.....	24
2.5.4 Frameworks para el desarrollo de aplicaciones web.....	26
2.5.5 Patrón de diseño Modelo – Vista – Controlador.....	26
2.5.6 Express	28
2.5.7 Servicios web	29
2.5.8 JSON	30
2.5.9 REST	30
2.6 Aplicaciones de una sola página.....	31
Marco Metodológico	34
3.1 Metodología de Desarrollo	34
3.2 Metodologías Tradicionales	34
3.3 Metodologías Ágiles.....	35
3.4 SCRUM.....	37
3.4.1 Flujo de un proyecto SCRUM.....	39
3.4.2 Bloque de tiempo asignado.....	40
3.4.3 Bloques de tiempo de SCRUM	41
3.4.4 Roles de un proyecto SCRUM	44
3.4.5 Roles centrales	45
3.4.6 Roles no centrales	46
3.4.7 Terminología asociada a la Metodología SCRUM	47
Marco Aplicativo	49
4.1 Descripción general de la solución	49
4.2 Requerimientos del sistema.....	53
4.3 Perfiles de usuario	54
4.4 Descripción del flujo asociado a la solución.....	56

4.5	Aplicación de la metodología SCRUM	58
4.5.1	Lista de objetivos (Pila del Producto)	58
4.5.2	Lista de tareas por iteración (Pila del Sprint).....	58
4.6	Análisis del modelo de datos y definición.....	62
4.6.1	Definición de las tablas de la base de datos	62
4.7	Servicios web.....	65
4.8	Descripción de los módulos del sistema y sus interfaces	68
4.8.1	Descripción de los módulos	68
4.8.2	Interfaces del sistema web.....	71
4.9	Fase de pruebas	94
4.9.1	Pruebas funcionales.....	94
4.9.2	Pruebas de aceptación de usuario.....	100
	Conclusiones y Recomendaciones	104
	Bibliografía	106

Índice de Ilustraciones

Figura 1. Flujo del proceso actual de Visita Clínica. Fuente: Adaptación del manual procedimientos de la compañía aseguradora.	5
Figura 2. Flujo de la solución. Fuente: Elaboración propia.....	9
Figura 3. Arquitectura implementada en el sistema web. Fuente: Elaboración propia..	10
Figura 4. Arquitectura Cliente – Servidor [11].	16
Figura 5. Separación de funciones [2].....	19
Figura 6. Modelo - Vista – Controlador [14].....	27
Figura 7. Ciclo de vida de una aplicación web tradicional y una SPA [23].	33
Figura 8. Flujo de SCRUM para un Sprint [28].....	39
Figura 9. Duración de los bloques de tiempo para las reuniones de SCRUM [28].....	44
Figura 10. Arquitectura implementada en el sistema web. Fuente: Elaboración propia	50
Figura 11. Tecnologías utilizadas para implementar la arquitectura. Fuente: Elaboración propia	52
Figura 12. Flujo de la solución. Fuente: Elaboración propia.....	57
Figura 13. Tablas de la base de datos. Fuente: Elaboración propia	63
Figura 14. Módulos del Sistema web. Fuente: Elaboración propia	69
Figura 15. Inicio de sesión. Fuente: Captura de pantalla	71
Figura 16. Listado de cartas avales. Fuente: Captura de pantalla	72
Figura 17. Búsqueda de cartas avales. Fuente: Captura de pantalla.....	72
Figura 18. Detalles de carta aval. Fuente: Captura de pantalla	73
Figura 19. Ingreso de fecha límite para realizar visita clínica. Fuente: Captura de pantalla.....	74
Figura 20. Notificación de visita clínica solicitada. Fuente: Captura de pantalla	74
Figura 21. Listado de visitas clínicas. Fuente: Captura de pantalla	75
Figura 22. Búsqueda de visitas clínicas. Fuente: Captura de pantalla	76
Figura 23. Detalles de visita clínica. Fuente: Captura de pantalla.....	76

Índice de Ilustraciones

Figura 24. Listado de visitadores clínicos para asignar visita. Fuente: Captura de pantalla.....	77
Figura 25. Notificación de visita asignada. Fuente: Captura de pantalla.....	78
Figura 26. Información para realizar visita clínica. Fuente: Captura de pantalla	79
Figura 27. Información del presupuesto original. Fuente: Captura de pantalla	80
Figura 28. Información de las modificaciones sobre el presupuesto. Fuente: Captura de pantalla.....	81
Figura 29. Edición de gasto del presupuesto. Fuente: Captura de pantalla	81
Figura 30. Gastos del presupuesto modificado. Fuente: Captura de pantalla.....	82
Figura 31. Carga de respuestas de la encuesta. Fuente: Captura de pantalla	82
Figura 32. Carga de documentos digitales. Fuente: Captura de pantalla.....	83
Figura 33. Notificación de visita clínica finalizada. Fuente: Captura de pantalla.....	83
Figura 34. Toma de decision sobre ordenar revisión o dar finalizada la visita clínica. Fuente: Captura de pantalla.....	84
Figura 35. Notificación de visita clínica dada por finalizada. Fuente: Captura de pantalla	85
Figura 36. Cambio de nombre de usuario. Fuente: Captura de pantalla.....	85
Figura 37. Cambio de contraseña. Fuente: Captura de pantalla	86
Figura 38. Listado de personas. Fuente: Captura de pantalla.....	87
Figura 39. Búsqueda de personas. Fuente: Captura de pantalla	87
Figura 40. Detalles de persona. Fuente: Captura de pantalla	88
Figura 41. Edición de datos de una persona. Fuente: Captura de pantalla.....	89
Figura 42. Listado de clínicas. Fuente: Captura de pantalla	89
Figura 43. Búsqueda de clínicas. Fuente: Captura de pantalla.....	90
Figura 44. Detalles de clínicas. Fuente: Captura de pantalla	90
Figura 45. Edición de datos de una clínica. Fuente: Captura de pantalla	91
Figura 46. Listado de presupuestos. Fuente: Captura de pantalla.....	91
Figura 47. Búsqueda de presupuestos. Fuente: Captura de pantalla	92

Índice de Ilustraciones

Figura 48. Detalles de presupuesto. Fuente: Captura de pantalla	92
Figura 49. Edición de datos del presupuesto (administrador). Fuente: Captura de pantalla.....	93
Figura 50. Carga de datos a través de .txt. Fuente: Captura de pantalla	94
Figura 51. Resultados de la pregunta N° 1. Fuente: Pruebas de aceptación con usuarios finales	101
Figura 52. Resultados de la pregunta N° 2. Fuente: Pruebas de aceptación con usuarios finales	101
Figura 53. Resultados de la pregunta N° 3. Fuente: Pruebas de aceptación con usuarios finales	101
Figura 54. Resultados de la pregunta N° 4. Fuente: Pruebas de aceptación con usuarios finales	102
Figura 55. Resultados de la pregunta N° 5. Fuente: Pruebas de aceptación con usuarios finales	102
Figura 56. Resultados de la pregunta N° 6. Fuente: Pruebas de aceptación con usuarios finales	102
Figura 57. Resultados de la pregunta N° 7. Fuente: Pruebas de aceptación con usuarios finales	103

Índice de Tablas

Tabla 1. Pila del producto. Fuente: Elaboración propia.....	59
Tabla 2. Instrumento de evaluación para las pruebas funcionales. Fuente: Elaboración propia	95
Tabla 3. Primer ciclo de pruebas funcionales para el módulo de Carta Aval. Fuente: Desarrollo de pruebas con usuarios finales	95
Tabla 4. Primer ciclo de pruebas funcionales sobre el módulo de Solicitud de Visita Clínica. Fuente: Desarrollo de pruebas con usuarios finales	96
Tabla 5. Segundo ciclo de pruebas funcionales sobre el módulo de Solicitud de Visita Clínica. Fuente: Desarrollo de pruebas con usuarios finales	97
Tabla 6. Primer ciclo de pruebas funcionales sobre el módulo de Personas. Fuente: Desarrollo de pruebas con usuarios finales	97
Tabla 7. Primer ciclo de pruebas funcionales sobre el módulo de Clínicas. Fuente: Desarrollo de pruebas con usuarios finales	98
Tabla 8. Primer ciclo de pruebas funcionales sobre el módulo de Presupuestos. Fuente: Desarrollo de pruebas con usuarios finales	98
Tabla 9. Primer ciclo de pruebas funcionales sobre el módulo de Datos. Fuente: Desarrollo de pruebas con usuarios finales	99
Tabla 10. Segundo ciclo de pruebas funcionales sobre el módulo de Datos. Fuente: Desarrollo de pruebas con usuarios finales	99
Tabla 11. Encuesta de Aceptación. Fuente: Elaboración propia	100

Introducción

La tecnología es un elemento vital para sustentar la operación de las organizaciones. El impacto que produce, les permite minimizar sus costos y optimizar los objetivos enfocados en la operación y el servicio al cliente. En particular, las empresas aseguradoras han encontrado en la tecnología un aliado muy importante para controlar sus múltiples y variados procesos, cuya complejidad es innegable.

Considerando la oportunidad que tienen las empresas aseguradoras de aplicar la tecnología en sus procesos de negocio con la intención de mejorarlos, el presente Trabajo Especial de Grado se centró en agilizar la gestión de un proceso conocido como Visita Clínica, el cual es llevado a cabo por una compañía aseguradora en la cobertura de su seguro de hospitalización o de salud. Este proceso tiene como objetivo verificar y determinar con exactitud el monto total de gastos a cubrir por parte de la compañía aseguradora ante el tratamiento realizado por el asegurado. La gestión que se mantenía sobre este proceso por parte de la compañía aseguradora generaba varios inconvenientes, los cuales buscaban ser mejorados por la misma a través del desarrollo de una Aplicación Web. En tal sentido, el objetivo principal de este Trabajo Especial de Grado consistió en el desarrollo de una Aplicación Web que permitiera solventar los inconvenientes que presentaba la gestión de este proceso, ofreciendo un conjunto de módulos funcionales que permitan al personal de la compañía involucrado en este proceso realizar sus tareas de forma más rápida y eficiente. En el desarrollo de esta Aplicación Web se utilizaron tecnologías como HTML, CSS, JavaScript, Angular.JS, Node.JS y PostgreSQL. Además, el desarrollo fue guiado a través de la Metodología de Desarrollo ágil SCRUM, dando la posibilidad de incorporar rápidamente nuevos requerimientos y módulos funcionales que surgieran a lo largo del desarrollo de la aplicación.

Para profundizar en el desarrollo de la aplicación anteriormente descrita, el presente documento de Trabajo Especial de Grado se encuentra dividido en cuatro capítulos que se detallaran a continuación:

- 1. Capítulo I (Problema de la Investigación):** en este capítulo se expone la problemática que se quiere solventar a través del presente Trabajo Especial de Grado y la solución que se desarrolla para solventarla. De igual forma se presenta el objetivo general, objetivos específicos, justificación y alcance de este trabajo.
- 2. Capítulo II (Marco Conceptual):** en este capítulo se presentan conceptos relacionados con las Sistemas de Información, Bases de Datos, Arquitectura Cliente – Servidor, Aplicaciones Web y las principales tecnologías utilizadas para el desarrollo de la solución.
- 3. Capítulo III (Marco Metodológico):** en este capítulo se presenta la Metodología de Desarrollo SCRUM, la cual fue utilizada para guiar el desarrollo de la Aplicación Web sobre la cual se centró el Trabajo Especial de Grado.
- 4. Capítulo IV (Marco Aplicativo):** en este capítulo se presenta la Metodología de Desarrollo SCRUM adaptada a las necesidades de la Aplicación Web desarrollada, detallando las iteraciones llevadas a cabo.

Para finalizar se presentan conclusiones, recomendaciones y referencias bibliográficas del presente Trabajo Especial de Grado.

Problema de la Investigación

1.1 Situación Actual

Las compañías aseguradoras que ofrecen entre sus servicios una póliza de hospitalización o de salud, deben expedir a los asegurados que vayan a hacer uso de su póliza de salud, un documento conocido como carta aval, el cual es garantía del cubrimiento del monto que en esta carta se especifica. La compañía aseguradora tramita la carta aval en base al presupuesto del tratamiento que se va a practicar el asegurado. Una vez que el asegurado se ha practicado el tratamiento se lleva a cabo un proceso conocido, en la compañía aseguradora sobre la cual se centra el presente estudio, como Visita Clínica. Este proceso es realizado entre la compañía aseguradora y la institución prestadora del servicio (generalmente una clínica) y consiste en verificar el total de gastos que debe cubrir la compañía aseguradora en cuanto al tratamiento del asegurado. El proceso requiere que el personal de la compañía involucrado en el mismo, tenga que mediar con varios documentos, entre ellos la carta aval y el presupuesto del tratamiento en cuestión, para así totalizar los gastos según los cambios que hayan surgido en los mismos luego de practicado el tratamiento.

1.2 Planteamiento del Problema

La gestión del proceso de Visita Clínica llevada a cabo por parte de la compañía aseguradora involucra la mediación con una gran cantidad de documentos en físico que pueden ser manipulados de forma digital, sin embargo, la compañía aseguradora no contaba con un sistema apropiado para acceder digitalmente a estos documentos. Además, se mantenía un control sobre la distribución de estos documentos a lo largo

del proceso de Visita Clínica a través de Hojas de Cálculo en Excel, y al ser una gran cantidad de documentos, resultaba difícil mantener los controles requeridos por la compañía. Manipular los documentos directamente en físico y llevar controles sobre la distribución de los mismos a lo largo del proceso de Visita Clínica generaban retrasos considerables en este proceso. Por esta razón, la compañía aseguradora tomó la iniciativa de desarrollar un sistema que permitiera solventar esta problemática. El objetivo de esta iniciativa es desarrollar un sistema que optimice la gestión actual del proceso de Visita Clínica, mejorando así la ejecución de este proceso y la eficiencia del personal de la compañía aseguradora involucrado en el mismo.

A continuación se describe más detalladamente cómo inicia el proceso de Visita clínica y el flujo de tareas que este involucra. El mismo puede ser visualizado en la figura 1:

Cuando alguno de los asegurados de la compañía aseguradora decide hacer uso de la póliza de salud a la cual se encuentra suscrito, debe entregar a la clínica en la cual se hará el tratamiento, una carta aval por parte de la compañía aseguradora que garantice el cubrimiento de los gastos amparados del tratamiento. El asegurado debe solicitar esta carta a la compañía aseguradora y para ello debe entregar ciertos recaudos, entre ellos la evaluación médica del paciente y el presupuesto de gastos de hospitalización debidamente desglosado. Una vez que la carta aval es emitida, esta es entregada por el asegurado a la clínica, la cual posteriormente se comunicará con la compañía aseguradora para solicitar la activación de la carta aval. Cuando la clínica se comunica con la compañía aseguradora para activar la carta aval, un analista de seguimiento de carta aval se encarga de solicitar una visita clínica para dicho caso.

La solicitud hecha por el analista de seguimiento de carta aval está conformada por una planilla, la cual es llenada por este analista con los datos requeridos del asegurado, además de la carta aval y el presupuesto de gastos del tratamiento

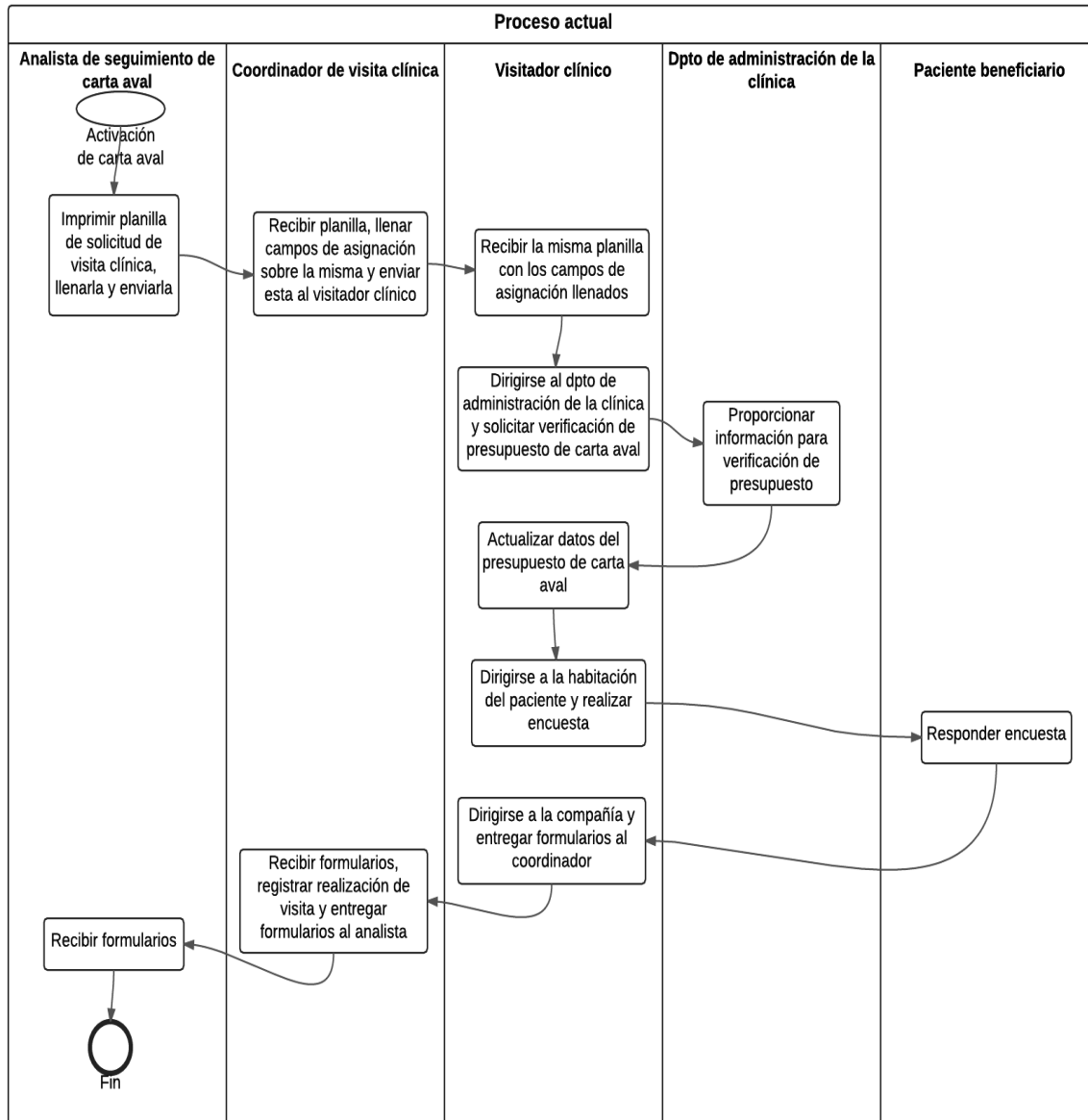


Figura 1. Flujo del proceso actual de Visita Clínica. Fuente: Adaptación del manual procedimientos de la compañía aseguradora.

entregado por el asegurado. Esta solicitud es enviada al coordinador de visitas clínicas, el cual es encargado de asignar dicha solicitud a algún visitador clínico de la compañía. Para ello, debe consultar un registro de visitadores clínicos y asignar dicha visita clínica a algún visitador que se encuentre disponible. Esta asignación se hace llenando ciertos campos en la misma planilla que recibe con la solicitud de visita, con los datos del visitador clínico asignado y luego se envía esta asignación al visitador junto con el resto de la información que conforma la solicitud de visita clínica. Posteriormente, esta asignación es recibida por el visitador, el cual debe dirigirse al departamento de presupuesto de la clínica para verificar el presupuesto entregado en solicitud de la carta aval, y luego debe realizar una encuesta al asegurado, con la intención de recaudar información acerca de la calidad de los servicios prestados por la clínica en cuestión. Finalmente, el visitador debe ir a la compañía de seguros y hacer entrega de la documentación al coordinador de visitas, el cual se encargará de registrar la realización de la visita clínica y posteriormente entregar la documentación de la visita al analista de seguimiento de carta aval, el cual a su vez, en base a la información recibida, prosigue con el protocolo llevado a cabo en la compañía aseguradora.

Esta gestión de visitas clínicas presentaba una serie de inconvenientes que podían ser mejorados. A continuación se destacan los principales:

- Las solicitudes de visitas clínicas y las asignaciones de las mismas, se llevan a cabo a través de documentos en físico, lo cual genera contratiempos en cuanto al traspaso de estos documentos y todo el trabajo manual que conlleva trabajar con ellos (imprimir las planillas, llenarlas, etc.).
- El coordinador de visitas lleva un control manual de los visitadores clínicos que se encuentran disponibles y los casos a los que se encuentran asignados. Esto dificulta la coordinación de las visitas ya que no se cuenta con un control

adecuado y detallado de los visitantes clínicos que se encuentran disponibles y de las visitas que tienen asignadas.

- Los presupuestos de gastos de tratamientos y las cartas avales se mantienen registradas en las bases de datos de la compañía aseguradora. Sin embargo, del personal de la compañía aseguradora que se ve envuelto en este proceso de visita clínica, el único que cuenta con un sistema para consultar esta información, es el analista de seguimiento de carta aval. Por esta razón es que los documentos requeridos para la visita clínica son impresos por este analista y luego se envían al coordinador de visitas, el cual posteriormente los envía al visitador clínico.
- Después de que el visitador clínico verifica el presupuesto y realiza la encuesta, debe regresar a la compañía de seguros para hacer entrega de la documentación. El retorno del visitador clínico a la compañía para entregar la información requerida, es algo que suma una cantidad considerable de tiempo al proceso de visitas clínicas y es algo que puede ser evitado.

La compañía aseguradora posee sucursales en diferentes estados nacionales del país. En cada una de estas sucursales se encuentran diferentes analistas de seguimiento de carta aval, coordinadores de visitas clínicas y visitadores clínicos. De acuerdo a las reglas de negocio de la compañía aseguradora se deben tener en cuenta las siguientes consideraciones:

- El analista de seguimiento de carta aval coincide con la ubicación geográfica donde fue tramitada la carta aval, la cual no necesariamente coincide con la ubicación geográfica de la institución prestadora del servicio donde el asegurado se practicará el tratamiento.

- El coordinador de visitas clínicas coincide con la ubicación geográfica de la institución prestadora de servicio. Este coordinador se encarga de distribuir las visitas clínicas dentro de un conjunto de visitadores clínicos encargado de coordinar. Estos visitadores clínicos coinciden también con la ubicación geográfica de la institución prestadora del servicio.

Teniendo en cuenta estas reglas de negocio, el proceso de visita clínica se vuelve más complejo en los casos en que la ubicación geográfica donde fue emitida la carta aval no coincide con la ubicación de la institución prestadora del servicio, ya que se deben llenar las planillas en físico, escanearlas y enviarlas por correo electrónico para que sea recibida por el personal de la sucursal que coincide con la ubicación de la institución prestadora del servicio, y esto a su vez genera más contratiempos en el proceso de Visita Clínica.

En vista de la problemática existente se desarrolló una solución a través de la cual se diera acceso digital a los documentos requeridos, se mantuviese automáticamente el control sobre las visitas asignadas y el personal involucrado pudiera comunicarse de forma transparente mediante la misma. En la figura 2 se presenta el flujo del proceso de visita una vez implantada la solución. Al proceso de incorpora el componente tecnológico (Sistema Web que ofrecerá las funcionalidades requeridas y con el cual el personal de la compañía aseguradora interactuará a lo largo del proceso de visita.

