

DESARROLLO DE UN APLICATIVO WEB PARA LA FIDELIZACIÓN DE LOS
ASPIRANTES DE LA CORPORACIÓN UNIVERSITARIA DEL HUILA – CORHUILA

PRESENTADO POR:
HENRY ANDRÉS PERDOMO ROA
MARTHA MILENA VANEGAS CAPERA

CORPORACIÓN UNIVERSITARIA DEL HUILA - CORHUILA.
FACULTAD DE INGENIERÍA.
PROGRAMA DE INGENIERÍA DE SISTEMAS
NEIVA – HUILA
2018

DESARROLLO DE UN APLICATIVO WEB PARA LA FIDELIZACIÓN DE LOS
ASPIRANTES DE LA CORPORACIÓN UNIVERSITARIA DEL HUILA - CORHUILA

PRESENTADO POR:
HENRY ANDRÉS PERDOMO ROA
MARTHA MILENA VANEGAS CAPERA.

Trabajo de Grado para obtener el Título de Ingeniero de Sistemas

Director
Mg. José Miguel Llanos Mosquera

CORPORACIÓN UNIVERSITARIA DEL HUILA - CORHUILA.
FACULTAD DE INGENIERÍA.
PROGRAMA DE INGENIERÍA DE SISTEMAS
NEIVA – HUILA
2018

Nota de aceptación

Presidente del Jurado

Jurado

Jurado

Dedicatoria

Andrés.

Este proyecto lo dedico a Dios por otorgarme la vida y dotarme de capacidades y habilidades para alcanzar objetivos. A todas aquellas personas que, con sus buenas energías y apoyo en este proceso, me inspiraron para poder culminar dicha labor. Especialmente a mi madre, abuelos y toda mi familia en general por su amor incondicional. A todos mis amigos y docentes por el acompañamiento y formación profesional.

Milena.

Dedico este proyecto primero que nada a Dios, por permitirme llegar a este punto con salud para lograr mis objetivos.

A mis padres, por haberme apoyado en cada paso de mi carrera, y cada uno de sus esfuerzos, por hacer de mí una mejor persona cada día, por el amor y apoyo brindado en cada momento difícil.

A mi hermano, por estar conmigo y apoyarme siempre ...Se le quiere un montón.

A mis mascotas, Nerón y Zeus que en los momentos difíciles me sacaron una sonrisa.

Agradecimientos

Andrés.

Quiero agradecer a Dios por el privilegio de permanecer con vida y permitirme lograr grandes cosas.

A mi familia en general por todo su apoyo y amor brindado.

A mi compañera Martha Milena Vanegas por ser un complemento para este proyecto y ser parte importante del mismo. De antemano al Ingeniero y magister José Miguel Llanos por su tiempo, acompañamiento y asesoramiento como director del proyecto.

Finalmente, a la Corporación Universitaria del Huila por los 5 años de formación y todas las experiencias y oportunidades ofrecidas en el transcurso de la carrera.

Milena.

A Dios, por permitirme llegar a este momento. A mis padres por ser las personas que me ha acompañado durante todo mi trayecto estudiantil y de vida, a mi compañero Henry Andrés Perdomo Roa, que gracias al equipo que formamos logramos llegar hasta el final del camino. A los profesores, gracias por su tiempo, apoyo, y la sabiduría que me transmitieron en el desarrollo de mi formación profesional, al programa de Ingeniería de sistemas a cargo de la directora de programa Edisney Gracia y su secretaria Blanquita por su colaboración en cada etapa.

A mi director de proyecto el Ingeniero José Miguel Llanos Mosquera por el tiempo dedicado y darnos la confianza de manejar un proyecto como UHCRM.

Tabla de Contenido

1.	Introducción	1
2.	Justificación	2
3.	Objetivo General	3
3.1.	Objetivos Específicos.....	3
4.	Planteamiento del Problema	4
5.	Marco Teórico.....	6
5.1.	CRM (Customer Relationship Management)	8
5.1.1.	En la Educación	8
5.1.2.	Implantación en las Universidades	10
5.2.	Marketing Educativo	11
5.2.1.	Herramientas del Marketing Educativo	12
5.3.	Manejo de la Imagen Institucional.....	14
5.3.1.	El Aspirante Como Objeto Principal	15
5.4	Presencia de las Instituciones en la Red	16
6.	Marco Conceptual	18
7.	Metodología de Investigación.....	22
7.1.	Investigación Descriptiva	22
7.1.1.	Etapas del Método Descriptivo	22
7.1.2.	Características de la Investigación Descriptiva	23
7.2.	Nivel de la Investigación	23
7.3.	Tipo de Investigación	24
7.4.	Diseño de la Investigación	24
7.4.1.	Diseño Transversal Descriptivo.....	24
7.5.	Población y Muestra	24
7.6.	Recolección de Datos	26
7.6.1.	Datos Primarios.....	26
7.6.1.1.	Observación No Participante	26
7.6.1.2	Entrevista No Estructurada	26
7.6.2.	Datos Secundarios.....	26
8.	Metodología de Desarrollo	27

8.1. Scrum	27
8.1.1. Roles Principales.....	28
8.1.1.1. Propietario del Producto o (Product Owner)	28
8.1.1.2 Scrum Máster	28
8.1.1.3Equipo de trabajo o (Scrum Team).....	28
8.1.2. Elementos del Scrum	29
8.1.2.1. Lista de Producto (Product Backlog):.....	29
8.1.2.2. Lista de pendientes (Sprint Backlog).....	30
8.1.3. Incremento (Sprint).....	31
8.2. Fase de Análisis	32
8.2.1. Requerimientos Funcionales del Aplicativo Web.....	32
8.2.2. Requerimientos No Funcionales del Aplicativo Web.....	36
8.3. Fase de Diseño	41
8.3.1. Modelo 4 + 1 de Philippe Kruchten.....	41
8.3.1.1. Escenarios	42
8.3.1.2. Vista Lógica o (Logical View)	52
8.3.1.3. Vista del Desarrollo o (Development View)	57
8.3.1.4. Vista Física o (Physical View).....	58
8.3.1.5. Vista de procesos o (Process View).....	59
8.3.2. Modelo Relacional de la Base de Datos	62
8.3.3. Modelo Vista Controlador (MVC).....	64
8.4. Fase de Desarrollo.....	67
8.4.1. Estándares	67
8.4.2. Código Limpio	69
8.4.3. Programación Orientada a Objetos (POO)	80
8.5. Fase de Prueba o Testing	86
8.5.1. Test de Usabilidad.....	86
8.5.2. Pruebas Unitarias