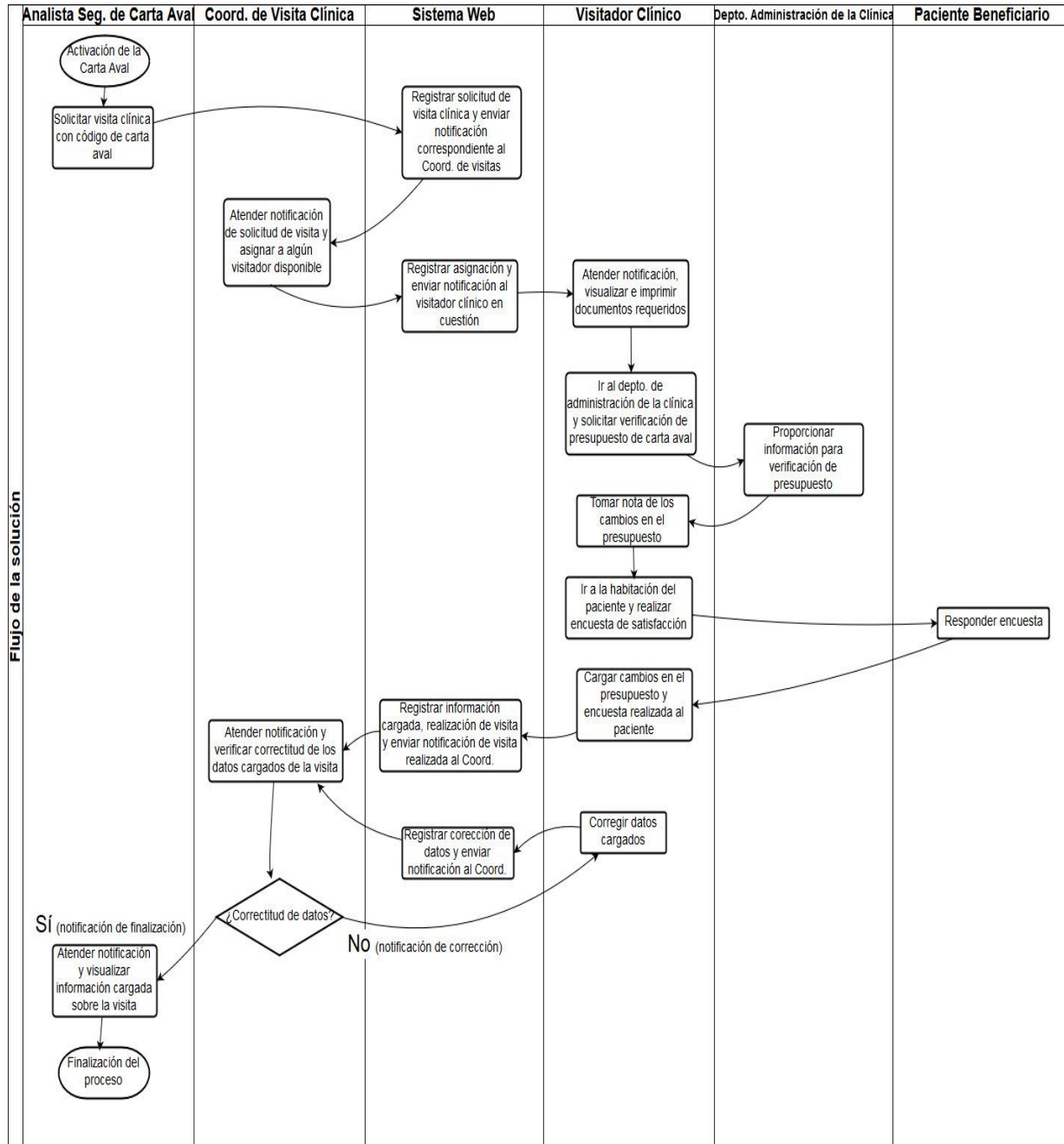


Figura 2. Flujo de la solución. Fuente: Elaboración propia

La arquitectura de la solución es presentada en la figura 3 y la misma fue implementada teniendo en cuenta el conjunto de tecnologías utilizadas en la compañía aseguradora y cuáles resultaban más convenientes para el desarrollo de este sistema.

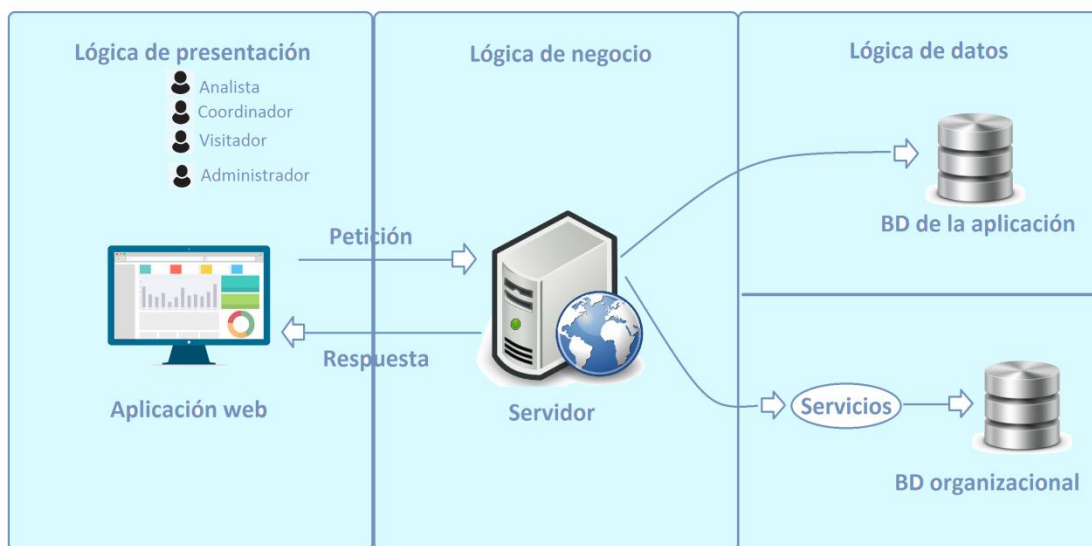


Figura 3. Arquitectura implementada en el sistema web. Fuente: Elaboración propia

La arquitectura consta de tres partes:

- Lógica de presentación: representa la parte visual del sistema (interfaces de usuario) a través de las cuales los usuarios interactúan con el sistema.
- Lógica de negocio: representa la implementación de las reglas de negocio que sigue el proceso de Visita Clínica.
- Lógica de datos: representanta todos los datos que necesita el sistema para su funcionamiento.

En el Marco Aplicativo se detalla el flujo de la solución desarrollada y la arquitectura de la misma.

1.3 Objetivo General

Desarrollar una solución web para el proceso de visitas clínicas de la compañía aseguradora.

1.4 Objetivos Específicos

A fin de cumplir el objetivo general, se plantearon los siguientes objetivos específicos:

- Identificar los requerimientos funcionales y no funcionales del sistema web.
- Diseñar el modelo de base de datos para el almacenamiento de los datos requeridos, las interfaces de usuario del sistema y servicios web.
- Implementar el modelo de base de datos, interfaces de usuario y servicios web.
- Desarrollar funcionalidades de la aplicación web.
- Realizar pruebas funcionales y de aceptación de usuarios.

1.5 Justificación

Para responder a la problemática encontrada se planteó desarrollar una aplicación web donde se lleva la gestión completa del proceso de Visita Clínica de forma óptima. A través de esta se podrá mantener automáticamente los controles que se requieren, brindando también acceso digital a los documentos que se necesitan en este proceso. De esta forma el personal encargado de atender el proceso tendrá que mediar en mucha menor proporción con los documentos en físico y no se verán obligados a llevar los controles de hojas de cálculo.

Además de agilizar el proceso de Visita Clínica, este sistema web sirve como base para futuros proyectos en la compañía aseguradora que requieran operar con datos generados en este proceso. Por lo tanto, resulta conveniente el desarrollo de un sistema que permita resolver la problemática actual correspondiente a la Visita Clínica y que tiene la posibilidad de alimentar futuros sistemas de la compañía de seguros.

1.6 Alcance

La aplicación web desarrollada debe ser capaz de:

- Ofrecer funcionalidades específicas para cada rol de usuario en el sistema.
- Enviar notificaciones de correo electrónico a lo largo del proceso.
- Ofrecer visualización del presupuesto, carta aval y encuesta de satisfacción.
- Gestionar datos de personas, usuarios, cartas avales, presupuestos y otros datos cargados en el sistema.
- Distribuir las visitas clínicas según la localización de los usuarios, origen de cartas avales y presupuestos.
- Cargar cambios en el presupuesto según la información obtenida durante la visita clínica.

Marco Conceptual

En este capítulo se presentan las bases teóricas para el desarrollo del presente Trabajo Especial de Grado. Entre estas bases teóricas se encuentran conceptos asociados a Sistemas de Información, Bases de Datos, Aplicación Web, Arquitectura Cliente – Servidor y diversas tecnologías utilizadas para el desarrollo de cada parte en esta arquitectura en el contexto de las Aplicaciones Web.

2.1 Sistema de Información

Un sistema de información se puede definir técnicamente como un conjunto de componentes interrelacionados que recolectan (o recuperan), procesan, almacenan y distribuyen información para apoyar los procesos de toma de decisiones y de control en una organización [1].

Los sistemas de información contienen información sobre personas, lugares y cosas importantes dentro de la organización, o en el entorno que la rodea [1]. Por información se hace referencia a los datos que se han modelado en una forma significativa y útil para los seres humanos. Mientras que los datos son flujos de elementos en bruto que representan los eventos que ocurren en las organizaciones o en el entorno físico antes de ordenarlos e interpretarlos en una forma que las personas puedan comprender y usar.

Una empresa de negocios tiene sistemas para dar soporte a los distintos grupos de niveles de administración y estos sistemas incluyen sistemas de procesamiento de transacciones (TPS), sistemas de información gerencial (MIS), sistemas de soporte de decisiones (DSS) y sistemas de apoyo a ejecutivos (ESS) [1].

2.1.1 Sistemas de Procesamiento de Transacciones

Un Sistema de Procesamiento de Transacciones (TPS) es un sistema computarizado que efectúa y registra las transacciones diarias de rutina necesarias para realizar negocios. El principal propósito de los sistemas de este tipo de sistemas es responder a las preguntas de rutina y rastrear el flujo de transacciones por toda la organización.

2.1.2 Sistema de Información Gerencial

Los Sistemas de Información Gerencial (MIS) proveen reportes sobre el desempeño actual de la organización. Esta información se utiliza para supervisar y controlar la empresa, además de predecir su desempeño en el futuro. Los MIS sintetizan e informan sobre las operaciones básicas de la compañía mediante el uso de datos suministrados por los sistemas de procesamiento de transacciones.

2.1.3 Sistemas de Soporte de Decisiones

Los sistemas de Soporte de Decisiones (DSS) brindan apoyo a la toma de decisiones que no es rutinaria. Se enfocan en problemas que son únicos y cambian con rapidez, para los cuales el proceso para llegar a una solución tal vez no está completamente definido de antemano.

2.1.4 Sistemas de Apoyo a Ejecutivos

Los Sistemas de Apoyo a Ejecutivos (ESS) se utilizan en la toma de decisiones no rutinarias que requieren de juicio, evaluación y perspectiva, debido a que no hay un procedimiento acordado de antemano para llegar a una solución. Los ESS presentan gráficos y datos de muchas fuentes a través de una interfaz sencilla de manejar para los gerentes del nivel superior.

2.2 Aplicación web

Una aplicación web se puede definir como una aplicación en la cual el usuario por medio de un navegador web realiza peticiones a una aplicación remota accesible a través de Internet (o a través de una Intranet) y que recibe una respuesta que se muestra en el propio navegador [2].

Este tipo de aplicaciones han tenido mucho éxito debido a lo fácil que resulta usar un navegador web como cliente ligero [3], esto además implica otras ventajas como la independencia del sistema operativo que se utilice. Otra razón de su éxito es la facilidad para actualizarlas y mantenerlas, ya que no requiere la distribución, instalación y actualización en todos los usuarios que utilicen la aplicación, sino que basta actualizar la aplicación en el servidor para contar con nuevas versiones del sistema.

Las aplicaciones web se encuadran dentro de las arquitecturas cliente/servidor: un ordenador solicita servicios (el cliente) y otro está a la espera de recibir solicitudes y las responde (el servidor) [2].

2.3 Arquitectura Cliente – Servidor

La arquitectura cliente - servidor es un modelo de aplicación distribuida en el que las tareas se distribuyen entre dos partes. Una parte son los proveedores de recursos o servicios, conocidos como servidores, y la otra parte son los demandantes de estos recursos o servicios, conocidos como clientes [3].

En esta arquitectura los recursos y servicios se centralizan en el servidor al cual los clientes se encuentran conectados. Para que los clientes puedan acceder a estos recursos o servicios, deben realizar peticiones, las cuales serán recibidas y procesadas por el servidor. La comunicación entre ambas partes se lleva a cabo a través de algún protocolo de comunicación, el cual suele ser el protocolo HTTP.

En la figura 4 se describe como es la interacción entre ambas partes en la arquitectura cliente - servidor.

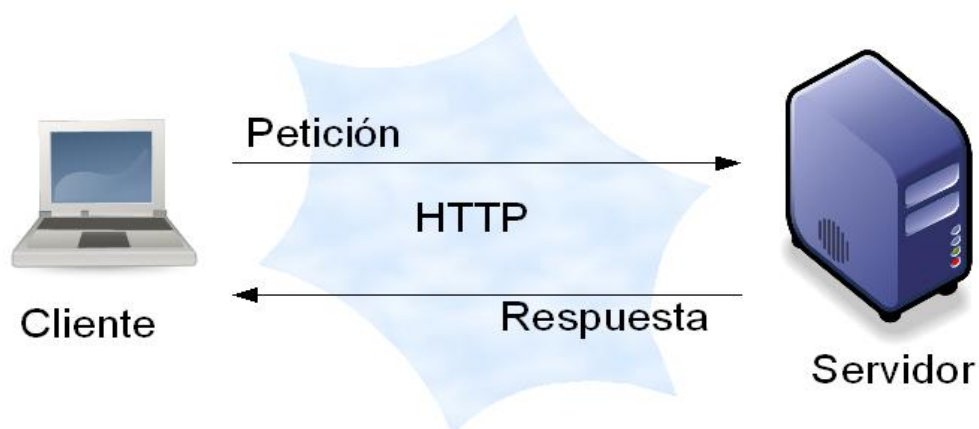


Figura 4. Arquitectura Cliente – Servidor [11].

Esta arquitectura implica la existencia de una relación entre procesos que solicitan servicios (clientes) y procesos que responden a estos servicios (servidores). Estos dos tipos de procesos pueden ejecutarse en el mismo procesador o en distintos [2].

Entre las características de la arquitectura cliente servidor se encuentran [3]:

- El remitente de una petición es conocido como cliente. Entre sus características se encuentran:
 - Inicia las solicitudes o peticiones. Tiene el papel de dispositivo maestro.
 - Generalmente, interactúa con el usuario final a través de una interfaz gráfica de usuario.
- Al receptor de una petición enviada por el cliente es conocido como servidor. Entre sus características se encuentran:
 - Está a la espera de las peticiones de los clientes.
 - Luego de recibir una petición, la procesa y posteriormente envía una respuesta al cliente.

Entre las ventajas de la arquitectura Cliente – Servidor se encuentran:

- La principal ventaja de esta arquitectura es que facilita la separación de las funciones según su servicio, permitiendo situar cada función en la plataforma más adecuada para su ejecución.
- Es posible aumentar y mejorar la capacidad de los clientes y servidores de forma independiente (escalabilidad).
- Al estar separadas las funciones y responsabilidades entre cliente y servidor, resulta cómodo realizar cambios en alguna de las partes sin necesidad de que la otra se vea afectada (fácil mantenimiento).

Entre las desventajas de la arquitectura Cliente – Servidor se encuentran:

- Cuando el servidor recibe una gran cantidad de peticiones de forma simultanea por parte de los clientes, el rendimiento y desempeño del sistema suele verse afectado, ya que el servidor debe satisfacer cada una de estas peticiones. Esto es conocido como sobrecarga.

2.3.1 Separación de funciones

La arquitectura cliente/servidor permite la separación de funciones en tres niveles [2], tal como se muestra en la Figura 5:

- **Lógica de presentación o cliente:** 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.
- **Lógica de negocio o procesamiento de la 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 que tiene que cumplir la aplicación y enviar el resultado del procesamiento al nivel de presentación.

- Lógica de datos o gestión de los 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.



Figura 5. Separación de funciones [2].

En el contexto de las aplicaciones web, el cliente es donde se encuentra el usuario final haciendo uso de la aplicación, y estaría representado por un navegador web (como Mozilla Firefox o Google Chrome) a través del cual se presenta la información. El cliente interactúa con el servidor, que es donde residen realmente los datos, reglas y lógica de la aplicación.

2.4 Tecnologías para desarrollo del cliente

Al momento de desarrollar una aplicación web, se deben tener en cuenta las tecnologías existentes que se pueden utilizar para desarrollar la parte del cliente y la parte del servidor. Las tecnologías para el desarrollo del cliente son independientes del servidor y son ejecutadas directamente por este, es decir, el navegador web.

Entre las tecnologías existentes para el desarrollo del cliente se encuentran:

2.4.1 HTML

HTML, siglas de HyperText Markup Language (lenguaje de marcas de hipertexto), es un lenguaje de marcado estándar utilizado para la elaboración de páginas web, definiendo su estructura básica y contenido (texto, imágenes, videos, entre otros) [4].

HTML es un estándar reconocido a nivel global y cuyas normas define un organismo llamado World Wide Web Consortium, más conocido como W3C. Como se trata de un estándar reconocido por todas las empresas relacionadas con el mundo de Internet, una misma página HTML se visualiza de forma muy similar en cualquier navegador de cualquier sistema operativo.

W3C define HTML como "un lenguaje reconocido universalmente y que permite publicar información de forma global". Desde su creación, el lenguaje HTML ha pasado de ser un lenguaje utilizado exclusivamente para crear documentos electrónicos a ser un lenguaje que se utiliza en muchas aplicaciones electrónicas como buscadores, tiendas online y banca electrónica [4].

2.4.2 CSS

CSS (cascading style sheets) es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML [5] y además, es una buena forma de separar el contenido y la presentación del mismo.

Separar la definición de los contenidos y la definición de su aspecto presenta numerosas ventajas, ya que obliga a crear documentos HTML bien de nidos y reduce la complejidad de su mantenimiento.

En la creación de una página web, se utiliza primeramente el lenguaje HTML para definir los contenidos, es decir, para designar la función de cada elemento dentro de la página: párrafo, titular, texto destacado, tabla, lista de elementos, etc. Una vez de nidos los contenidos, se utiliza el lenguaje CSS para definir el aspecto de cada elemento: color, tamaño y tipo de letra del texto, separación horizontal y vertical entre elementos, posición de cada elemento dentro de la página, etc.

2.4.3 JavaScript

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas, las cuales son aquellas que incorporan efectos atractivos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario [6].

Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden ejecutar directamente en cualquier navegador sin necesidad de procesos intermedios.

2.4.4 jQuery

jQuery es una biblioteca de JavaScript que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web [7].

jQuery es una de las bibliotecas más utilizadas de JavaScript, ya que ofrece una gran cantidad de funcionalidades basadas en JavaScript que de otra manera requerirán de mucho más código, es decir, con las funciones propias de esta biblioteca se pueden lograr grandes resultados en menos tiempo y código.

2.4.5 AJAX

AJAX, acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas [28]. Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios, mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, mejorando la interactividad, velocidad y usabilidad en las aplicaciones [8].

En las aplicaciones web tradicionales, las acciones del usuario en la página (hacer click en un botón, seleccionar un valor de una lista, etc.) generan llamadas al servidor. Una vez procesada la petición del usuario, el servidor devuelve una nueva página HTML al navegador del usuario. Esta técnica tradicional para crear aplicaciones web funciona correctamente, pero no crea una buena sensación al usuario. Al realizar peticiones continuas al servidor, el usuario debe esperar a que se recargue la página con los cambios solicitados. Si la aplicación debe realizar peticiones continuas, su uso podrá convertirse un poco molesto.

AJAX permite mejorar la interacción del usuario con la aplicación, evitando las recargas constantes de la página, ya que el intercambio de información con el servidor se produce en un segundo plano, y una vez que el intercambio de información se ha concretado, es posible actualizar la página HTML desde el propio navegador [8].

2.4.6 AngularJS

AngularJS es un framework de JavaScript de código abierto, mantenido por Google, que se utiliza para crear y mantener aplicaciones de una sola página [9] y tiene como objetivo aumentar las aplicaciones basadas en navegador con capacidad de Modelo Vista Controlador (MVC), en un esfuerzo para hacer que el desarrollo y las pruebas sean más fáciles [9].

Este framework adapta y amplía el HTML tradicional para servir mejor el contenido dinámico a través del data-binding bidireccional, el cual permite la sincronización automática de modelos y vistas. Como resultado, AngularJS mejora la testeabilidad y el rendimiento de las aplicaciones [9].

2.4.7 Bootstrap

Bootstrap es un framework que ofrece un conjunto de herramientas de software libre para diseño de sitios y aplicaciones web, contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como, extensiones de JavaScript adicionales [10].

Una característica de Bootstrap es la de adaptar la interfaz del sitio web al tamaño del dispositivo en que se visualice. Es decir, el sitio web se adapta automáticamente al tamaño de la pantalla de un PC, una Tablet u otro dispositivo. Esta técnica de diseño y desarrollo se conoce como responsive design o diseño adaptativo.

2.5 Tecnologías para el desarrollo del servidor

En el servidor es donde realmente residen los datos y la lógica de negocio, y para mantener estos datos y definir estas reglas se puede hacer uso de tecnologías como las que se presentan a continuación.

2.5.1 Bases de datos

Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados de forma sistemática para su posterior uso [11].

En otras palabras, se puede decir que una base de datos es un almacén "que permite guardar una gran cantidad de datos de forma organizada, con la intención de que luego se puedan utilizar o analizar.

2.5.2 Sistemas gestores de bases de datos

Los sistemas gestores de bases de datos (SGBD) consisten en una colección de datos interrelacionados, conocida como base de datos, y un conjunto de programas para acceder a dichos datos [11]. El objetivo principal de un SGDB es proporcionar una forma de almacenar y recuperar la información de una base de datos de manera práctica y eficiente.

2.5.3 PostgreSQL

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. PostgreSQL

utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. De esta forma un fallo en uno de los procesos no afectará al resto y el sistema continuará funcionando [12].

PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando.

Las características técnicas de este SBDB lo hacen uno de los más potentes y robustos del mercado. La estabilidad, potencia, robustez, facilidad de administración e implementación de estándares han sido los aspectos que más se han tenido en cuenta durante su desarrollo. PostgreSQL funciona muy bien con grandes cantidades de datos y una alta concurrencia de usuarios accediendo a la vez al sistema.

Entre sus características se encuentran [12]:

- Es una base de datos 100% ACID
- Integridad referencial
- Replicación asincrónica/sincrónica
- Copias de seguridad en caliente
- Acceso encriptado vía SSL
- Completa documentación y Licencia BSD
- Disponible para Linux y UNIX en todas sus variantes (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows 32/64bit.
- Funciones/procedimientos almacenados (stored procedures) en numerosos lenguajes de programación, entre ellos PL/pgSQL, PL/Perl, PL/Python y PL/Tcl.
- Soporta el almacenamiento de objetos binarios grandes (gráficos, videos, sonido, entre otros)

- Numerosos tipos de datos y posibilidad de definir nuevos tipos. Además de los tipos estándares en cualquier base de datos, tenemos disponibles, entre otros, tipos geométricos, de direcciones de red, de cadenas binarias, UUID, XML, matrices

2.5.4 Frameworks para el desarrollo de aplicaciones web

Un framework para aplicaciones web está diseñado para apoyar el desarrollo de sitios web dinámicos, aplicaciones web y servicios web. Este tipo de frameworks intenta aliviar el exceso de carga asociado con actividades comunes usadas en desarrollos web. Por ejemplo, muchos framework proporcionan bibliotecas para acceder a bases de datos, estructuras para plantillas y gestión de sesiones, y con frecuencia facilitan la reutilización de código [13].

Generalmente, los frameworks de desarrollo web suelen ser implementaciones de patrones de diseño ya conocidos, siendo el patrón de diseño Modelo Vista Controlador (MVC) uno de los más utilizados en el desarrollo de aplicaciones web.

2.5.5 Patrón de diseño Modelo – Vista – Controlador

MVC es un patrón de arquitectura de software frecuentemente utilizado para la construcción de aplicaciones web. Este patrón divide la aplicación en tres elementos principales, que realmente representan capas, y define la forma en que se comunican entre sí [14]. En la figura 4 se presentan los componentes del patrón MVC.

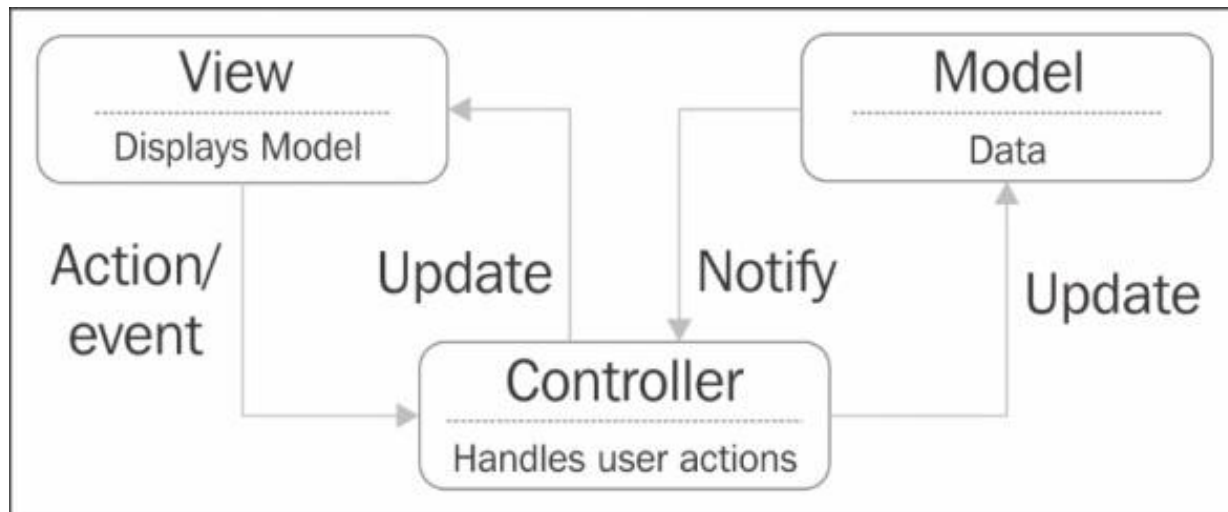


Figura 6. Modelo - Vista – Controlador [14].

Este patrón de arquitectura de software está basado en la reutilización de código y la separación de conceptos, con lo cual se intenta facilitar el desarrollo de aplicaciones y el mantenimiento de las mismas [15].

A continuación se describen brevemente los componentes propuestos por MVC [1]:

- **Modelo:** El modelo representa la parte de la aplicación que implementa la lógica de negocio. Esto significa que es responsable de la recuperación de datos convirtiéndolos en conceptos significativos para la aplicación, así como su procesamiento, validación, asociación y cualquier otra tarea relativa a la manipulación de dichos datos.

- **Controlador:** El controlador gestiona las peticiones de los usuarios. Es responsable de responder la información solicitada con la ayuda tanto del modelo como de la vista.
- **Vista:** La vista hace una presentación de los datos del modelo estando separada de los objetos del modelo. Es responsable del uso de la información de la cual dispone para producir cualquier interfaz de presentación de cualquier petición que se presente.

2.5.6 Express

Express es un framework web minimalista y flexible para Node.js. Este último (Node.js) es un entorno en tiempo de ejecución que proporciona un conjunto de módulos para el manejo de sistemas de archivos, operaciones de entrada/salida, tráfico de red, entre otros, para desarrollar aplicaciones en el lado del servidor haciendo uso de JavaScript.

Express proporciona un conjunto robusto de características para aplicaciones web y móviles. Este framework está basado en JavaScript, ofrece una gran variedad de métodos HTTP y características, que permiten el desarrollo de una potente Interfaz de Programación de Aplicaciones (API por sus singlas en inglés) de forma rápida y fácil [16].

La filosofía a de Express es proporcionar un pequeño pero robusto conjunto de herramientas para servidores HTTP, lo cual lo convierte en una gran solución para aplicaciones de una sola página (SPA por sus siglas en ingles), páginas web, aplicaciones híbridas o para el desarrollo de servicios web [17].

2.5.7 Servicios web

Un servicio web es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones, independientemente del lenguaje de programación en el cual hayan sido desarrolladas y de la plataforma en la cual se estén ejecutando [18].

En los servicios web se hace uso de diversos formatos (XML, JSON) para representar los datos o información a intercambiar, así como también se hace uso de un conjunto de protocolos sobre los cuales se establece la comunicación.

Entre las características de los servicios web se encuentran [19]:

- Utilización de estándares de internet: La única forma para que los servicios web sean utilizados por la cantidad de sistemas heterogéneos existentes en Internet, es el empleo del protocolo de transferencia de datos HTTP utilizado por todos los navegadores Web.
- Basados en tecnologías de paso de mensajes: La interacción entre el cliente y el proveedor del servicio es empaquetada en unidades auto - descriptivas denominadas mensajes. Dicha interacción se describe en función de los mensajes intercambiados.
- Combinan lo mejor de la tecnología de componentes y de la tecnología web: Los servicios web presentan una funcionalidad de caja negra que puede ser reutilizada sin preocuparse de cómo es implementada y esto permite interfaces bien definidas.

Entre las ventajas de utilizar servicios web se encuentran [18]:

- Al apoyarse en HTTP, pueden esquivar medidas de seguridad basadas en firewall, cuyas reglas tratan de bloquear o auditar la comunicación entre programas a ambos lados de la barrera.
- Para realizar transacciones no pueden compararse en su grado de desarrollo con los estándares abiertos de computación distribuida como CORBA (Common Object Request Broker Architecture).

2.5.8 JSON

JSON (JavaScript Object Notation - Notación de Objetos de JavaScript) es un formato ligero de intercambio de datos, basado en un subconjunto del Lenguaje de Programación JavaScript, Standard ECMA-262 3rd Edition - Diciembre 1999 [20]. JSON es un formato de texto que es completamente independiente del lenguaje pero utiliza convenciones que son ampliamente conocidos por los programadores en lenguajes como C, C++, Java, JavaScript, Perl, Python, y muchos otros. Estas propiedades hacen que JSON sea un lenguaje ideal para el intercambio de datos.

Es ampliamente usado para intercambio de información entre servicios web y APIs REST. Su simplicidad y facilidad de implementación le otorgan un gran desempeño y lo convierten en una alternativa a XML [21].

2.5.9 REST

El término REST (Representational State Transfer) se originó en el año 2000 y se refería originalmente a un conjunto de principios de arquitectura. En la actualidad el

término es ampliamente utilizado en la comunidad del desarrollo y se usa en el sentido más general para describir cualquier interfaz entre sistemas que utilice directamente HTTP para obtener datos o indicar la ejecución de operaciones sobre los datos, en cualquier formato (XML, JSON, etc) [22].

Un concepto importante en REST es la existencia de recursos (elementos de información), que pueden ser accedidos utilizando un identificador global (URL), y para manipular estos recursos, los componentes de la red (clientes y servidores) se comunican a través de una interfaz estándar (HTTP) e intercambian representaciones de estos recursos, es decir, la información que se descarga y se envía.

Entre las características de REST se encuentran [22]:

- Conjunto de operaciones bien definidas que se aplican a todos los recursos de información: HTTP en sí define un conjunto pequeño de operaciones, las más importantes son POST, GET, PUT y DELETE.
- Sintaxis universal para identificar los recursos: En un sistema REST, cada recurso es referenciado únicamente a través de su URL.
- Independiente de las tecnologías utilizadas para el desarrollo de los servicios web.

2.6 Aplicaciones de una sola página

En una aplicación web tradicional, cada vez que la aplicación hace una llamada al servidor, el servidor responde con una nueva página HTML, lo cual hace que se refresque la página en el navegador. Por otro lado, están las aplicaciones de una sola

página o single-page application (SPA), las cuales son aplicaciones web que cargan solo una página HTML y la misma se va actualizando dinámicamente a medida que el usuario interactúa con la aplicación [23].

En la figura 7 se presenta la diferencia entre el ciclo de vida de una aplicación web tradicional y el de una SPA.

Las SPAs hacen uso de Ajax y HTML5 para crear aplicaciones web fluidas, sin refrescar la página constantemente. Sin embargo, se debe tener en cuenta que gran parte de la lógica de la aplicación, residirá en la parte del cliente, siendo desarrollada generalmente con JavaScript [23].

Entre los beneficios de las SPAs se encuentra que son más fluidas, ya que se elimina el efecto de recarga de la página con cada llamada al servidor. Otro beneficio es que permite separar la capa de presentación (HTML) de la capa de lógica de negocio (peticiones Ajax y objetos JSON como respuesta). La separación entre la capa de presentación (aplicación web) y la capa de lógica de negocio (servidor) permite realizar modificaciones en cada capa, sin que la otra sea vea afectada.

Bajo el enfoque de las SPAs, después de cargar la página HTML inicial, el servidor actúa puramente como una capa de servicios, el cliente solo necesita saber la petición HTTP que debe realizar para obtener la información que necesita. Además, este enfoque permite hacer un cambio total en la parte del servidor y el cliente seguirá funcionando correctamente, siempre y cuando se mantenga la interfaz de servicios. De igual forma sucede para el cliente, puede ser cambiado totalmente sin necesidad de modificar los servicios [23].

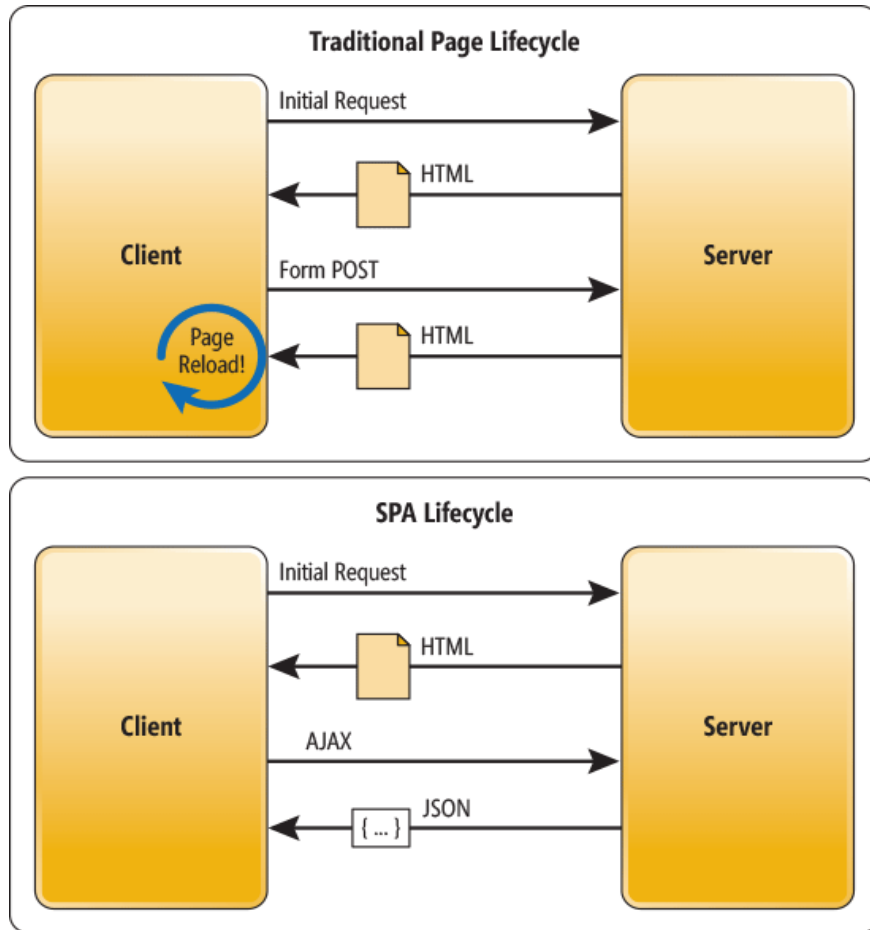


Figura 7. Ciclo de vida de una aplicación web tradicional y una SPA [23].

Marco Metodológico

El desarrollo de este proyecto fue guiado bajo la metodología SCRUM, la cual es presentada en este capítulo, ya que es una metodología de adaptación, rápida, flexible y eficaz, diseñada para ofrecer valor significativo de forma rápida en todo proyecto. Permitió a lo largo del desarrollo presentar al cliente soluciones operables, permitiéndole observar resultados y decidir cambios en marcha o realizar alguna adaptación que se ajustara más a las necesidades de la compañía de seguros.

3.1 Metodología de Desarrollo

En la Ingeniería de Software una Metodología de Desarrollo es un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo en sistemas de información [24]. Este marco de trabajo incluye notaciones, reglas, sugerencias de diseño y guías de procesos [25].

3.2 Metodologías Tradicionales

Teniendo en cuenta la filosofía de desarrollo de las metodologías, aquellas con mayor énfasis en la planificación y control de proyecto, en especificación precisa de requisitos y modelado, reciben el apelativo de Metodologías Tradicionales o Pesadas [26].

Estas metodologías tradicionales imponen una disciplina de trabajo sobre el proceso de desarrollo de software, con el fin de conseguir un software más eficiente.

Para ello se hace énfasis en la planificación total de todo el trabajo a realizar y una vez que esta todo detallado, comienza el ciclo de desarrollo del producto de software y se centran especialmente en el control del proceso, mediante una rigurosa definición de roles, actividades, artefactos, herramientas y notaciones para el modelado y documentación detallada, sin embargo, las metodologías tradicionales no se adaptan adecuadamente a los cambios, por lo que no son métodos adecuados cuando se trabaja en un entorno donde los requisitos no pueden predecirse o bien pueden variar [26].

3.3 Metodologías Ágiles

Los negocios operan en un entorno global que suele cambiar con gran rapidez. Deben responder a nuevas oportunidades y mercados, condiciones económicas cambiantes y la aparición de nuevos productos y servicios. El software está presente en casi todas las operaciones de negocio y es fundamental que el software nuevo se desarrolle rápidamente para aprovechar las nuevas oportunidades y responder a la presión competitiva, por lo que el desarrollo y entrega de forma rápida son frecuentemente los requerimientos más críticos de los sistemas de software [27]. Además, debido al entorno cambiante donde operan estos negocios, es difícil definir un conjunto completo de requerimientos estables acerca del software a desarrollar.

Los procesos de desarrollo de software basados en una completa especificación de los requerimientos y posterior diseño, construcción y pruebas del sistema, no se ajustan al desarrollo rápido de aplicaciones [27]. Cuando los requerimientos del sistema cambian, el diseño o implementación del sistema se tiene que realizar o probar nuevamente. Por esta razón, suele extenderse el desarrollo del software y el mismo

se entrega al cliente mucho tiempo después de que fuera inicialmente especificado.

Los procesos de desarrollo ágiles de software se proponen con la intención de aliviar los inconvenientes generados por los cambios sobre los requerimientos del software cuando se trabaja con un proceso de desarrollo basado en especificación. Los procesos ágiles permiten que los desarrolladores de software se centren en el software mismo en vez de su diseño y documentación [27].

Los métodos ágiles universalmente dependen de un enfoque iterativo para la especificación, desarrollo y entrega de software, y principalmente fueron diseñados para apoyar al desarrollo de aplicaciones de negocio donde los requerimientos del sistema normalmente cambian rápidamente durante el proceso de desarrollo. Están pensados para entregar software funcional de forma rápida a los clientes, quienes pueden entonces proponer que se incluyan en iteraciones posteriores del sistema nuevos requerimientos o cambios en los mismos.

Todos los métodos tienen límites, y los métodos ágiles son solo apropiados para algunos tipos de desarrollo de sistemas. Son más idóneos para el desarrollo de sistemas de negocios pequeños y de tamaño medio y para el desarrollo de productos para computadoras personales. No son adecuados para el desarrollo de sistemas a gran escala con equipos de desarrollo ubicados en diferentes lugares y donde pueda haber complejas interacciones con otros sistemas hardware y software. No se deben utilizar los métodos ágiles para el desarrollo de sistemas críticos en los que es necesario un análisis detallado de todos los requerimientos del sistema para comprender sus implicaciones de seguridad y protección.

3.4 SCRUM

SCRUM es una de las metodologías ágiles más populares. Es una metodología de adaptación, iterativa, rápida, flexible y eficaz, diseñada para ofrecer un valor significativo de forma rápida en todo el proyecto, garantizando transparencia en la comunicación y crea un ambiente de responsabilidad colectiva y de progreso continuo [28]. El marco de SCRUM, está estructurado de tal manera que es compatible con los productos y el desarrollo de servicios en todo tipo de industrias y en cualquier tipo de proyecto, independientemente de su complejidad [28].

SCRUM propone el software funcional sobre la excesiva documentación. Se van presentando al cliente las soluciones operables y no solo reportes de progresos, de esta forma el cliente puede decidir avanzar o parar, a diferencia de otros enfoques en los que solo se ven resultados al final [29].

De igual forma, SCRUM promueve la colaboración con el cliente en lugar de una rígida negociación de contratos. Por lo cual, es importante tener capacidad de respuesta para los cambios en lugar de seguir estrictamente una planificación, partiendo del principio de que el proyecto de software es cambiante. El propósito es que el cliente vaya observando los resultados, pueda decidir cambios en la marcha o incluso darle un giro completo al proyecto.

Entre las características de SCRUM se encuentran [29]:

- Enfoque iterativo e incremental.
- El marco SCRUM es compatible con el desarrollo de cualquier tipo proyecto.
- Gran parte del éxito del proyecto radica en la forma como el equipo se organice para trabajar.

- Promueve la colaboración con el cliente en lugar de una rígida negociación de contratos. Propone el software funcional sobre la excesiva documentación.

Algunas de las ventajas principales de la utilización de SCRUM en cualquier proyecto son [28]:

- Adaptabilidad: El control del proceso empírico y el desarrollo iterativo hacen que los proyectos sean adaptables y abiertos a la incorporación del cambio.
- Retroalimentación continua: La retroalimentación continua se proporciona a través de las reuniones diarias y demostraciones y validación de requerimientos.
- Entrega continúa de valor: Los procesos iterativos permiten la entrega continua de valor tan frecuentemente como el cliente lo requiere.
- Ritmo sostenible: Los procesos SCRUM están diseñados de tal manera que las personas involucradas pueden trabajar a un ritmo sostenible que, en teoría, se puede continuar indefinidamente.
- Entrega anticipada de alto valor: Asegura que los requisitos de mayor valor del cliente sean los primeros en cumplirse.
- Proceso de desarrollo eficiente: La asignación de un bloque de tiempo fijo y la reducción al mínimo del trabajo que no es esencial conducen a mayores niveles de eficiencia.
- Resolución de problemas de forma más rápida: La colaboración y colocación de equipos interfuncionales conducen a la resolución de problemas con mayor rapidez.

- Centrado en el cliente: El poner énfasis en el valor del negocio y tener un enfoque de colaboración con los socios asegura un marco orientado al cliente.
- Alta velocidad: Un marco de colaboración que le permite a los equipos interfuncionales altamente cualificados alcanzar su potencial y alta velocidad.

3.4.1 Flujo de un proyecto SCRUM

Una fortaleza clave de SCRUM radica en el uso de equipos inter-funcionales, auto-organizados, y empoderados que dividen su trabajo en ciclos de trabajo cortos y concentrados llamados Sprints [28]. La figura 8 proporciona una visión general del flujo de un proyecto SCRUM.

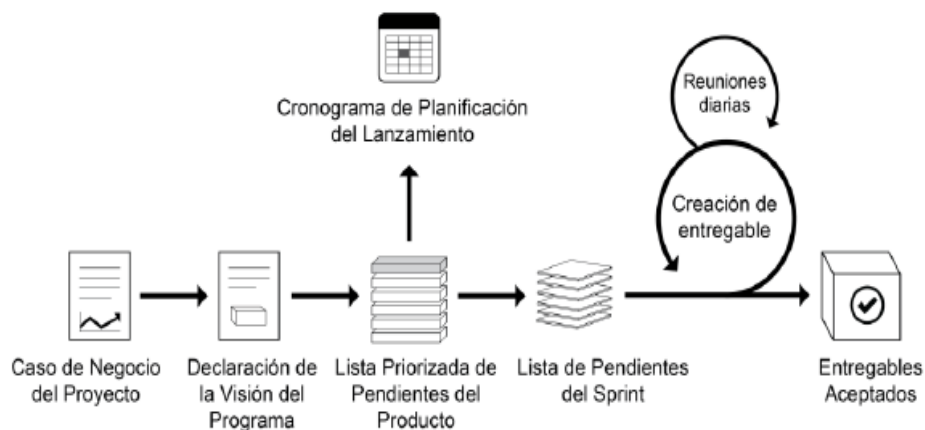


Figura 8. Flujo de SCRUM para un Sprint [28].

El ciclo de SCRUM comienza con una reunión de los socios, durante la cual se crea la visión del proyecto. Después, el propietario del producto desarrolla una Lista priorizada de pendientes del producto que contiene una lista requerimientos del negocio por orden de importancia en forma de una Historia de usuario. Cada sprint comienza con una Reunión de planificación del sprint durante la cual se consideran las historias de usuario de alta prioridad para su inclusión en el sprint.

Un sprint suele durar entre una y seis semanas durante las cuales el equipo SCRUM trabaja en la creación de Entregables en incrementos del producto potencialmente listos. Durante el sprint, se llevan a cabo Reuniones diarias muy breves y concretas en las que los miembros del equipo discuten progresos diarios. A medida que concluye el sprint, se lleva a cabo una Reunión de planificación del sprint en la cual se proporciona una demostración de los entregables al propietario del producto y a los socios relevantes. El propietario del producto acepta los entregables sólo si cumplen con los criterios de aceptación predefinidos. El ciclo del sprint termina con una Reunión de retrospectiva del sprint, donde el equipo presenta maneras para mejorar los procesos y el rendimiento a medida que avanzan al siguiente sprint.

3.4.2 Bloque de tiempo asignado

SCRUM trata al tiempo como uno de los limitantes más importantes en la gestión de un proyecto. Para hacer frente a la restricción del tiempo, SCRUM introduce un concepto de asignación de un bloque de tiempo, que propone la fijación de una cierta cantidad de tiempo para cada proceso y actividad en un proyecto SCRUM. Esto garantiza que los miembros del Equipo SCRUM no ocupen demasiado o muy poco

tiempo para un trabajo determinado, y que no desperdicien su tiempo y energía en un trabajo para el cual tienen poca claridad [28].

Algunas de las ventajas de la asignación del bloque de tiempo son las siguientes:

- Proceso de desarrollo eficiente
- Menos gastos generales
- Alta velocidad para los equipos

La asignación del bloque de tiempo es una práctica muy importante en SCRUM, y esta debe aplicarse con cuidado. Una asignación de un bloque de tiempo arbitraria puede llevar a la desmotivación del equipo y tener como consecuencia la creación de un ambiente tenso, por lo que esta se debe utilizar de manera apropiada.

3.4.3 Bloques de tiempo de SCRUM

La metodología de desarrollo SCRUM define los siguientes bloques de tiempo de forma que el equipo SCRUM pueda organizar sus actividades y el desarrollo de los objetivos a cumplir en el proyecto [28]:

- **Sprint:** Un sprint es una iteración que se asigna a un bloque de tiempo de una a seis semanas de duración durante el cual el SCRUM Master guía, facilita y protege al equipo SCRUM de impedimentos tanto internos como externos durante el proceso de creación de entregables. Esto ayuda a evitar una expansión de la visión más allá de su objetivo original, lo que podría afectar la meta del sprint. Durante este tiempo, el equipo trabaja para convertir las

necesidades de la lista priorizada de pendientes del producto en funcionalidades de productos fáciles de enviar. Para obtener los máximos beneficios de un proyecto SCRUM, siempre se recomienda mantener el sprint dentro de un bloque de tiempo de 4 semanas, a menos que existan proyectos con requisitos muy estables, en los que los sprints pueden extenderse hasta seis semanas.

- **Reunión diaria:** Esto se refiere a una reunión diaria de corta duración, que se asigna a un bloque de tiempo de 15 minutos. En esta reunión rápida, los miembros del equipo se reúnen para informar sobre cómo marcha el proyecto, respondiendo a las siguientes tres preguntas: **¿Qué fue concluido ayer?, ¿Qué se concluirá hoy?, ¿Qué impedimentos y obstáculos (si existen) se están enfrentando en la actualidad del proyecto?**. Esta reunión se lleva a cabo por el equipo como parte del proceso de la realización de la reunión diaria.
- **Reunión de planificación del sprint:** Esta reunión se lleva a cabo antes del sprint, como parte del proceso de creación de la lista de pendientes del sprint. Se asigna a un bloque de tiempo de ocho horas durante un sprint de un mes de duración. La reunión de planificación del sprint se divide en dos partes:
 1. **Definición del objetivo:** Durante la primera mitad de la reunión, el propietario del producto explica la máxima prioridad de las historias de usuario o requisitos en la lista priorizada de pendientes del producto para el equipo SCRUM. Después, el equipo define el objetivo del sprint en colaboración con el propietario del producto.
 2. **Estimación de tareas:** Durante la segunda mitad de la reunión, el equipo SCRUM decide como completar la lista priorizada de pendientes del producto seleccionados para cumplir con la meta del Sprint.

A veces, a las reuniones de planificación de tareas (efectuadas durante el proceso de creación de tareas) y a las reuniones de estimación de tareas, (efectuadas durante el proceso de estimación de tareas), también se les conoce como reuniones de planificación del sprint.

- **Reunión de revisión del sprint:** La reunión de revisión del sprint se asigna a un bloque de tiempo de cuatro horas en un sprint de un mes. Durante la reunión de revisión del sprint que se efectúa en el proceso de demostración y validación del sprint, el equipo SCRUM le presenta los entregables del sprint actual al propietario del producto. Este revisa el producto (o incremento del producto) para compararlo con los criterios de aceptación acordados y luego acepta o rechaza las historias de usuario completadas.
- **Reunión de retrospectiva del sprint:** Esta reunión se asigna a un bloque de tiempo de 4 horas para un sprint de un mes, y se lleva a cabo como parte del proceso retrospectiva del sprint. Durante esta reunión, el equipo SCRUM revisa y reflexiona sobre el sprint anterior en relación a los procesos que se siguieron, las herramientas empleadas, la colaboración, y los mecanismos de comunicación, al igual que otros aspectos de interés para el proyecto. El equipo discute lo que salió bien durante el sprint anterior y lo que no salió bien, con el objetivo de aprender y mejorar sprints futuros. Algunas oportunidades de mejora o las mejores prácticas de esta reunión también podrían actualizarse como parte de los documentos del cuerpo de asesoramiento de SCRUM.

La figura 9 ilustra las duraciones de los bloques de tiempo para las reuniones relacionadas con SCRUM.

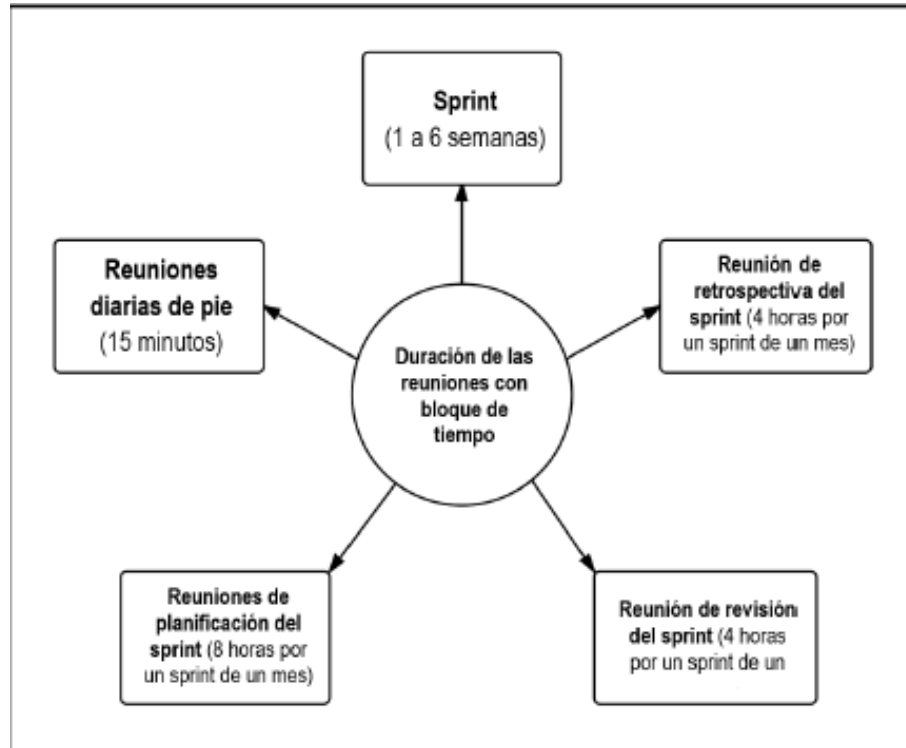


Figura 9. Duración de los bloques de tiempo para las reuniones de SCRUM [28].

3.4.4 Roles de un proyecto SCRUM

El entendimiento de los roles y las responsabilidades definidas es muy importante para asegurar la implementación exitosa de los proyectos SCRUM.

Los roles de SCRUM se dividen en dos categorías [28]:

- **Roles centrales:** Los roles centrales son aquellos que obligatoriamente se requieren para crear el producto del proyecto, están comprometidos con el

proyecto, y por último son los responsables del éxito de cada sprint del proyecto y del proyecto en su totalidad.

- **Roles no centrales:** Los roles centrales son aquellos que no son obligatoriamente necesarios para el proyecto SCRUM, y pueden incluir miembros de los equipos que tengan interés en el proyecto, pero que no tienen ninguna función formal en el equipo del proyecto. Ellos pueden interactuar con el equipo, pero no son responsables del éxito del proyecto. Los roles no centrales también deben tenerse en cuenta en cualquier proyecto de SCRUM.

3.4.5 Roles centrales

Hay tres roles centrales en SCRUM que en última instancia llevan la responsabilidad de cumplir con los objetivos del proyecto. Los roles centrales son el propietario del producto, el SCRUM Master y el equipo SCRUM. En conjunto se les conoce como el equipo principal de SCRUM y es importante tener en cuenta que, de estos tres roles, ningún rol tiene autoridad sobre los otros [28].

- **Propietario del producto:** El propietario del producto es la persona responsable de maximizar el valor del negocio para el proyecto. Este rol es responsable de articular los requisitos del cliente y de mantener la justificación del negocio del proyecto. El propietario del producto representa la voz del cliente.
- **SCRUM Master:** El SCRUM Master es un facilitador que asegura que el equipo SCRUM esté dotado de un ambiente propicio para completar con éxito el desarrollo del producto. El SCRUM Master guía, facilita e imparte prácticas de

SCRUM a todos los participantes en el proyecto, elimina los impedimentos que enfrenta el equipo, y asegura que se estén siguiendo los procesos de SCRUM.

- **Equipo SCRUM:** El equipo SCRUM es un grupo o equipo de personas que son responsables de la comprensión de los requerimientos del negocio que se especifican por el propietario del producto, de la estimación de las historias de usuarios y de la creación final de los entregables del proyecto.

3.4.6 Roles no centrales

Los roles no centrales son aquellos roles que no son obligatoriamente necesarios para el proyecto SCRUM y pueden no participar en el proceso de SCRUM [21]. Sin embargo, es importante tener conocimiento sobre estos roles no centrales, ya que podrían desempeñar un rol importante en algunos proyectos de SCRUM.

Los roles no centrales pueden incluir los siguientes [28]:

- **Socios:** El(los) socio(s) es un término colectivo que incluye a los clientes, usuarios y patrocinadores, que a menudo interactúan con el propietario del producto, el SCRUM Master y el equipo SCRUM para proporcionarles las entradas y facilitar la creación del producto del proyecto, servicio, o cualquier otro resultado. El(los) socio(s) influyen en el proyecto a lo largo del desarrollo del mismo. Los socios también pueden desempeñar un rol en los procesos importantes de SCRUM tales como el desarrollo de épica(s), la creación de la lista priorizada de pendientes del producto, la realización del plan de lanzamiento y la retrospectiva del sprint.

- **Vendedores:** Los vendedores incluyen a individuos u organizaciones externas que ofrecen productos y servicios que no están dentro de las competencias básicas de la organización del proyecto.
- **Cuerpo de asesoramiento de SCRUM:** El cuerpo de asesoramiento de SCRUM es una función opcional. Por lo general, se compone de un grupo de documentos y/o un grupo de expertos que normalmente están involucrados en la definición de los objetivos relacionados con la calidad, las regulaciones gubernamentales, la seguridad y otros parámetros clave de la organización. Estos objetivos guían la labor llevada a cabo por el propietario del producto, el SCRUM Master y el equipo SCRUM. El cuerpo de asesoramiento de SCRUM también ayuda a captar las mejores prácticas que deben utilizarse en todos los proyectos de SCRUM en la organización.

3.4.7 Terminología asociada a la Metodología SCRUM

A continuación se presentan una serie de términos claves al momento de estudiar la Metodología SCRUM [28]:

- **Sprint:** Un sprint es una iteración con un bloque de tiempo asignado de una a seis semanas de duración durante el cual el equipo SCRUM crea y trabaja en los entregables del sprint.
- **Lista priorizada de pendientes del producto:** La lista priorizada de pendientes del producto es un solo documento de requisitos que define el ámbito del

proyecto, proporcionando una lista de priorizada de las características del producto o servicio que habrá de entregar el proyecto.

- **Creación de la lista de pendientes del sprint:** En este proceso, el equipo principal de SCRUM lleva a cabo reuniones de planificación del sprint donde el grupo crea una lista de pendientes del sprint y se incluyen todas las tareas a realizar durante el mismo.
- **Lista de tareas:** Lista completa que contiene todas las tareas a las que se ha abocado el equipo SCRUM en el actual sprint. Contiene descripciones de cada tarea.
- **Entregables del sprint:** Los entregables del sprint son los incrementos del producto o los entregables que se terminan al final de cada sprint.
- **Creación de entregables:** La creación de entregables es el proceso en el cual el equipo SCRUM trabaja en las tareas de la lista de pendientes del sprint para crear entregables del sprint.

Marco Aplicativo

En el presente capítulo se presenta una descripción general acerca de la solución planteada para agilizar el proceso de Visitas Clínicas de la compañía de seguros, la arquitectura y el conjunto de tecnologías utilizado para implementarla, y la forma en que se aplicó la metodología de desarrollo de software SCRUM para guiar el desarrollo de esta solución. Igualmente, se detallan acerca de la solución los requerimientos a cumplir, los perfiles de usuarios que interactúan y las funcionalidades que abarcan los módulos desarrollados en la misma.

4.1 Descripción general de la solución

Con la intención de agilizar el proceso de Visitas Clínicas de la compañía aseguradora, se planteó desarrollar para el presente Trabajo Especial de Grado, una aplicación web que permitiese al personal de la compañía involucrado en este proceso, mantener una mejor organización sobre las tareas que están envueltas en este proceso, así como también los documentos que se necesitan durante el desarrollo de estas tareas. A través de la aplicación web se podrán realizar solicitudes de Visita Clínica, tener control sobre de los visitantes clínicos que se encuentran disponibles y las visitas que tienen asignadas. Además, la misma permitirá a los visitantes clínicos obtener la información y la documentación necesaria para realizar la Visita Clínica y posteriormente cargar la información obtenida durante la misma.

Para esta aplicación web se implementó la arquitectura descrita en la figura 10, la cual se corresponde con una arquitectura del tipo Cliente - Servidor. En la

arquitectura que se planteó se encuentran tres capas lógicas que son: Lógica de Presentación, Lógica de Negocio y Lógica de Datos.

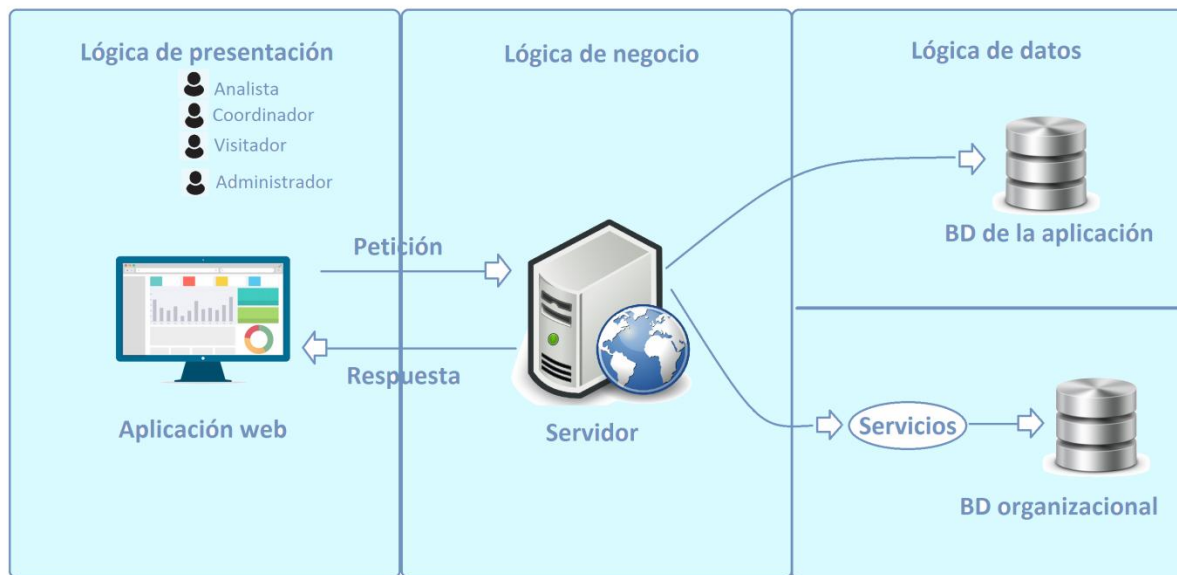


Figura 10. Arquitectura implementada en el sistema web. Fuente: Elaboración propia

La Lógica de Presentación se corresponde con la aplicación web, la cual será utilizada por el analista de seguimiento de carta aval, el coordinador de visitas y los visitantes clínicos. El analista podrá hacer la solicitud de la visita clínica mediante la aplicación. El coordinador de visitas recibirá las solicitudes de visita a través de la misma, podrá consultar los visitantes clínicos que se encuentran disponibles y podrá hacer la asignación a alguno de estos visitantes. Finalmente, el visitante clínico podrá visualizar las visitas clínicas que tiene asignadas y la información necesaria para

realizar dicha visita. Después de realizar la visita, el visitador podrá acceder a la aplicación para cargar la información obtenida.

La Lógica de Negocio se corresponde con la implementación de todas las reglas de negocio que se ven envueltas en este proceso de la compañía aseguradora. El desarrollo de la Lógica de Negocio se llevó a cabo a través del framework Express, el cual es un framework web minimalista y flexible para Node.js.

La parte de la Lógica de Datos representa los datos necesarios para el funcionamiento de la aplicación. El almacenamiento de estos datos se encuentra dividido en dos partes: la base de datos propia de la aplicación y la base de datos organizacional. En la base de datos de la aplicación se almacenan los datos generados a través del uso de esta aplicación entre los que se encuentran: las solicitudes de visita generadas por los analistas, qué asignaciones tiene cada visitador clínico y la información cargada acerca de las visitas clínicas realizadas. Esta base de datos se implementó haciendo uso de PostgreSQL, ya que el sistema desarrollado como solución es un sistema que es de uso diario y se requiere de una base de datos que ofrezca una gran estabilidad y que pueda ser utilizada bajo diversos sistemas operativos, con la intención de disminuir el acoplamiento de la base de datos con el sistema operativo. En la base de datos organizacional están los datos de los analistas, los coordinadores y visitadores clínicos, los datos de los asegurados, los presupuestos dados en solicitud de la carta aval y los datos de la misma. Estos datos serán accedidos por servicios web desarrollados por parte de la compañía de seguros.

Teniendo en cuenta el posible desarrollo a futuro de aplicaciones que puedan agilizar más la gestión de este proceso de visitas clínicas y que necesiten acceder a los datos generados a partir del uso de esta aplicación web, se decidió implementar el acceso a los datos que esta aplicación genera, a partir de servicios web para así desacoplar el acceso a estos datos y las aplicaciones que necesiten acceder a los mismos.

El desarrollo de los servicios web se hizo bajo el enfoque REST, ya que es una manera fácil y sencilla de acceder a los mismos, está basado en HTTP, y es fácil de desarrollar y mantener. Como formato de intercambio de información a través de los servicios web se utiliza JSON, por ser un formato simple y con gran desempeño para el desarrollo de servicios web.

Al estar orientado el acceso y manipulación sobre los datos en el servidor a través de servicios web, se propuso desarrollar una aplicación de una sola página (SPA) que consuma estos servicios directamente. Para el desarrollo de esta aplicación se utilizaron tecnologías como HTML5, CSS3 y JavaScript, haciendo uso del framework Bootstrap para facilitar el diseño de las interfaces de la aplicación y el framework Angular.JS, el cual es está enfocado para el desarrollo de aplicaciones SPA.

En la figura 11 se presenta la arquitectura implementada con las tecnologías mencionadas anteriormente, contemplando únicamente las partes correspondientes al desarrollo de este trabajo, ya que los servicios para acceder a la base de datos organizacional son desarrollados por la propia compañía de seguros.

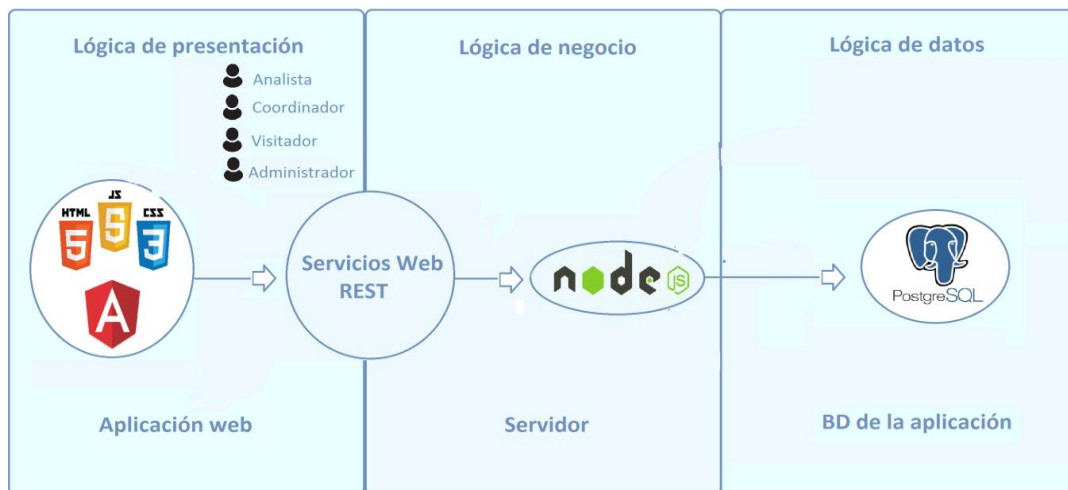


Figura 11. Tecnologías utilizadas para implementar la arquitectura. Fuente: Elaboración propia

4.2 Requerimientos del sistema

Los requerimientos acordados con el cliente (la compañía aseguradora) se clasifican de la siguiente forma:

Requerimientos funcionales:

- Gestionar las cuentas de usuario
- Crear y registrar solicitudes de visitas clínicas
- Consultar solicitudes de visitas clínicas
- Hacer seguimiento a las solicitudes de visita de acuerdo a un estatus
- Consultar cartas avales y sus correspondientes presupuestos
- Permitir la carga de documentos digitales que avalen la veracidad de la información cargada acerca de la visita clínica
- Enviar notificaciones a través de correo electrónico a lo largo de desarrollo del proceso
- Gestionar el registro de clínicas afiliadas a la compañía aseguradora
- Gestionar el registro de la información del personal involucrado en el proceso de visitas clínicas
- Gestionar el registro de la información asociada a las pólizas de los clientes
- Gestionar el registro de la información asociada a las cartas avales y sus correspondientes presupuestos

Requerimientos No funcionales:

- Usar tecnologías multiplataforma para el desarrollo del sistema
- Desarrollar el sistema a partir de servicios web que puedan alimentar futuros sistemas de la compañía aseguradora
- Permitir la carga de datos en el sistema a través de algún formato de fácil gestión

- Interfaces intuitivas y de fácil uso

4.3 Perfiles de usuario

- **Analista de Seguimiento de Carta Aval:** es el usuario encargado de hacer solicitud de una visita clínica para una determinada carta aval. Entre sus actividades se encuentran:
 - Buscar y visualizar cartas avales con sus presupuestos asociados
 - Solicitar una visita clínica sobre una determinada carta aval
 - Cancelar la solicitud de una determinada visita clínica
 - Hacer seguimiento de sus solicitudes de visita de acuerdo al estatus de las mismas
 - Visualizar cambios cargados sobre el presupuesto de sus solicitudes de visita clínica
 - Visualizar los datos de las encuestas cargadas sobre sus solicitudes de visita clínica
 - Visualizar documentos digitales que avalen la veracidad de los cambios hechos sobre el presupuesto y los datos cargados de la encuesta
 - Configurar parámetros básicos de su cuenta de usuario
- **Coordinador de Visitas Clínicas:** es el usuario encargado de distribuir las solicitudes de visitas clínicas entre el grupo de visitadores clínicos encargado de coordinar. Entre sus actividades se encuentran:
 - Buscar y visualizar una solicitud de visita clínica, junto con su carta aval y presupuesto asociado
 - Asignar una determinada visita clínica

- Cancelar la asignación de una determinada visita clínica
 - Hacer seguimiento de las solicitudes de visita que ha coordinado de acuerdo al estatus de las mismas
 - Visualizar cambios cargados sobre el presupuesto las solicitudes de visita clínica que ha coordinado
 - Visualizar encuestas cargadas sobre las solicitudes de visita clínica que ha coordinado
 - Visualizar documentos digitales que avalen la veracidad de los cambios hechos sobre el presupuesto y los datos cargados de la encuesta
 - Ordenar revisión sobre los cambios hechos sobre el presupuesto y los datos cargados de la encuesta
 - Autorizar la finalización de una determinada visita clínica
 - Configurar parámetros básicos de su cuenta de usuario
-
- **Visitador Clínico:** es el usuario encargado de llevar a cabo la visita clínica. Una vez realizada esta, procede a cargar la información obtenida durante la realización de la misma. Entre sus actividades se encuentran:
 - Buscar y visualizar una solicitud de visita clínica asignada, junto con su carta aval y presupuesto asociado
 - Hacer seguimiento de las solicitudes de visita que tiene asignadas de acuerdo al estatus de las mismas
 - Cargar cambios sobre el presupuesto
 - Cargar datos sobre la encuesta
 - Cargar documentos digitales que avalen la veracidad de los cambios hechos sobre el presupuesto y los datos cargados de la encuesta

4.4 Descripción del flujo asociado a la solución

A continuación se detalla la solución desarrollada para el proceso de visitas clínicas, la cual a su vez se encuentra representada gráficamente en la figura 12:

En primer lugar se encuentra el analista, el cual luego de la activación de la carta aval solicita una visita clínica para dicha carta aval. Luego, se registra en el sistema la solicitud de esta visita clínica y se envía al coordinador de visitas una notificación indicando que tiene una solicitud de visita pendiente por asignar. Una vez recibida la notificación, el coordinador puede consultar la lista de visitantes clínicos que coordina y hace la asignación de la visita a alguno de estos. Después de haber realizado la asignación, el sistema envía una notificación al visitador clínico en cuestión indicando que tiene una visita por realizar. El visitador clínico luego de recibir la notificación, puede visualizar e imprimir la documentación necesaria para llevar a cabo el proceso de visita clínica y también cargar la información obtenida en la visita clínica. Posteriormente, esta información cargada es visualizada por el coordinador de visitas, quien se encarga de verificar la información cargada y autoriza el envío de esta información (en caso de que no haya problema) al analista de seguimiento de carta aval para que la pueda visualizar y continuar con el protocolo seguido por la compañía de seguros. En caso de que el coordinador detecte o visualice alguna incoherencia con la información cargada por el visitador clínico, puede ordenar una revisión sobre esta información, lo cual envía al visitador clínico una notificación indicando que debe hacer una revisión sobre la información cargada en el proceso de visita clínica. Una vez hecha la corrección, se notifica al coordinador y se continúa con el flujo del proceso.

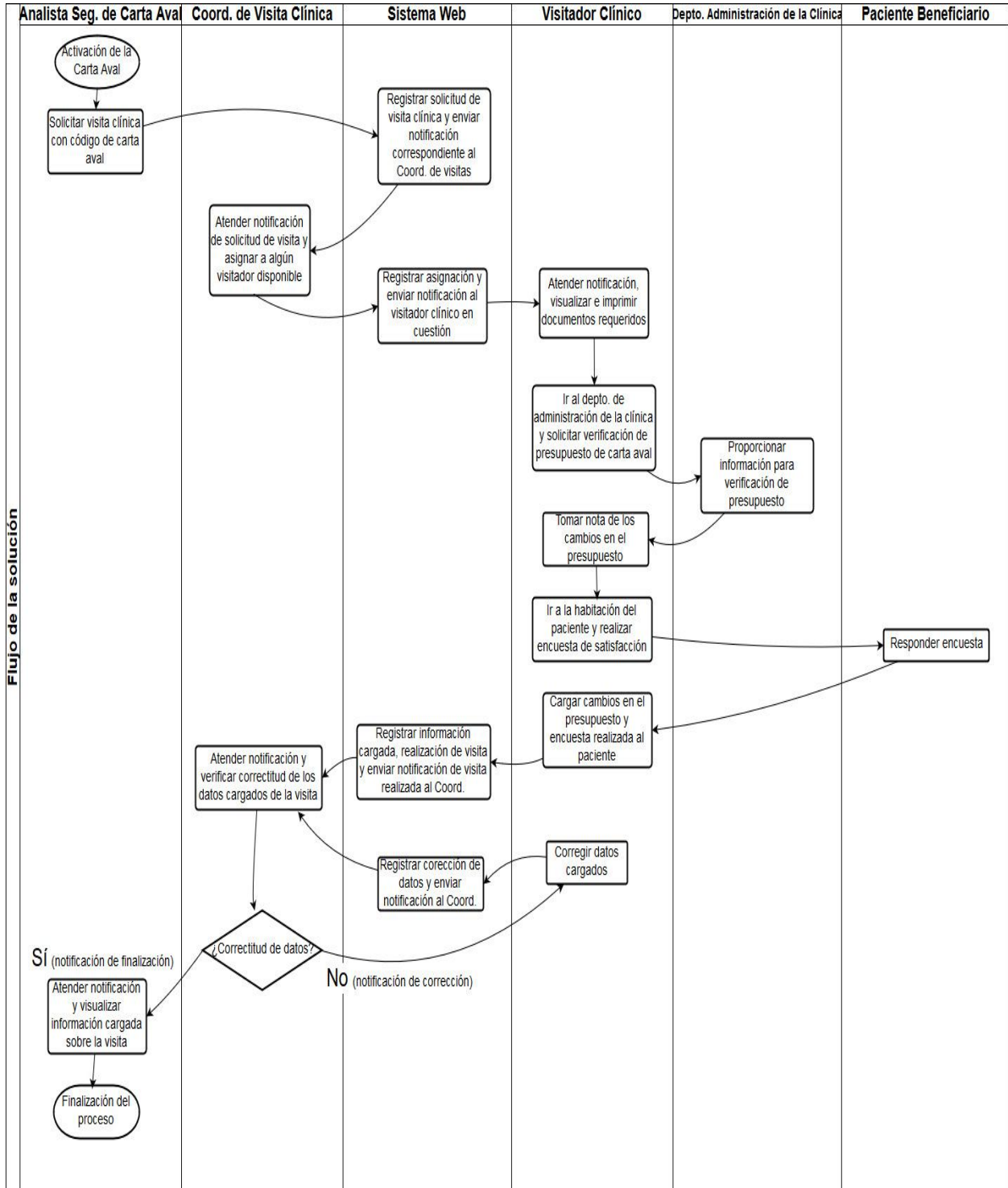


Figura 12. Flujo de la solución. Fuente: Elaboración propia

4.5 Aplicación de la metodología SCRUM

A continuación se presenta el desarrollo de la metodología SCRUM para el desarrollo de la solución. El dueño del producto estuvo representando por el Prof. Franky Uzcátegui y el equipo de desarrollo SCRUM estuvo compuesto por el Br. Argenis Chirinos.

4.5.1 Lista de objetivos (Pila del Producto)

En la tabla 1 se presenta la Pila del Producto que se definió para desarrollo del sistema web. Los sprints estuvieron compuestos por tareas desde la instalación del ambiente de desarrollo y entrenamiento en las diversas herramientas, hasta la implementación final de los módulos requeridos en el sistema. Además, los lapsos de tiempo para cada sprint variaron entre una y seis semanas.

4.5.2 Lista de tareas por iteración (Pila del Sprint)

- **Sprint 1:** Se realizó la instalación de las herramientas requeridas para el desarrollo del sistema web, entre estas se encontraron: NPM (versión), Angular.JS (versión), Node.JS (versión), Python (versión), PostgreSQL (versión), Apache (versión), phpPgAdmin (versión) y finalmente GIT (versión) como software de control de versiones. Además, se realizaron diversas pruebas como creación bases de datos y tablas, creación de esquemas bases de datos y gestionarlos a través del administrador gráfico phpPgAdmin. Igualmente, se desarrollaron pequeñas aplicaciones mediante servicios web en Node.JS (parte

del servidor) que posteriormente fueron consumidos desde el navegador haciendo uso de Angular.JS (parte del cliente).

Sprint	Actividad	Fecha
1	<ul style="list-style-type: none"> • Instalación del ambiente de desarrollo • Entrenamiento en Node.JS 	24/10/16 – 30/10/16
2	<ul style="list-style-type: none"> • Diseño lógico de la base de datos 	31/10/16 – 13/11/16
3	<ul style="list-style-type: none"> • Diseño prototipo de interfaces de usuario • Presentación y aprobación de interfaces de usuario 	14/11/16 – 20/11/16
4	<ul style="list-style-type: none"> • Desarrollo de servicios web para Visitas Clínicas 	21/11/16 – 04/12/16
5	<ul style="list-style-type: none"> • Desarrollo de las vistas del módulo de Carta Aval • Desarrollo de las funcionalidades asociadas a las vistas • Aplicación de pruebas sobre las funcionalidades • Presentación y aprobación del módulo de Carta Aval 	05/12/16 – 15/01/16
6	<ul style="list-style-type: none"> • Desarrollo de las vistas del módulo de Visitas Clínicas • Desarrollo de las funcionalidades asociadas a las vistas • Aplicación de pruebas sobre las funcionalidades • Presentación del módulo de Visitas Clínicas 	16/01/17 – 05/02/17
7	<ul style="list-style-type: none"> • Integración de cambios requeridos en la presentación anterior • Aplicación de pruebas sobre los cambios desarrollados • Presentación y aprobación del módulo de Visitas Clínicas 	06/02/17 – 12/02/17
8	<ul style="list-style-type: none"> • Desarrollo de los módulos administrativos • Aplicación de pruebas sobre las funcionalidades • Presentación de los módulos de administración 	13/02/17 – 26/02/17
9	<ul style="list-style-type: none"> • Integración de cambios requeridos en la presentación anterior • Aplicación de pruebas sobre los cambios desarrollados • Presentación y aprobación de los módulos administrativos 	27/02/17 – 19/03/17

Tabla 1. Pila del producto. Fuente: Elaboración propia

- **Sprint 2:** Se realizó el diseño lógico de la base de datos a ser utilizada y seguidamente se pasó a su implementación en PostgreSQL. Luego de la implementación del diseño de la base de datos, se procedió a cargar datos que permitieran ir haciendo las primeras pruebas de almacenamiento de acuerdo al diseño implementado.
- **Sprint 3:** Se realizaron los primeros prototipos de interfaces de usuario siguiendo en ellas el mismo patrón visual y colores con la intención de que sean intuitivas ante el usuario final del sistema. El desarrollo de estas interfaces fue apoyado haciendo uso del framework de HTML5 Bootstrap, el cual ofrece una gran variedad de elementos visuales que, además de agilizar el desarrollo de las mismas, contribuyen a mejorar la usabilidad del sistema, la interacción humano - computador y la capacitación en el uso del software desarrollado. Estos prototipos fueron presentados y aprobados por el cliente.
- **Sprint 4:** Se desarrollaron los servicios web para implementar la lógica de negocio del sistema y el acceso a la base de datos del mismo. Entre estos servicios web se encuentran los asociados a la sesión del usuario, al módulo de carta aval y al de visitas clínicas. Los servicios web fueron probados haciendo uso de la herramienta Postman, con la intención de verificar el correcto funcionamiento de estos últimos sin necesidad de desarrollar una aplicación cliente, ya que en este Sprint solo se buscaba desarrollar y probar el funcionamiento de los servicios web sobre los cuales se alimentaría posteriormente la aplicación desarrollada en Angular.JS.
- **Sprint 5:** Se desarrollaron las vistas asociadas al módulo de carta aval, siguiendo los lineamientos aprobados en los prototipos de interfaces, con sus respectivas funcionalidades implementadas a través de los servicios web. Una vez desarrollado este módulo se procedió a realizar pruebas sobre el

funcionamiento del mismo y posteriormente se presentó ante el cliente, el cual aprobó el módulo.

- **Sprint 6:** Se desarrollaron las vistas asociadas al módulo de visitas clínicas, siguiendo los lineamientos aprobados en los prototipos de interfaces, con sus respectivas funcionalidades implementadas a través de los servicios web. Una vez desarrollado este módulo se procedió a realizar pruebas sobre el funcionamiento del mismo y posteriormente se presentó ante el cliente, el cual requirió añadir nuevas funcionalidades y hacer correcciones sobre el funcionamiento de las ya desarrolladas.
- **Sprint 7:** Se implementaron los cambios requeridos por el cliente en el módulo de visitas clínicas y además, se agregaron las funcionalidades sobre el mismo. Al finalizar los cambios y las nuevas funcionalidades se procedió a realizar nuevamente pruebas de funcionamiento. Posteriormente se presentó el módulo al cliente y fue aprobado.
- **Sprint 8:** Se desarrollaron las vistas correspondientes al módulo de gestión de datos por parte del administrador del sistema. Una vez finalizado este módulo se realizaron las respectivas pruebas de funcionalidad y se presentó ante el cliente, el cual solicitó añadir nuevas funcionalidades y la corrección de otras ya desarrolladas.
- **Sprint 9:** Se integraron los cambios solicitados por el cliente en el módulo de gestión de datos y se hicieron las pruebas correspondientes. Una vez hecho esto, se presentó nuevamente ante el cliente, el cual aprobó la finalización de este módulo.

4.6 Análisis del modelo de datos y definición

Para el almacenamiento de los datos requeridos para el funcionamiento del sistema web se crearon 27 tablas, las cuales se presentan gráficamente en la figura 13:

4.6.1 Definición de las tablas de la base de datos

- **Clinica:** Tabla utilizada para almacenar los datos correspondientes a las clínicas afiliadas a la compañía aseguradora.
- **Clinica_Email:** Tabla utilizada para almacenar los múltiples correos electrónicos que mantengan las clínicas en cuestión.
- **Clinica_Telefono:** Tabla utilizada para almacenar los múltiples números telefónicos que mantengan las clínicas en cuestión.
- **Respuesta:** Tabla utilizada para almacenar la respuesta a una determinada pregunta hecha en la encuesta de satisfacción del cliente durante la visita clínica.
- **Presupuesto:** Tabla utilizada para almacenar los datos básicos (cabecera) del presupuesto clínico que el asegurado entrega en solicitud de la carta aval.
- **Presupuesto_Imagen:** Tabla utilizada para almacenar la dirección (en disco) de las imágenes cargadas como aval de las modificaciones hechas en el presupuesto durante la visita clínica.
- **Comentario:** Tabla utilizada para almacenar los comentarios y observaciones que se hacen en el sistema en referencia a alguna visita clínica.
- **Formulario:** Tabla utilizada para almacenar los datos básicos (cabecera) de la encuesta de satisfacción del cliente.
- **Carta_Aval:** Tabla utilizada para almacenar la información correspondiente a las cartas avales.

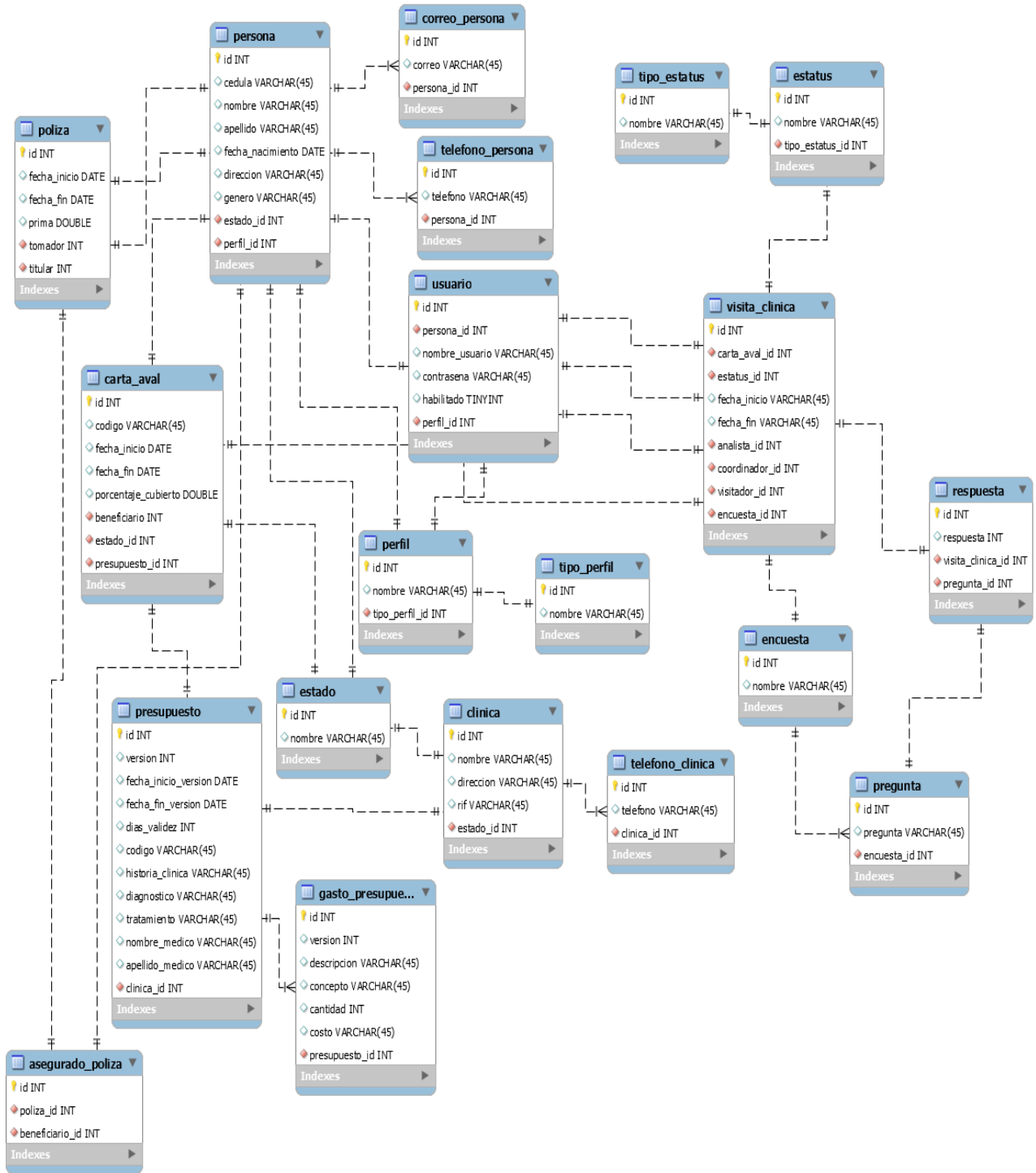


Figura 13. Tablas de la base de datos. Fuente: Elaboración propia

- **Item:** Tabla utilizada para almacenar los distintos “items” o cargos que se facturan en el presupuesto entregado en solicitud de la carta aval.
- **Persona:** Tabla utilizada para almacenar los datos correspondientes a las distintas personas (analistas, coordinadores, visitadores, asegurados).
- **Persona_Email:** Tabla utilizada para almacenar los múltiples correos electrónicos que mantengan las personas en cuestión.
- **Persona_Telefono:** Tabla utilizada para almacenar los múltiples números telefónicos que mantengan las personas en cuestión.
- **Poliza:** Tabla utilizada para almacenar los datos correspondientes a las pólizas de seguro de la compañía aseguradora.
- **Perfil:** Tabla utilizada para almacenar los distintos perfiles de personas y usuarios utilizados en el sistema.
- **Tipo_Perfil:** Tabla utilizada para almacenar los distintos tipos de perfil.
- **Pregunta:** Tabla utilizada para almacenar las distintas preguntas que pertenecen a una encuesta de satisfacción del cliente.
- **Solicitud:** Tabla utilizada para almacenar los datos correspondientes a una solicitud de visita clínica y a la visita clínica en general.
- **Sesion:** Tabla utilizada para almacenar los datos correspondientes a la sesión que mantienen los usuarios en el sistema.
- **Estado:** Tabla utilizada para almacenar los estados nacionales correspondientes a Venezuela.
- **Estatus:** Tabla utilizada para almacenar los distintos estatus que se utilizan a lo largo del uso del sistema.
- **Tipo_Estatus:** Tabla utilizada para almacenar los distintos tipos de estatus que se mantienen en el sistema.
- **Usuario:** Tabla utilizada para almacenar los datos correspondientes a los usuarios del sistema web.

4.7 Servicios web

El sistema web es alimentado a través de servicios web, los cuales fueron desarrollados siguiendo el enfoque REST. A continuación se describen los servicios desarrollados para el sistema:

1. **Nombre:** Autenticación

1.1. **Método HTTP:** POST

1.1.1. Descripción: Servicio que permite al usuario autenticarse o abrir sesión en sistema.

1.1.2. Entrada: Nombre de Usuario y Contraseña.

1.1.3. Salida: “Token” de autenticación e información básica de la persona asociada al usuario (nombre de la persona, cédula y correo).

1.2. **Método HTTP:** DELETE

1.2.1. Descripción: Servicio que permite al usuario cerrar sesión en sistema.

1.2.2. Entrada: “Token de autenticación”.

1.2.3. Salida: Notificación de éxito o fracaso.

2. **Nombre:** Clínicas

2.1. **Método HTTP:** GET

2.1.1. Descripción: Servicio que permite obtener el listado de clínicas registrado en el sistema.

2.1.2. Entrada: Parámetros del filtrado.

2.1.3. Salida: Listado de clínicas registradas en el sistema, según parámetros de filtrado.

3. **Nombre:** Personas

3.1. **Método HTTP:** GET

3.1.1. Descripción: Servicio que permite obtener el listado de personas registradas en el sistema.

3.1.2. Entrada: Parámetros de filtrado.

3.1.3. Salida: Listado de personas registradas en el sistema, según parámetros de filtrado.

4. Nombre: Usuarios

4.1. Método HTTP: GET

4.1.1. Descripción: Servicio que permite obtener el listado de usuarios registrados en el sistema.

4.1.2. Entrada: Parámetros de filtrado.

4.1.3. Salida: Listado de usuarios registrados en el sistema, según parámetros de filtrado.

4.2. Método HTTP: POST

4.2.1. Descripción: Servicio que permite registrar un usuario en el sistema.

4.2.2. Entrada: Nombre de usuario, contraseña, rol de usuario y persona a la que está asociado.

4.2.3. Salida: Notificación de éxito o fracaso.

5. Nombre: Encuestas

5.1. Método HTTP: GET

5.1.1. Descripción: Servicio que permite obtener las encuestas registradas en el sistema.

5.1.2. Entrada: Parámetros de filtrado.

5.1.3. Salida: Listado de encuestas (con preguntas y respuestas), según parámetros de filtrado.

6. Nombre: Presupuestos

6.1. Método HTTP: GET

6.1.1. Descripción: Servicio que permite obtener los presupuestos registrados en el sistema.

6.1.2. Entrada: Parámetros de filtrado.

6.1.3. Salida: Listado de presupuestos, según parámetros de filtrado.

7. Nombre: Carta Aval

7.1. Método HTTP: GET

7.1.1. Descripción: Servicio que permite obtener las cartas avales cargadas en el sistema.

7.1.2. Entrada: Parámetros de filtrado.

7.1.3. Salida: Listado de cartas avales, según parámetros de filtrado.

8. Nombre: Visitas Clínicas

8.1. Método HTTP: GET

8.1.1. Descripción: Servicio que permite obtener las visitas clínicas registradas en el sistema.

8.1.2. Entrada: Parámetros de filtrado.

8.1.3. Salida: Listado de visitas clínicas, según parámetros de filtrado.

8.2. Método HTTP: POST

8.2.1. Descripción: Servicio que permite registrar una visita clínica en el sistema.

8.2.2. Entrada: Carta aval a la que está asociada, analista que la ha ordenado y la fecha límite para realizarla.

8.2.3. Salida: Notificación de éxito o fracaso.

9. Nombre: Comentarios

9.1. Método HTTP: GET

9.1.1. Descripción: Servicio que permite obtener los comentarios u observaciones registradas en una visita clínica.

9.1.2. Entrada: ID de la visita clínica de la cual se quieren obtener los comentarios u observaciones.

9.1.3. Salida: Comentarios u observaciones registradas para dicha visita clínica.

9.2. Método HTTP: POST

9.2.1. Descripción: Servicio que permite registrar un comentario asociado a una visita clínica en el sistema.

9.2.2. Entrada: Comentario y ID de la visita clínica a la que estará asociado el comentario.

9.2.3. Salida: Notificación de éxito o fracaso.

10.Nombre: Cargar Datos

10.1. Método HTTP: POST

10.1.1. Descripción: Servicio que permite cargar datos de diversas entidades en el sistema a través de un archivo .txt.

10.1.2. Entrada: Tipo de entidad sobre la cual se quieren cargar datos y el archivo .txt que contiene los datos propiamente.

10.1.3. Salida: Notificación de éxito o fracaso.

4.8 Descripción de los módulos del sistema y sus interfaces

4.8.1 Descripción de los módulos

El sistema web se encuentra dividido en seis módulos que se describen a continuación y se presentan en la figura 14:

- **Módulo de Carta Aval:** este módulo está dedicado al analista de seguimiento de carta aval y a través de este, el analista podrá visualizar y buscar cualquier carta aval registrada en la base de datos del sistema. Además, podrá visualizar los datos básicos del beneficiario asociado a esta carta aval y de la institución prestadora de servicio. Igualmente podrá visualizar el presupuesto entregado en solicitud de esta carta aval y el documento que representa propiamente a la carta aval.

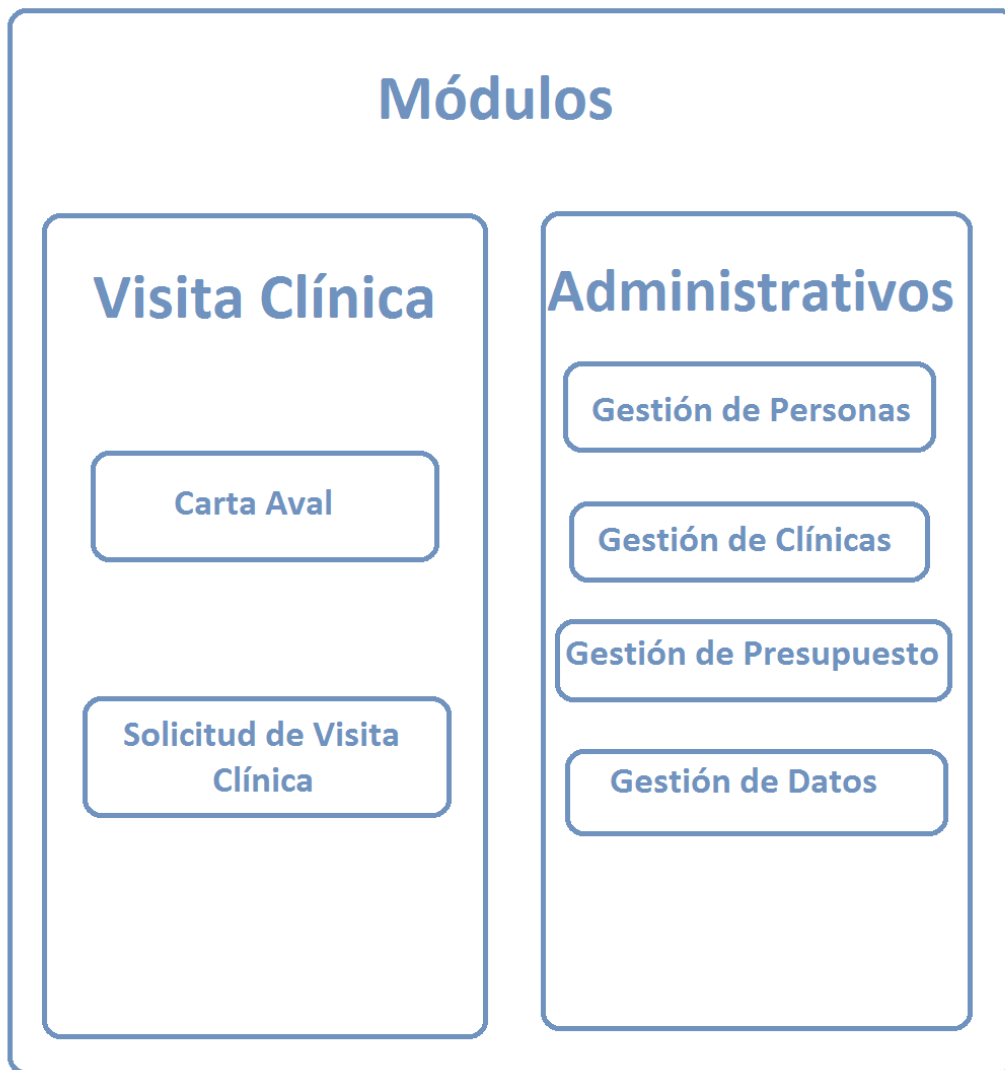


Figura 14. Módulos del Sistema web. Fuente: Elaboración propia

- **Solicitud de Visita Clínica:** este módulo está dedicado al analista de seguimiento de carta aval, al coordinador de visitas clínicas y al visitador clínico. A continuación se mencionan las funciones principales que brinda a cada usuario:

- **Analista de seguimiento de carta aval:** a través de este módulo podrá visualizar la información asociada a las visitas clínicas que ha solicitado y hacer seguimiento de estas según el estatus de las mismas.
- **Coordinador de visitas clínicas:** a través de este módulo podrá visualizar las visitas clínicas que ha coordinado y las que todavía no han sido coordinadas. Podrá hacer seguimiento sobre las visitas clínicas según el estatus de las mismas y además, podrá visualizar la información, cargada por el visitador clínico, de cada visita clínica.
- **Visitador clínico:** a través de este módulo podrá visualizar las visitas clínicas que se le han asignado, podrá hacer seguimiento sobre estas según el estatus de las misma, podrá visualizar la información del presupuesto y carta aval asociada a las solicitudes de visitas, y podrá cargar la información obtenida durante el desarrollo de la visita clínica.
- **Personas:** este módulo está dedicado al administrador del sistema. A través del mismo se podrá gestionar la información referente a las personas que están registradas en la base de datos del sistema.
- **Clínicas:** este módulo está dedicado al administrador del sistema. A través del mismo se podrá gestionar la información referente a las clínicas o instituciones prestadoras de servicio que se encuentran afiliadas a la compañía aseguradora.
- **Presupuestos:** este módulo está dedicado al administrador del sistema. A través de este módulo se podrá gestionar la información referente a los presupuestos entregados en solicitud de la carta aval.

- **Datos:** este módulo está dedicado al administrador del sistema. A través de este módulo se podrán cargar datos acerca de las personas, clínicas, cartas avales y presupuestos mediante archivos en formato “.txt”.

4.8.2 Interfaces del sistema web

- **Interfaz correspondiente al inicio de sesión:** En la figura 15 se presenta la interfaz asociada al inicio de sesión en el sistema. En esta vista el usuario introduce su nombre de usuario y contraseña para realizar la autenticación. En la misma se encuentra la opción de “Recuperar Contraseña” en caso de que el usuario la haya olvidado.



Figura 15. Inicio de sesión. Fuente: Captura de pantalla

- **Interfaz inicial del Analista de Seguimiento de Carta Aval:** En la figura 16 se presenta la primera interfaz que visualiza el analista al iniciar sesión en el sistema. Esta muestra las cartas avales que se encuentran registradas en base de datos y en la parte izquierda se despliega el menú con los módulos con que el analista interactúa.

Código Carta Aval	Estatus	Expide	Vence	CI Beneficiario	Nombre Beneficiario	Acciones
1	activada	4/4/2017	14/2/2018	99182712	JOSE PEREZ	Acción
2	activada	4/4/2017	14/2/2018	76281921	CARLOS BRACAMONTE	Acción
3	activada	4/4/2017	14/2/2018	10291827	JOSEFINA SANCHEZ	Acción
4	activada	4/4/2017	14/2/2018	28371623	PAOLA VERA	Acción
5	activada	4/4/2017	14/2/2018	11829182	ESTRELLA COLMENARES	Acción
6	activada	4/4/2017	14/2/2018	28121623	JULIA RAMIREZ	Acción

Figura 16. Listado de cartas avals. Fuente: Captura de pantalla

- **Interfaz (modal) correspondiente a la búsqueda de cartas avals:** En la figura 17 se presenta la interfaz utilizada para llevar a cabo la búsqueda de una determinada carta aval según los parámetros de búsqueda establecidos.

Figura 17. Búsqueda de cartas avals. Fuente: Captura de pantalla

- **Interfaz correspondiente a la visualización de la información de una carta aval:** En la figura 18 se presenta la interfaz en la cual se visualiza la información correspondiente a una determinada carta. En ella se permite también la descarga en formato PDF del presupuesto y la carta aval.

Detalles de la Carta Aval de Código: 1

Datos de Carta Aval

Estatus: activada Código Carta Aval: 1 Expide: 4/4/2017 Vence: 14/2/2018 [Solicitar Visita](#)

[Descargar Presupuesto](#) [Descargar Carta Aval](#)

Datos de la Institución Prestadora de Servicio

Nombre: CLINICA CENTRAL DE CARABOBO RIF: J-00168870-8 Estado: CARABOBO

Teléfonos: 0237-6729018 Email: CENTRO.CARABOBO@EMAIL.COM

Dirección: AV. PRINCIPAL DE JOSE ANTONIO ORTIZ. CARABOBO

Datos del Beneficiario

Nombre: JOSE PEREZ Cédula: 99182712 Género: MASCULINO

Edad: 47 Teléfonos: 0212-6729032 Email: J.PEREZ@EMAIL.COM

Dirección: AV. URDANETA, EDIFICIO ROSALINDA, PISO 01, APTO 01. URB EL ROSAL. CARACAS

Figura 18. Detalles de carta aval. Fuente: Captura de pantalla

- **Interfaz (modal) correspondiente a la solicitud de visita clínica para una determinada carta aval:** En la figura 19 se presenta la interfaz para solicitar una visita. En ella se solicita la fecha límite para realizar la visita clínica y un campo para indicar las observaciones que sean necesarias sobre esta solicitud.



Figura 19. Ingreso de fecha límite para realizar visita clínica. Fuente: Captura de pantalla

- **Interfaz siguiente luego de la generación de una solicitud de visita clínica:**
En la figura 20 se presenta la interfaz siguiente luego de solicitar una visita clínica. En esta se informa sobre la exitosa generación de la misma y se actualiza la parte del menú correspondiente a las solicitudes de visitas clínicas.



Figura 20. Notificación de visita clínica solicitada. Fuente: Captura de pantalla

- **Interfaz correspondiente a las visitas clínicas solicitadas:** En la figura 21 se presenta la interfaz donde se listan las visitas clínicas solicitadas junto con el código de visita, el código de la carta aval a la que está asociada, la fecha de solicitud y su estatus actual.

VisitadorClínico Ubicación: DISTRITO CAPITAL MARIA

Búsqueda de Solicitud de Visita

Solicitudes de Visitas - Todas

« 1 »

Código de Visita	Código Carta Aval	Fecha Solicitud	Estatus	Acción
501	1	5/4/2017	por asignar	Ver

Figura 21. Listado de visitas clínicas. Fuente: Captura de pantalla

- **Interfaz (modal) correspondiente a la búsqueda de visitas clínicas:** En la figura 22 se presenta la interfaz a través de la cual se pueden buscar visitas clínicas de acuerdo a los parámetros de búsqueda.

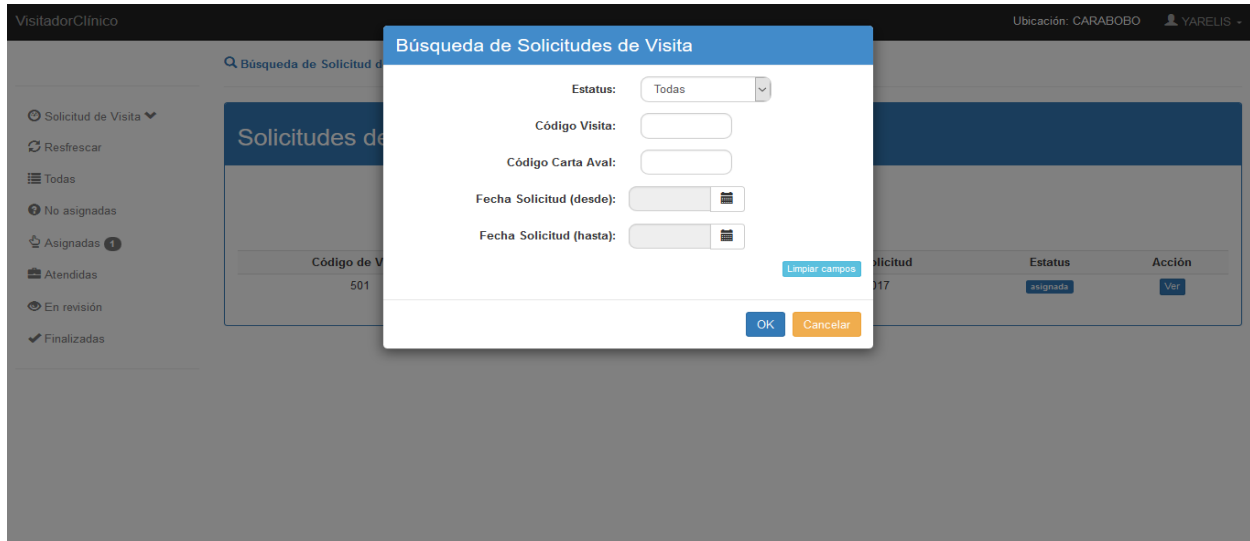


Figura 22. Búsqueda de visitas clínicas. Fuente: Captura de pantalla

- **Interfaz correspondiente a la visualización de una visita clínica:** En la figura 23 se presenta la interfaz donde visualiza la información básica de la solicitud de visita clínica, dando la posibilidad de hacer la asignación de esta visita a alguno de sus visitadores clínicos.

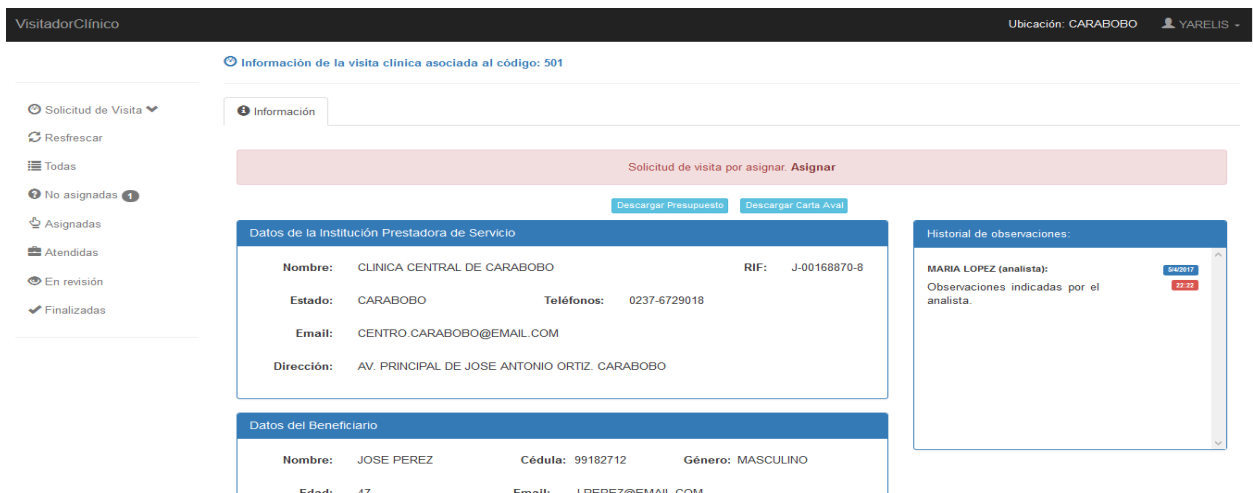


Figura 23. Detalles de visita clínica. Fuente: Captura de pantalla

- **Interfaz (modal) correspondiente a la asignación de una visita clínica:** En la figura 24 se presenta la interfaz donde se hace la asignación de la visita clínica a alguno de los visitadores clínicos. En la interfaz se visualiza la cédula del visitador, nombre y cantidad de visitas que tiene actualmente asignadas.

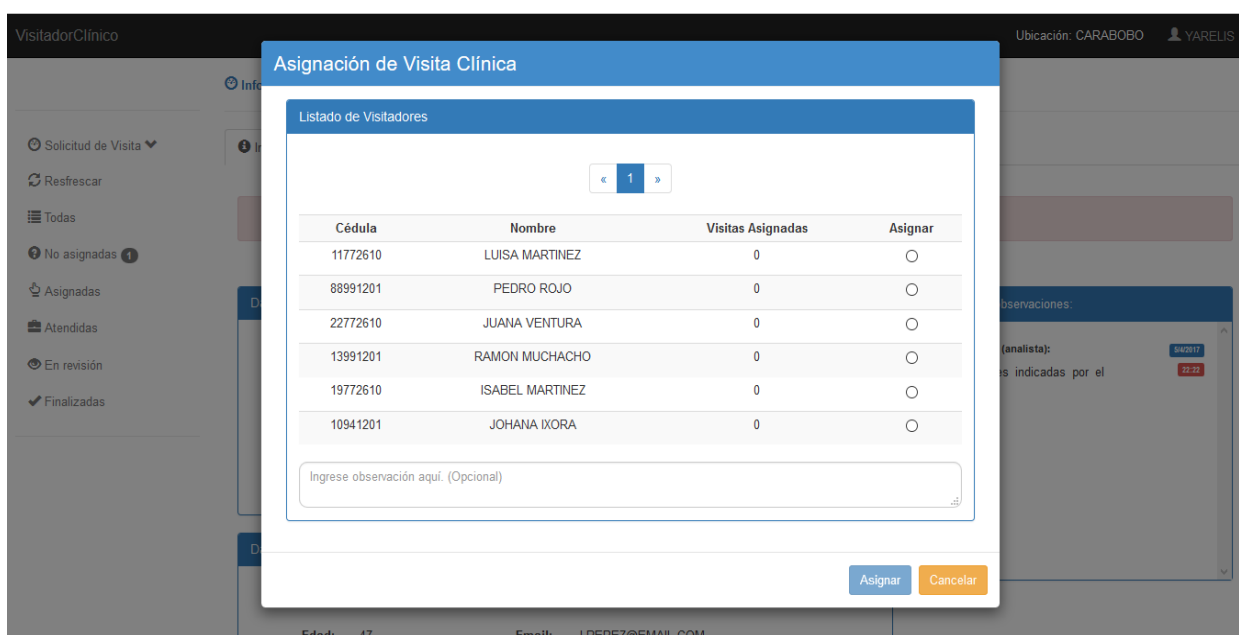


Figura 24. Listado de visitadores clínicos para asignar visita. Fuente: Captura de pantalla

- **Interfaz siguiente luego de la asignación de una visita clínica:** En la figura 25 se presenta la interfaz siguiente luego de haber asignado una visita clínica. En ella se notifica la asignación exitosa de la visita y además ofrece la posibilidad de cambiar la asignación realizada.

VisitadorClínico

✓ Listo
Asignación hecha con éxito.

Información de la visita clínica asociada al código: 501

Solicitud de Visita

Resfreshar

Todas

No asignadas

Asignadas 1

Atendidas

En revisión

Finalizadas

Información

Solicitud de visita asignada. Cambiar Asignación

Descargar Presupuesto Descargar Carta Aval

Datos de la Institución Prestadora de Servicio

Nombre: CLINICA CENTRAL DE CARABOBO RIF: J-00168870-8

Estado: CARABOBO Teléfonos: 0237-6729018

Email: CENTRO.CARABOBO@EMAIL.COM

Dirección: AV. PRINCIPAL DE JOSE ANTONIO ORTIZ, CARABOBO

Datos del Beneficiario

Nombre: JOSE PEREZ Cédula: 99182712 Género: MASCULINO

Edad: 47 Email: J.PEREZ@EMAIL.COM

Historial de observaciones:

MARIA LOPEZ (analista): 5/4/2017
Observaciones indicadas por el analista. 22:32

YARELIS CORREA (coordinador): 5/4/2017
Observaciones del coordinador. 23:15

Figura 25. Notificación de visita asignada. Fuente: Captura de pantalla

- Interfaz correspondiente a la Información requerida para llevar a cabo la visita clínica:** En la figura 26 se presenta la interfaz a través de la cual el visitador clínico puede obtener la información necesaria para realizar la visita. Esta interfaz se puede visualizar la información asociada a la clínica o institución prestadora de servicio y del beneficiario. Además, en esta pestaña se tiene la posibilidad de descargar en formato PDF el presupuesto de carta aval, la carta aval y la encuesta de satisfacción del paciente.

VisitorClínico Ubicación: CARABOBO | JOHANA

Información de la visita clínica asociada al código: 501

Solicitud de Visita ▾
 Resfreshar
 Todas
 Por atender 1
 En revisión
 Finalizadas

Información | Presupuesto | Encuesta | Adjuntos

Solicitud de visita pendiente. Finalizar

Descargar Presupuesto | Descargar Carta Aval | Descargar Encuesta

Datos de la Institución Prestadora de Servicio		
Nombre:	CLINICA CENTRAL DE CARABOBO	RIF: J-00168870-8
Estado:	CARABOBO	Teléfonos: 0237-6729018
Email:	CENTRO.CARABOBO@EMAIL.COM	
Dirección:	AV. PRINCIPAL DE JOSE ANTONIO ORTIZ, CARABOBO	

Datos del Beneficiario		
Nombre:	JOSE PEREZ	Cédula: 99182712 Género: MASCULINO
Edad:	47	Email: J.PEREZ@EMAIL.COM

Historial de observaciones:

- MARIA LOPEZ (analista): 5/4/2017 22:25
Observaciones indicadas por el analista.
- YARELIS CORREA (coordinador): 5/4/2017 23:15
Observaciones del coordinador.

Figura 26. Información para realizar visita clínica. Fuente: Captura de pantalla

- Interfaz correspondiente a la información del presupuesto:** En la figura 27 se presenta la interfaz correspondiente a la pestaña de presupuesto. Esta pestaña se encuentra a su vez compuesta por dos sub-pestañas que son: Presupuesto Original y Presupuesto Modificado. En la pestaña de Presupuesto Original se puede visualizar el presupuesto entregado en solicitud de la carta aval, el cual es el presupuesto sobre el cual se pueden hacer cambios según la información obtenida durante la visita clínica.

VisitadorClínico Ubicación: CARABOBO JOHANA

Información de la visita clínica asociada al código: 501

Información | Presupuesto | Encuesta | Adjuntos

Presupuesto

Presupuesto Original | Presupuesto Modificado

No.	Concepto	Descripción	Cantidad	Costo Bs (unitario)
1	GASTOS HOSPITALIZACION	MEDICINAS	1	13.000
2	GASTOS HOSPITALIZACION	MATERIAL MÉDICO QUIRURGICO	1	100.000
3	REQUERIMIENTOS	RX TORAX (2P)	2	1.200
4	REQUERIMIENTOS	ELECTROCARDIOGRAMA	2	300,9
5	SERVICIOS CLINICOS	CARDIOLOGO	1	190.000
6	SERVICIOS CLINICOS	MEDICO PERFUSIONISTA	1	200.001
7	SERVICIOS CLINICOS	ANESTESIOLOGO	1	200.000
8	SERVICIOS CLINICOS	LABORATORIO CLINICO	2	10.000
9	SERVICIOS CLINICOS	BANCO DE SANGRE BAZETTI	1	20.000

Figura 27. Información del presupuesto original. Fuente: Captura de pantalla

- Interfaz correspondiente a la sub-pestaña Presupuesto Modificado:** En la figura 28 se presenta la interfaz para hacer cargar los cambios hechos sobre el presupuesto original. En esta pestaña se tiene básicamente la funcionalidad de editar los gastos del presupuesto de acuerdo a la información obtenida durante la visita clínica. Los cambios que se realizan sobre el presupuesto generan nuevas versiones del mismo y se tiene la posibilidad de visualizar las versiones anteriores del presupuesto partiendo desde la versión original del mismo.

No.	Concepto	Descripción	Cantidad	Costo Bs (unitario)	Acción
1	GASTOS HOSPITALIZACION	MEDICINAS	1	13.000	Editar
2	GASTOS HOSPITALIZACION	MATERIAL MÉDICO QUIRURGICO	1	100.000	Editar
3	REQUERIMIENTOS	RX TORAX (2P)	2	1.200	Editar
4	REQUERIMIENTOS	ELECTROCARDIOGRAMA	2	300,9	Editar
5	SERVICIOS CLINICOS	CARDIOLOGO	1	190.000	Editar
6	SERVICIOS CLINICOS	MEDICO PERFUSIONISTA	1	200.001	Editar
7	SERVICIOS CLINICOS	ANESTESIOLOGO	1	200.000	Editar
8	SERVICIOS CLINICOS	LABORATORIO CLINICO	2	10.000	Editar
9	SERVICIOS CLINICOS	BANCO DE SANGRE RAZETTI	1	20.800	Editar
10	SERVICIOS CLINICOS	UNIDAD QUIRURGICA VASCULAR	1	17.500	Editar
11	SERVICIOS CLINICOS	SERVICIO INTEGRAL DE PABELLON	1	7.500	Editar

Figura 28. Información de las modificaciones sobre el presupuesto. Fuente: Captura de pantalla

- **Interfaz (modal) correspondiente a la edición de un gasto del presupuesto:**
 En la figura 29 se presenta la interfaz a través de la cual se pueden editar los gastos del presupuesto. En ella se pueden editar el concepto de gasto, el nombre del gasto la cantidad y el monto del mismo.

Figura 29. Edición de gasto del presupuesto. Fuente: Captura de pantalla

- **Interfaz correspondiente a las modificaciones hechas sobre el presupuesto:** En la figura 30 se presenta la interfaz en la cual se pueden visualizar los gastos del presupuesto que han sido modificados (marcados en color rojo) para luego cargar los mismos.

No.	Concepto	Descripción	Cantidad	Costo Bs (unitario)	Acción
1	GASTOS HOSPITALIZACION	MEDICINAS	1	13.000	Editar
2	GASTOS HOSPITALIZACION	MATERIAL MÉDICO QUIRURGICO	1	4.567	Editar
3	REQUERIMIENTOS	RX TORAX (2P)	2	1.200	Editar
4	REQUERIMIENTOS	ELECTROCARDIOGRAMA	2	8.765	Editar
5	SERVICIOS CLINICOS	CARDIOLOGO	1	190.000	Editar
6	SERVICIOS CLINICOS	MEDICO PERFUSIONISTA	1	200.001	Editar
7	SERVICIOS CLINICOS	ANESTESIOLOGO	1	12.000	Editar
8	SERVICIOS CLINICOS	LABORATORIO CLINICO	2	10.000	Editar

Figura 30. Gastos del presupuesto modificado. Fuente: Captura de pantalla

- **Interfaz correspondiente a la pestaña Encuesta:** en la figura 31 se presenta la interfaz para realizar la carga de las respuestas del paciente sobre la encuesta de satisfacción. Cada carga de respuestas generará una nueva versión partiendo desde la hecha por primera vez.

No.	Aspecto a Calificar	Calificación
1	IMPRESION GENERAL SOBRE LA INSTITUCION PRESTADORA DE SERVICIO	<input checked="" type="radio"/> Excelente <input type="radio"/> Bueno <input type="radio"/> Regular <input type="radio"/> Malo <input type="radio"/> Muy malo
2	ATENCION RECIBIDA POR PARTE DEL MEDICO TRATANTE	<input checked="" type="radio"/> Excelente <input type="radio"/> Bueno <input type="radio"/> Regular <input type="radio"/> Malo <input type="radio"/> Muy malo
3	ATENCION RECIBIDA POR PARTE DEL PERSONAL DE ENFERMERIA	<input checked="" type="radio"/> Excelente <input type="radio"/> Bueno <input type="radio"/> Regular <input type="radio"/> Malo <input type="radio"/> Muy malo
4	INFORMACION OFRECIDA POR EL PERSONAL CLINICO	<input checked="" type="radio"/> Excelente <input type="radio"/> Bueno <input type="radio"/> Regular <input type="radio"/> Malo <input type="radio"/> Muy malo
5	DOTACION DE EQUIPOS MEDICOS E INSTRUMENTAL	<input type="radio"/> Excelente <input checked="" type="radio"/> Bueno <input type="radio"/> Regular <input type="radio"/> Malo <input type="radio"/> Muy malo
6	LIMPIEZA DE ASEOS Y ZONAS COMUNES	<input type="radio"/> Excelente <input checked="" type="radio"/> Bueno <input type="radio"/> Regular <input type="radio"/> Malo <input type="radio"/> Muy malo
7	COMODIDAD DE LAS INSTALACIONES EN GENERAL	<input type="radio"/> Excelente <input checked="" type="radio"/> Bueno <input type="radio"/> Regular <input type="radio"/> Malo <input type="radio"/> Muy malo

[Cargar](#)
Las nuevas cargas sustituirán las anteriores

Figura 31. Carga de respuestas de la encuesta. Fuente: Captura de pantalla

- **Interfaz correspondiente a la pestaña Adjuntos:** en la figura 32 se presenta la interfaz para cargar los documentos digitales que avalen los datos cargados sobre el presupuesto y la encuesta de satisfacción del cliente.



Figura 32. Carga de documentos digitales. Fuente: Captura de pantalla

- **Interfaz correspondiente a la finalización de la visita por parte del visitador clínico:** En la figura 33 se presenta la interfaz siguiente luego de que el visitador clínico hace todos los cambios sobre el presupuesto, carga las respuestas de la encuesta y los documentos digitales. Después de esto, debe esperar a que el coordinador tome la decisión de autorizar u ordenar una revisión sobre la visita.

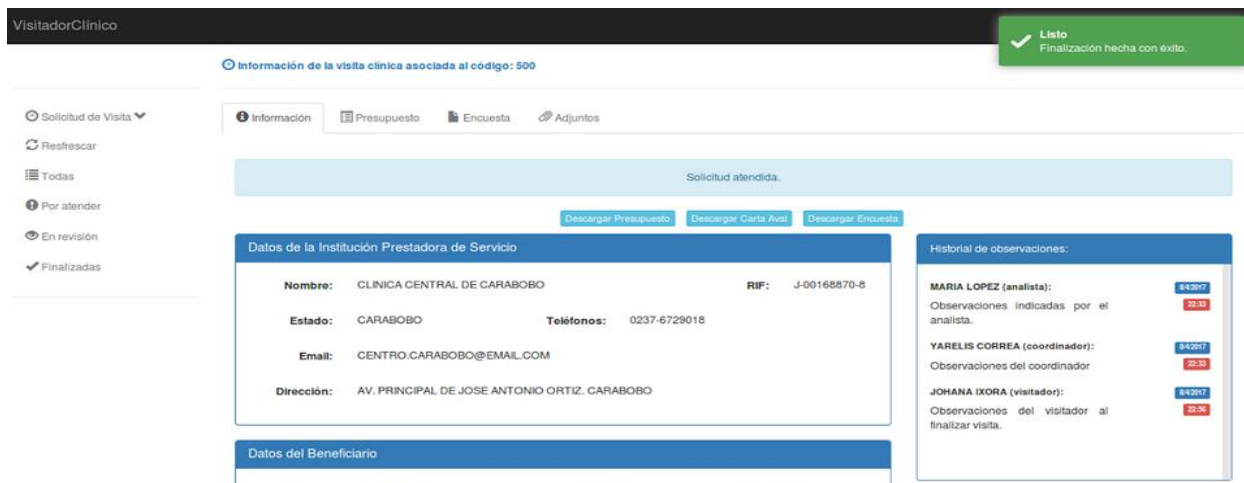


Figura 33. Notificación de visita clínica finalizada. Fuente: Captura de pantalla

- **Interfaz correspondiente a la decisión que toma el coordinador clínico sobre autorizar u ordenar una revisión sobre la visita clínica finalizada:** En la figura 34 se presenta la interfaz en la que el coordinador clínico toma la decisión de autorizar u ordenar revisión. Para ello se encuentran las opciones “Autorizar” y “Revisar” respectivamente.

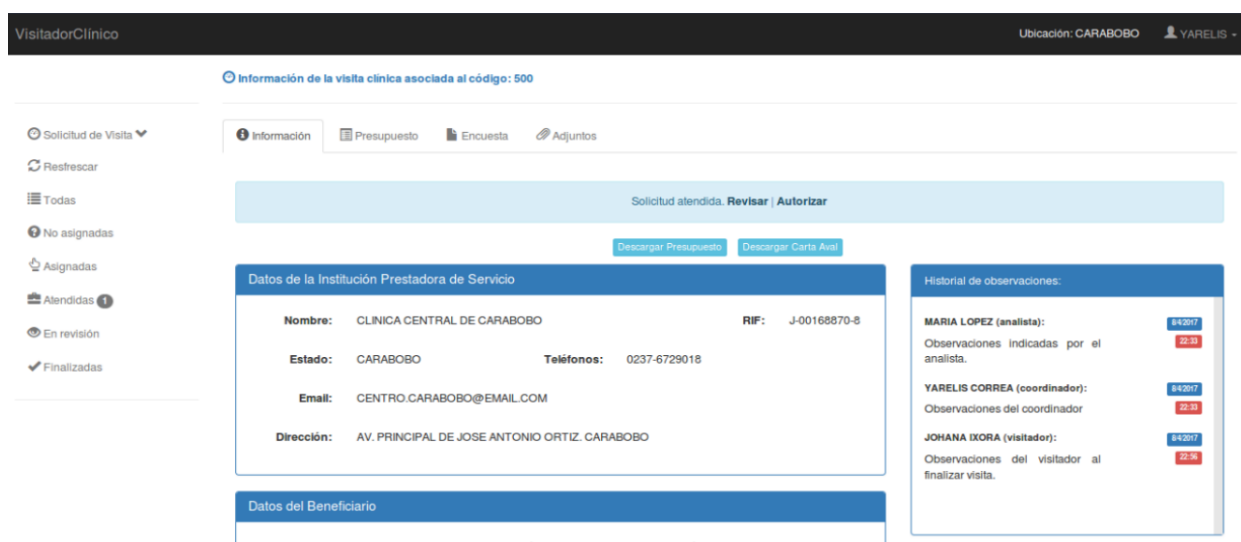


Figura 34. Toma de decision sobre ordenar revisión o dar finalizada la visita clínica. Fuente: Captura de pantalla

- **Interfaz correspondiente a una visita clínica finalizada (autorizada):** En la figura 35 se presenta la interfaz de una visita clínica autorizada. En esta interfaz el analista de seguimiento de carta aval puede visualizar toda la información asociada a la visita, incluyendo los cambios en el presupuesto, las respuestas de la encuesta y los documentos digitales cargados por el visitador clínico.

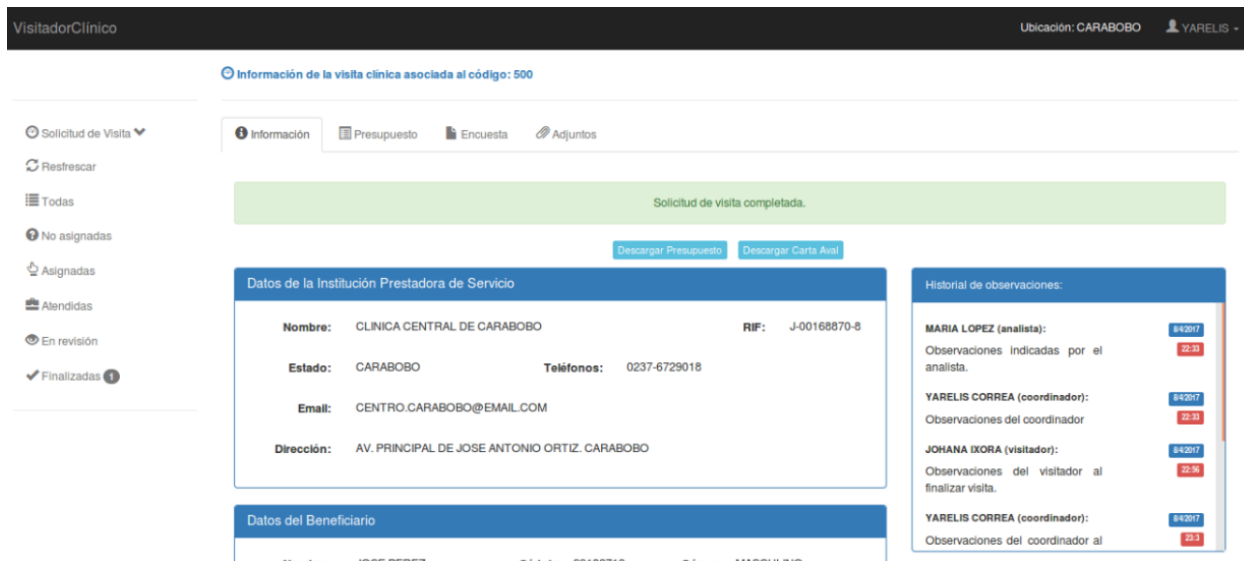


Figura 35. Notificación de visita clínica dada por finalizada. Fuente: Captura de pantalla

- **Interfaz (modal) correspondiente al cambio de nombre de usuario:** En la figura 36 se presenta la interfaz en la cual los usuarios del sistema pueden cambiar su nombre de usuario por otro de su preferencia.



Figura 36. Cambio de nombre de usuario. Fuente: Captura de pantalla

- **Interfaz (modal) correspondiente al cambio de contraseña del usuario:** En la figura 37 se presenta la interfaz en la que los usuarios pueden modificar su contraseña por alguna de su preferencia.

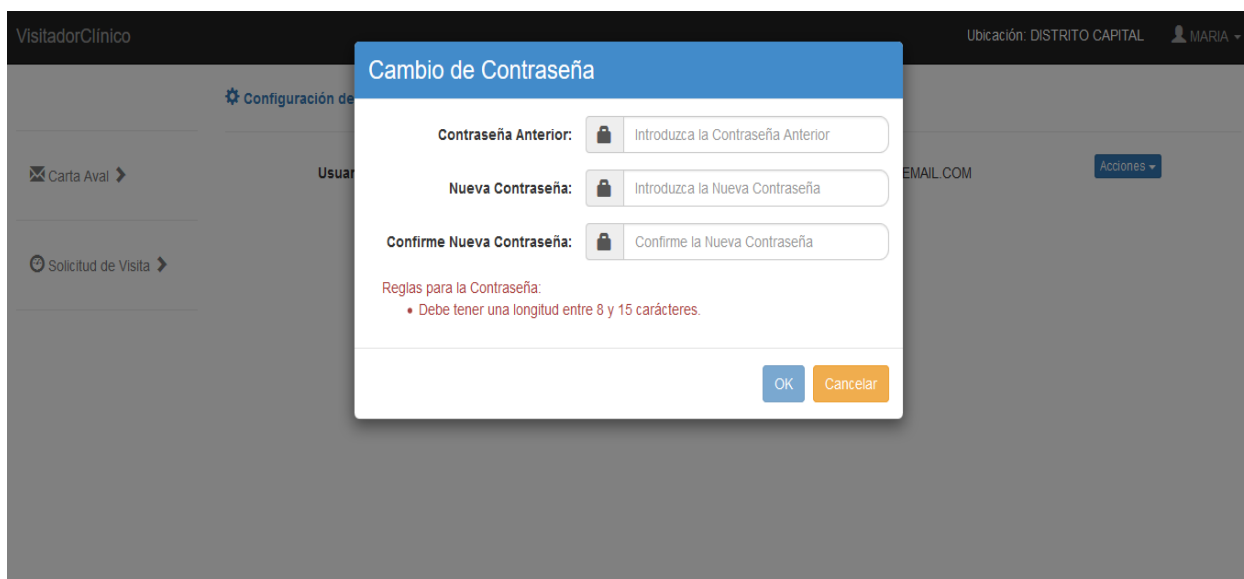
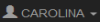


Figura 37. Cambio de contraseña. Fuente: Captura de pantalla

- **Interfaz inicial del administrador:** En la figura 38 se presenta la interfaz inicial del administrador. En esta interfaz inicial se listan las personas registradas en el sistema y que se ven envueltas en el proceso de visitas clínicas a través del uso del sistema web, es decir, analistas, coordinador, visitador y también los administradores.

VisitadorClínico Ubicación: CARABOBO  CAROLINA

Búsqueda de Personas

Personas

« 1 2 3 4 »

CI	Nombre	Cargo	Estado	Acción
10503817	MARIA LOPEZ	ANALISTA	DISTRITO CAPITAL	Ver
99182712	JOSE PEREZ	ANALISTA	DISTRITO CAPITAL	Ver
76281921	CARLOS BRACAMONTE	COORDINADOR	DISTRITO CAPITAL	Ver
10291827	JOSEFINA SANCHEZ	COORDINADOR	DISTRITO CAPITAL	Ver
28371623	PAOLA VERA	VISITADOR	DISTRITO CAPITAL	Ver
11829182	ESTRELLA COLMENARES	VISITADOR	DISTRITO CAPITAL	Ver

Figura 38. Listado de personas. Fuente: Captura de pantalla

- **Interfaz (modal) correspondiente a la búsqueda de personas:** En la figura 39 se presenta la interfaz a través de la cual se puede hacer la búsqueda de una persona según los parámetros de búsqueda.

Búsqueda de Personas

Cédula:

Nombre:

Apellido:

Cargo:

Ubicación:

[Limpiar campos](#)

[OK](#) [Cancelar](#)

CI	Nombre	Cargo	Estado
10503817	MARIA LOPEZ	ANALISTA	DISTRITO CAPITAL
99182712	JOSE PEREZ	ANALISTA	DISTRITO CAPITAL
76281921	CARLOS BRACAMONTE	COORDINADOR	DISTRITO CAPITAL
10291827	JOSEFINA SANCHEZ	COORDINADOR	DISTRITO CAPITAL
28371623	PAOLA VERA	VISITADOR	DISTRITO CAPITAL

Figura 39. Búsqueda de personas. Fuente: Captura de pantalla

- **Interfaz correspondiente a la visualización de los detalles de una persona:**
En la figura 40 se presenta la interfaz a través de la cual se pueden visualizar los detalles de una persona en particular. Además, en esta interfaz se podrán modificar datos del usuario asociado a la persona en caso de tenerlo. En caso de no tener un usuario asociado permite la creación de un usuario.

The screenshot shows the 'Visitador Clínico' application interface. At the top, it displays 'Ubicación: CARABOBO' and the user 'CAROLINA'. The main content area is titled 'Detalles de la persona:' and contains two data panels. The first panel, 'Datos de la persona', lists: Cédula: 10503817, Nombre: MARIA LOPEZ, Cargo: ANALISTA, Edad: 47, Género: FEMENINO, Email: M.LOPEZ@EMAIL.COM, and Teléfonos: 0212-6729054. The second panel, 'Datos del usuario asociado a la persona', lists: Usuario: MLOPEZ, Estatus: habilitado, and Rol: ANALISTA. A sidebar on the left contains navigation options: Personas, Clínicas, Presupuestos, and Datos.

Figura 40. Detalles de persona. Fuente: Captura de pantalla

- **Interfaz (modal) correspondiente a la edición de datos de una persona:** En la figura 41 se presenta la interfaz a través de la cual se pueden editar los datos de las personas, por ejemplo: nombre, apellido, cédula, teléfonos y correos electrónicos.

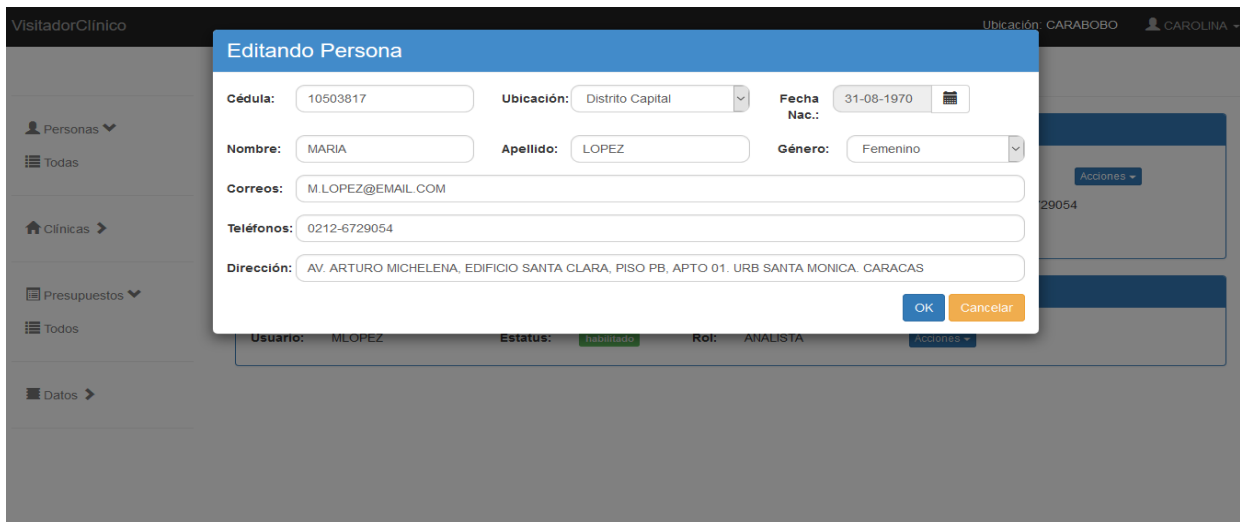


Figura 41. Edición de datos de una persona. Fuente: Captura de pantalla

- **Interfaz correspondiente al listado de clínicas afiliadas a la compañía aseguradora:** En la figura 42 se presenta la interfaz donde se listan todas las clínicas que están afiliadas a la compañía aseguradora.

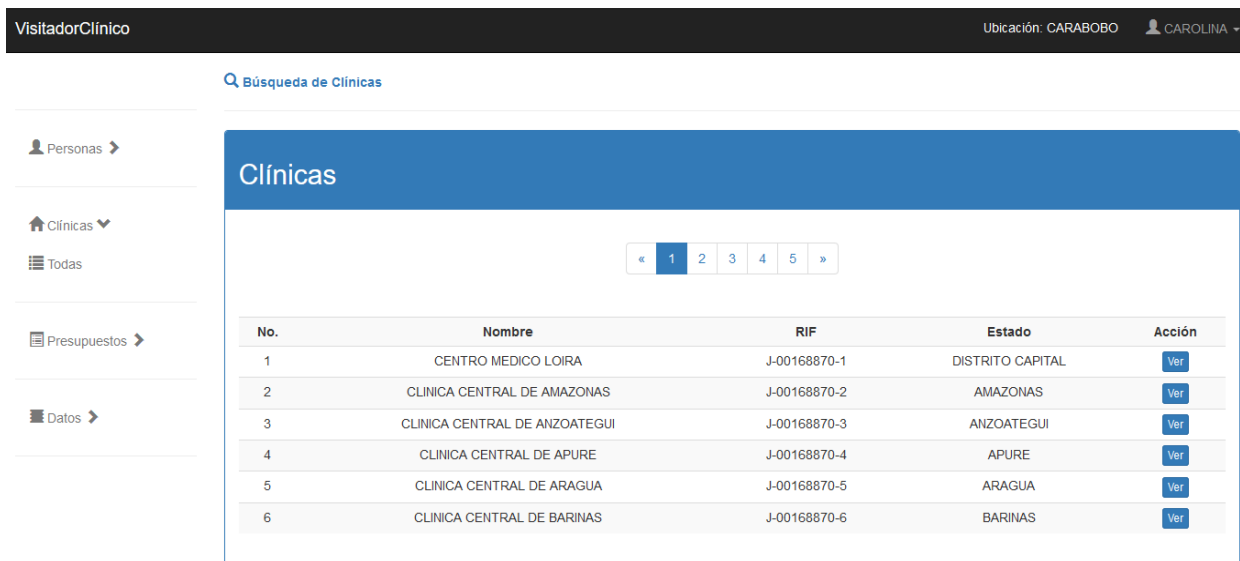


Figura 42. Listado de clínicas. Fuente: Captura de pantalla

- **Interfaz (modal) correspondiente a la búsqueda de clínicas:** En la figura 43 se presenta la interfaz a través de la cual se pueden buscar clínicas según los parámetros de búsqueda.



Figura 43. Búsqueda de clínicas. Fuente: Captura de pantalla

- **Interfaz correspondiente a la visualización de los detalles de una clínica:** En la figura 44 se presenta la interfaz a través de la cual se pueden visualizar los datos de una clínica, entre ellos: nombre, RIF, dirección y ubicación.



Figura 44. Detalles de clínicas. Fuente: Captura de pantalla

- **Interfaz (modal) correspondiente a la edición de datos de una clínica:** En la figura 45 se presenta la interfaz a través de la cual se pueden editar los datos de una clínica.

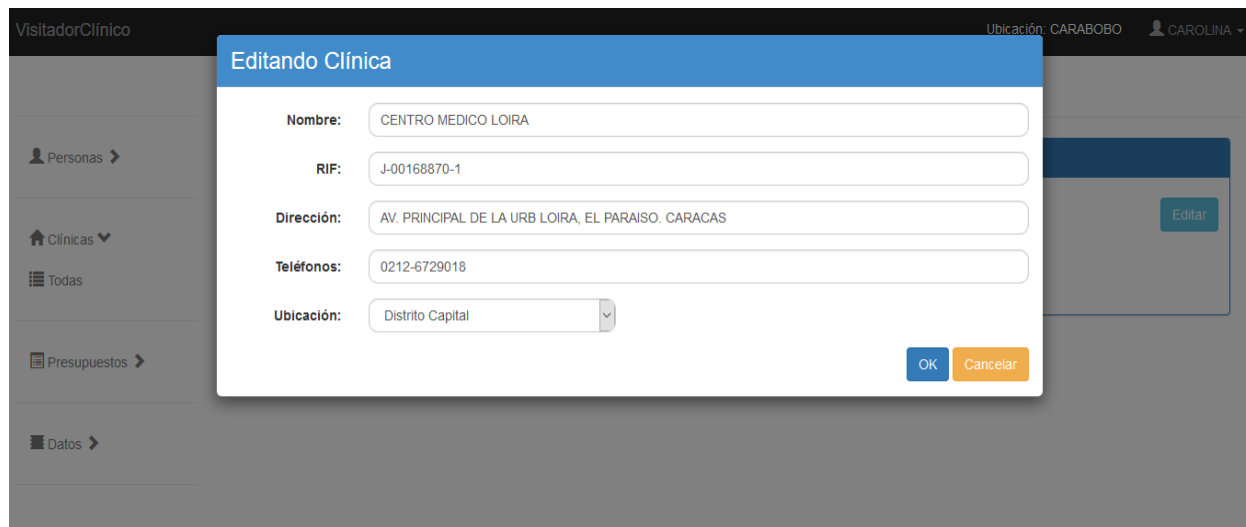


Figura 45. Edición de datos de una clínica. Fuente: Captura de pantalla

- **Interfaz correspondiente al listado de presupuestos cargados en el sistema:** En la figura 46 se presenta la interfaz en la cual se listan todos los presupuestos cargados en el sistema entregados en solicitud de carta aval.

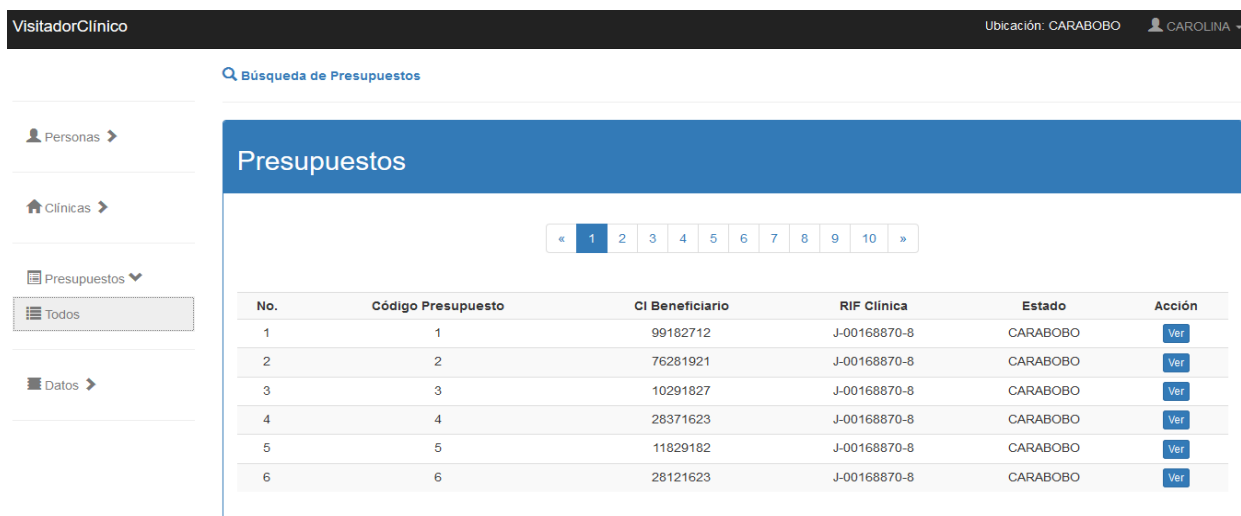


Figura 46. Listado de presupuestos. Fuente: Captura de pantalla

- **Interfaz (modal) correspondiente a la búsqueda de presupuestos:** En la figura 47 se presenta la interfaz a través de la cual se pueden buscar presupuestos según los parámetros de búsqueda definidos.




Figura 47. Búsqueda de presupuestos. Fuente: Captura de pantalla

- **Interfaz correspondiente a la visualización de los detalles de un presupuesto:** En la figura 48 se presenta la interfaz a través de la cual se pueden ver los detalles de un presupuesto. Esta interfaz se encuentra compuesta por 2 pestañas: Información y Presupuesto.



Figura 48. Detalles de presupuesto. Fuente: Captura de pantalla

- **Interfaz correspondiente a la edición del presupuesto:** En la figura 49 se presenta la interfaz a través de la cual se puede modificar el presupuesto en caso de que haya habido algún error al momento de cargarlo en el sistema.

VisitaadorClínico Ubicación: CARABOBO  CAROLINA

Detalles del Presupuesto:

Personas > Presupuesto

No.	Concepto	Descripción	Cantidad	Costo Bs (unitario)	Acción
1	GASTOS HOSPITALIZACION	MEDICINAS	1	13.000	Editar
2	GASTOS HOSPITALIZACION	MATERIAL MÉDICO QUIRURGICO	1	100.000	Editar
3	REQUERIMIENTOS	RX TORAX (2P)	2	1.200	Editar
4	REQUERIMIENTOS	ELECTROCARDIOGRAMA	2	300,9	Editar
5	SERVICIOS CLINICOS	CARDIOLOGO	1	190.000	Editar
6	SERVICIOS CLINICOS	MEDICO PERFUSIONISTA	1	200.001	Editar
7	SERVICIOS CLINICOS	ANESTESIOLOGO	1	200.000	Editar
8	SERVICIOS CLINICOS	LABORATORIO CLINICO	2	10.000	Editar
9	SERVICIOS CLINICOS	BANCO DE SANGRE RAZETTI	1	20.800	Editar
10	SERVICIOS CLINICOS	UNIDAD QUIRURGICA VASCULAR	1	17.500	Editar
11	SERVICIOS CLINICOS	SERVICIO INTEGRAL DE PABELLON	1	7.500	Editar
12	SERVICIOS CLINICOS	OXIDO NITROSO	6	1.500	Editar
13	SERVICIOS CLINICOS	INSTRUMENTAL QUIRURGICO	1	3.500	Editar

Figura 49. Edición de datos del presupuesto (administrador). Fuente: Captura de pantalla

- **Interfaz correspondiente a la carga de datos en el sistema a través de archivo .txt:** En la figura 50 se presenta la interfaz a través de la cual el administrador podrá cargar datos en el sistema mediante archivos .txt. El usuario debe seleccionar el tipo de entidad a cargar y luego el archivo correspondiente.

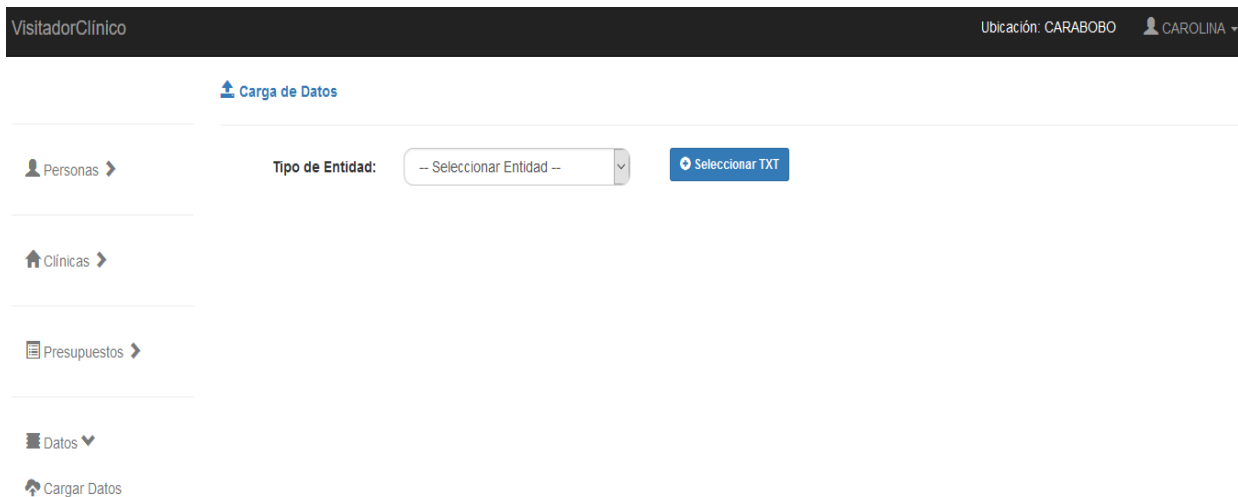


Figura 50. Carga de datos a través de .txt. Fuente: Captura de pantalla

4.9 Fase de pruebas

Las pruebas realizadas sobre el sistema se encuentran clasificadas en dos: Pruebas Funcionales y Pruebas de Aceptación de Usuario. Las Pruebas Funcionales se realizaron para verificar el cumplimiento de los requerimientos funcionales planteados para el sistema, mientras que las Pruebas de Aceptación de Usuario se realizaron para medir que tan fácil e intuitivo es el sistema para los usuarios finales, y así seguir mejorando el mismo en cuanto a aspectos relacionados a la Interacción Humano – Computador.

4.9.1 Pruebas funcionales

En estas pruebas se evaluó el cumplimiento de las funcionalidades requeridas para cada módulo y para ello se elaboró un instrumento de evaluación, presentado en la tabla 2, que consiste en una tabla donde se listan las funcionalidades requeridas o que se quieren evaluar de cada módulo junto con el estatus de aprobación (para cada

funcionalidad) por parte del usuario final del sistema. Las pruebas funcionales fueron realizadas de forma cíclica por cada módulo hasta que finalmente se cumplieran las funcionalidades requeridas para el mismo. (**N** es la cantidad de ciclos e **I** la cantidad de funcionalidades de cada módulo).

Nombre del Módulo	Ciclo	Funcionalidad	Estatus de Aprobación
Módulo X	N	Funcionalidad 1	Estatus 1
		Funcionalidad 2	Estatus 2
		Funcionalidad i	Estatus i

Tabla 2. Instrumento de evaluación para las pruebas funcionales. Fuente: Elaboración propia

En la tabla 3 se presenta el ciclo de pruebas llevado a cabo para el módulo de Carta Aval:

Nombre del Módulo	Ciclo	Funcionalidad	Estatus de Aprobación
Carta Aval	1	Buscar carta aval	Aprobado
		Visualizar datos de carta aval	Aprobado
		Descargar carta aval en PDF	Aprobado
		Descargar presupuesto en PDF	Aprobado
		Solicitar visita clínica para una carta aval	Aprobado
		Ver historial de visitas clínicas para una carta aval	Aprobado
		Cancelar visita clínica para una carta aval	Aprobado

Tabla 3. Primer ciclo de pruebas funcionales para el módulo de Carta Aval. Fuente: Desarrollo de pruebas con usuarios finales

El total de las funcionalidades comprendidas por este módulo fue aprobado en el primer ciclo de pruebas con los usuarios finales, por lo que no hubo necesidad de hacer otro ciclo de pruebas con este módulo y queda así culminado el desarrollo del mismo.

En la tabla 4 se presenta el primer ciclo de pruebas llevado a cabo para el módulo de Solicitud de Visita Clínica:

Nombre del Módulo	Ciclo	Funcionalidad	Estatus de Aprobación
Solicitud de Visita Clínica	1	Buscar visita clínica	Aprobado
		Visualizar datos de visita clínica	Aprobado
		Descargar presupuesto en PDF	Aprobado
		Descargar carta aval en PDF	Aprobado
		Descargar encuesta en PDF	Aprobado
		Ver historial de observaciones de una visita clínica	Aprobado
		Asignar solicitud de visita clínica	Aprobado
		Cambiar asignación de visita clínica	No Aprobado
		Modificar presupuesto original	Aprobado
		Ver historial de cambios en el presupuesto	No Aprobado
		Cargar respuestas de la encuesta	Aprobado
		Ver historial de respuestas cargadas de la encuesta	No Aprobado
		Cargar documentos digitales que avalan las respuestas cargadas y cambios en el presupuesto	Aprobado
		Visualizar los documentos digitales cargados	No Aprobado
		Finalizar visita clínica	Aprobado

Tabla 4. Primer ciclo de pruebas funcionales sobre el módulo de Solicitud de Visita Clínica. Fuente: Desarrollo de pruebas con usuarios finales

En el primer ciclo de pruebas para el módulo de Solicitud de Visita Clínica los usuarios aprobaron la mayoría de las funcionalidades correspondientes al módulo. Sin embargo, faltaron por aprobar cuatro funcionalidades. Se integraron cambios y correcciones sobre las funcionalidades no aprobadas y se procedió a realizar un segundo ciclo de pruebas, el cual se presenta en la tabla 5:

Nombre del Módulo	Ciclo	Funcionalidad	Estatus de Aprobación
Solicitud de Visita Clínica	2	Cambiar asignación de visita clínica	Aprobado
		Ver historial de cambios en el presupuesto	Aprobado
		Ver historial de respuestas cargadas de la encuesta	Aprobado
		Visualizar los documentos digitales cargados	Aprobado

Tabla 5. Segundo ciclo de pruebas funcionales sobre el módulo de Solicitud de Visita Clínica. Fuente: Desarrollo de pruebas con usuarios finales

En el segundo ciclo de pruebas para el módulo de Solicitud de Visita Clínica, los usuarios aprobaron las funcionalidades que faltaban por aprobación, quedando así culminado el desarrollo de este módulo.

En la tabla 6 se presenta el primer ciclo de pruebas para el módulo administrativo Personas:

Nombre del Módulo	Ciclo	Funcionalidad	Estatus de Aprobación
Personas	1	Buscar personas	Aprobado
		Visualizar datos de personas	Aprobado
		Editar datos de personas	Aprobado
		Crear usuario en el sistema a una persona	Aprobado
		Habilitar usuario de una persona	Aprobado
		Deshabilitar usuario de una persona	Aprobado
		Cambiar rol de usuario de una persona	Aprobado

Tabla 6. Primer ciclo de pruebas funcionales sobre el módulo de Personas. Fuente: Desarrollo de pruebas con usuarios finales

El total de las funcionalidades comprendidas por este módulo fue aprobado en el primer ciclo de pruebas con los usuarios finales, por lo que no hubo necesidad de hacer otro ciclo de pruebas con este módulo y queda así culminado el desarrollo del mismo.

En la tabla 7 se presenta el primer ciclo de pruebas para el módulo administrativo Clínicas:

Nombre del Módulo	Ciclo	Funcionalidad	Estatus de Aprobación
Clínicas	1	Buscar clínicas	Aprobado
		Visualizar datos de clínicas	Aprobado
		Editar datos de clínicas	Aprobado

Tabla 7. Primer ciclo de pruebas funcionales sobre el módulo de Clínicas. Fuente: Desarrollo de pruebas con usuarios finales

El total de las funcionalidades comprendidas por este módulo fue aprobado en el primer ciclo de pruebas con los usuarios finales, por lo que no hubo necesidad de hacer otro ciclo de pruebas con este módulo y queda así culminado el desarrollo del mismo.

En la tabla 8 se presenta el primer ciclo de pruebas para el módulo administrativo de Presupuestos:

Nombre del Módulo	Ciclo	Funcionalidad	Estatus de Aprobación
Presupuestos	1	Buscar presupuestos	Aprobado
		Visualizar datos de presupuestos	Aprobado
		Editar datos de presupuestos	Aprobado

Tabla 8. Primer ciclo de pruebas funcionales sobre el módulo de Presupuestos. Fuente: Desarrollo de pruebas con usuarios finales

El total de las funcionalidades comprendidas por este módulo fue aprobado en el primer ciclo de pruebas con los usuarios finales, por lo que no hubo necesidad de hacer otro ciclo de pruebas con este módulo y queda así culminado el desarrollo del mismo.

En la tabla 9 se presenta el primer ciclo de pruebas para el módulo administrativo de carga de datos a través de archivos en formato “.txt”:

Nombre del Módulo	Ciclo	Funcionalidad	Estatus de Aprobación
Personas	1	Cargar datos de personas	No Aprobado
		Cargar datos de clínicas	No Aprobado
		Cargar datos de pólizas	No Aprobado
		Cargar datos de presupuestos	No Aprobado
		Cargar datos de carta aval	No Aprobado

Tabla 9. Primer ciclo de pruebas funcionales sobre el módulo de Datos. Fuente: Desarrollo de pruebas con usuarios finales

En el primer ciclo de pruebas sobre este módulo las funcionalidades no cumplían con los requerimientos planteados y además, se hallaron fallas en las funcionalidades. Se planificó un segundo ciclo de pruebas sobre el este módulo, el cual se presenta en la tabla 10, teniendo en cuenta las consideraciones del primer ciclo y añadiendo las modificaciones necesarias para cumplir con los requerimientos.

Nombre del Módulo	Ciclo	Funcionalidad	Estatus de Aprobación
Datos	2	Cargar datos de personas	Aprobado
		Cargar datos de clínicas	Aprobado
		Cargar datos de pólizas	Aprobado
		Cargar datos de presupuestos	Aprobado
		Cargar datos de carta aval	Aprobado

Tabla 10. Segundo ciclo de pruebas funcionales sobre el módulo de Datos. Fuente: Desarrollo de pruebas con usuarios finales

4.9.2 Pruebas de aceptación de usuario

Para verificar el cumplimiento de los requerimientos no funcionales especificados para el sistema y el nivel de aceptación de los usuarios finales en cuanto a las interfaces de usuario, se elaboró una encuesta de siete preguntas presentada en la tabla 11. Esta encuesta fue aplicada a un grupo de seis personas correspondientes al conjunto de usuarios finales para así analizar la retroalimentación obtenida por parte de estos, y detectar en que aspectos se pueden mejorar las interfaces con la intención de mejorar la experiencia del usuario.

Nº	Pregunta	Totalmente de acuerdo	De acuerdo	En desacuerdo	Totalmente en desacuerdo
1	¿Las interfaces de usuario tienen una adecuada organización y estructura?				
2	¿Se hace un contraste adecuado en las interfaces de usuario?				
3	¿El diseño de las interfaces es consistente?				
4	¿Considera que el sistema es de fácil uso?				
5	¿El sistema es de navegación intuitiva?				
6	¿El lenguaje usado en el sistema es claro y conciso?				
7	¿El sistema ofrece mensajes y notificaciones claras acerca de los eventos ocurridos?				

Tabla 11. Encuesta de Aceptación. Fuente: Elaboración propia

A continuación se presentan los resultados obtenidos para cada pregunta de la encuesta:

¿Las interfaces de usuario tienen una adecuada organización y estructura?

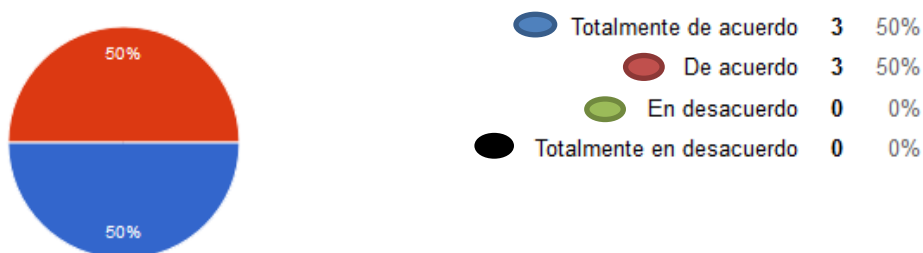


Figura 51. Resultados de la pregunta N° 1. Fuente: Pruebas de aceptación con usuarios finales

¿Se hace un contraste adecuado en las interfaces de usuario?

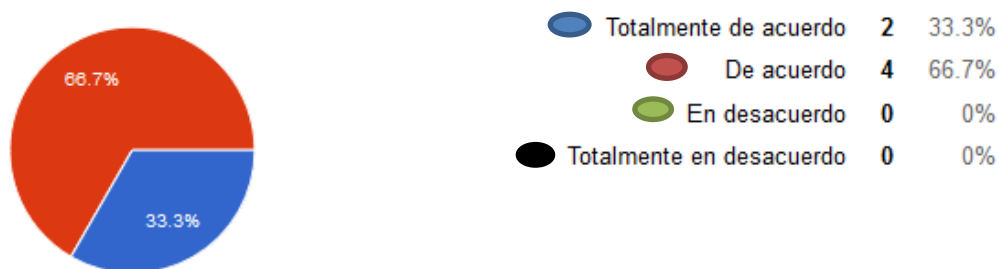


Figura 52. Resultados de la pregunta N° 2. Fuente: Pruebas de aceptación con usuarios finales

¿El diseño de las interfaces es consistente?

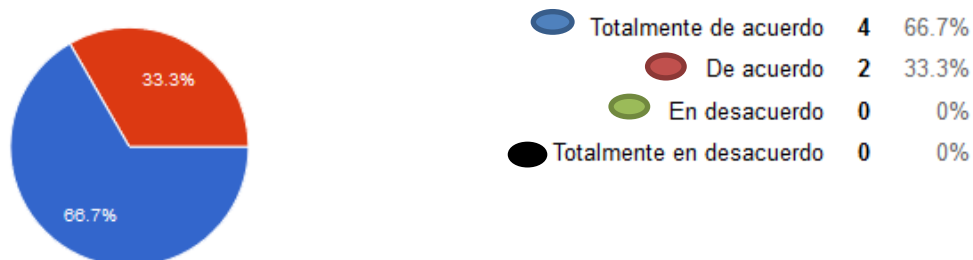


Figura 53. Resultados de la pregunta N° 3. Fuente: Pruebas de aceptación con usuarios finales

¿Considera que el sistema es de fácil uso?

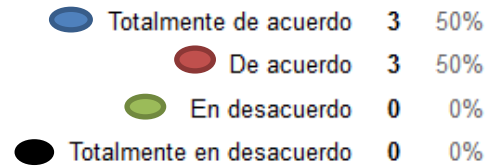
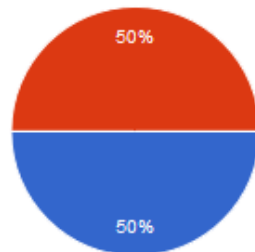


Figura 54. Resultados de la pregunta N° 4. Fuente: Pruebas de aceptación con usuarios finales

¿El sistema es de navegación intuitiva?

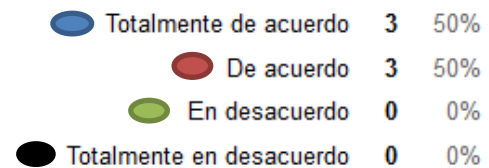
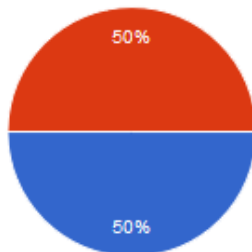


Figura 55. Resultados de la pregunta N° 5. Fuente: Pruebas de aceptación con usuarios finales

¿El lenguaje usado en el sistema es claro y conciso?

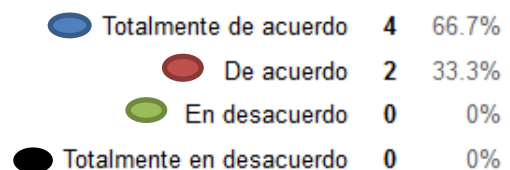
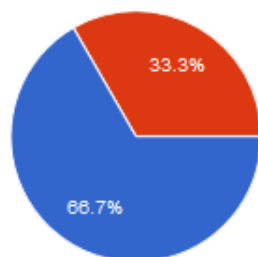


Figura 56. Resultados de la pregunta N° 6. Fuente: Pruebas de aceptación con usuarios finales

¿El sistema ofrece mensajes y notificaciones claras acerca de los eventos ocurridos?



Figura 57. Resultados de la pregunta N° 7. Fuente: Pruebas de aceptación con usuarios finales

Según los resultados obtenidos a través de la encuesta realizada, se puede concluir que los usuarios perciben el sistema como fácil de usar, de navegación intuitiva, interfaces con un contraste adecuado, interfaces organizadas y consistentes, además de utilizar un lenguaje claro, conciso e informar claramente acerca de los eventos que están ocurriendo en el sistema. Es importante destacar en estas pruebas participaron tres (3) usuarios finales especializados en el proceso de visita clínica (coordinadores en el proceso), por lo que se consideró que un total de seis (6) participantes era suficiente para realizar las mismas y detectar cualquier inconveniente con sistema desarrollado. Además, a través de la realización de estas pruebas los usuarios pudieron manifestar sus opiniones sobre otros aspectos del sistema, como son: el tamaño de letra, color de la letra en algunos títulos y opinar acerca del posicionamiento de elementos de las interfaz. Estas opiniones permitieron ajustar más las interfaces de usuarios de acuerdo a las necesidades de los usuarios finales.

Conclusiones y Recomendaciones

A través del presente Trabajo Especial de Grado se desarrolló un Sistema de Información que cumplió con los objetivos y requerimientos planteados para agilizar el proceso de Visitas Clínicas de una compañía aseguradora. Para el cumplimiento de los objetivos y requerimientos fue crucial el análisis y entendimiento previo del proceso de negocio que se quería agilizar, detectando así las debilidades y fortalezas que poseía inicialmente para luego plantear una solución que optimizase la ejecución del mismo. Una vez planteada la solución fue importante evaluar y analizar el conjunto de tecnologías que podían utilizarse para la implementación de la misma con la intención de decidir acertadamente las tecnologías de desarrollo de acuerdo a los requerimientos y especificaciones definidas. Para el desarrollo de este Sistema de Información se utilizaron tecnologías como HTML, CSS, JavaScript, Angular.JS, Node.JS, Bootstrap y PostgreSQL, ya que el uso de estas se adaptaban a los requerimientos y especificaciones definidas para la solución desarrollada. Fue igualmente importante guiar el desarrollo de la solución mediante la aplicación de una Metodología de Desarrollo que se adaptara a las necesidades de la solución y que permitiera ir cumpliendo adecuadamente los objetivos de la misma. La solución desarrollada fue guiada mediante una adaptación de la Metodología de Desarrollo SCRUM, en la cual el Propietario del Producto estuvo representado por el Prof. Franky Uzcátegui y el Equipo SCRUM estuvo representado por el Br. Argenis Chirinos. Fue de vital importancia la participación de los usuarios durante el desarrollo de la solución, la verificación de los requerimientos y pruebas funcionales, ya que ofrecieron una retroalimentación que permitió entender mejor sus necesidades y así desarrollar una solución que cumpliera con sus expectativas.

Se debe tener en cuenta que es de suma importancia para todo sistema adaptarse a nuevas necesidades e incluir con rapidez funcionalidades que permitan

Conclusiones y Recomendaciones

satisfacer las mismas, por esta razón se debe de tener en cuenta las siguientes recomendaciones:

- El despliegue del sistema web no se realizó en un servidor dedicado, sin embargo, se sugiere hacer el despliegue en un servidor cuyos recursos estén dedicados al funcionamiento del sistema.
- Desarrollar una aplicación móvil a través de la cual el visitador pueda cargar la información levantada durante la visita.
- Incorporar una aplicación de Inteligencia de Negocio que permita generar indicadores del funcionamiento del proceso, contribuyendo así a la toma de decisiones en el nivel operacional de la compañía de seguros involucrado en el proceso de Visitas Clínicas.
- Mantener actualizadas las tecnologías utilizadas en el desarrollo del sistema para así aprovechar ampliamente las bondades que puedan ofrecer. Entre ellas: el manejador de base de datos, Node.JS y el framework Angular.JS.
- Considerar en todo momento la agregación de una posible extensión al sistema, como podría ser una aplicación móvil. Cualquier funcionalidad que se añada al sistema web debería de poder ser utilizada también por otras aplicaciones.
- Mantener los estándares del API de los servicios web. Cualquier cambio que se haga en los servicios web implementados en el sistema desarrollado no debería alterar la interfaz del servicio, de lo contrario también deberían ajustarse todos los clientes que consumen dichos servicios.

Bibliografía

- [1] Laudon, K y Laudon, J. (2012). Sistemas de Información Gerencial. Estados Unidos. PEARSON.
- [2] Luján, S. (2002). Programación de Aplicaciones Web. España. Editorial Club Universitario.
- [3] Ferrer, J. (2012). Implantación de Aplicaciones Web. RA-MA Editorial.
- [4] Wikipedia. (SF). HTML. Recopilado de: <https://es.wikipedia.org/wiki/HTML>.
- [5] Eguiluz, J. (SF). CSS. Recopilado de: <http://librosweb.es/libro/css>.
- [6] Eguiluz, J. (SF). JavaScript. Recopilado de: <http://librosweb.es/libro/javascript>.
- [7] Wikipedia. (SF). jQuery. Recopilado de: <https://es.wikipedia.org/wiki/JQuery>.
- [8] Eguiluz, J. (SF). Ajax. Recopilado de: <http://librosweb.es/libro/ajax>.
- [9] Wikipedia. (SF). AngularJS. Recopilado de: <https://es.wikipedia.org/wiki/AngularJS>.
- [10] Wikipedia. (SF). Twitter Bootstrap. Recopilado de: https://es.wikipedia.org/wiki/Twitter_Bootstrap.
- [11] Silberschatz, A. (2002). Fundamentos de Bases de Datos. Estados Unidos. McGraw - Hill.
- [12] PostgreSQL - es. (SF). Sobre PostgreSQL. Recopilado de: http://www.postgresql.org.es/sobre_postgresql.
- [13] Wikipedia. (SF). Framework para aplicaciones web. Recopilado de: https://es.wikipedia.org/wiki/Framework_para_aplicaciones_web.
- [14] Sattari, H y Kunjumohamed, S. (2016). Spring Essentials. Mumbai. Packt Publishing.

- [15] Wikipedia. (SF). Modelo - Vista - Controlador. Recopilado de: <https://es.wikipedia.org/wiki/Modelo%E2%80%93vista%E2%80%93controlador>.
- [16] Express. (SF). Expressjs. Recopilado de: <http://expressjs.com/es/>.
- [17] NPM. (SF). Express. Recopilado de: <https://www.npmjs.com/package/express>.
- [18] Wikipedia. (SF). Servicios web. Recopilado de: https://es.wikipedia.org/wiki/Servicio_web.
- [19] EcuRed. (SF). Servicios web. Recopilado de: http://www.ecured.cu/Servicios_Web.
- [20] JSON. (SF). Introducción a json. Recopilado de: <http://www.json.org/json-es.html>.
- [21] Hipertextual. (2014). Formatos para intercambiar información. Recopilado de: <https://hipertextual.com/archivo/2014/05/xml-json-yaml/>.
- [22] Wikipedia. (SF). Representational state transfer. Recopilado de: https://es.wikipedia.org/wiki/Transferencia_de_Estado_Representacional.
- [23] Wasson, M. (2013). ASP.NET - Single-Page Applications: Build Modern, Responsive Web Apps with ASP.NET. Recopilado de: <https://msdn.microsoft.com/en-us/magazine/dn463786.aspx>.
- [24] Wikipedia. (SF). Metodología de desarrollo de software. Recopilado de: https://es.wikipedia.org/wiki/Metodolog%C3%ADa_de_desarrollo_de_software.
- [25] Valdéz, J. (2014). Implementación del modelo integral colaborativo (mdsic) como fuente de innovación para desarrollo ágil. 2014. Recopilado de: <http://www.eumed.net/tesis-doctorales/2014/jlcv/#indice>.
- [26] EcuRed. (2012). Metodologías tradicionales. Recopilado de: https://www.ecured.cu/Metodolog%C3%ADas_Tradicionales.
- [27] Sommerville, I. (2005). Ingeniería del Software. Estados Unidos. PEARSON.

[28] Satpathy, T. (2016). Una guía para el conocimiento de SCRUM. Estados Unidos. SCRUMstudy.

[29] Pérez, O. (2011). Cuatro enfoques metodológicos para el desarrollo de software.
Recopilado de:
<http://biblioteca.uniminuto.edu/ojs/index.php/Inventum/article/view/9/9>.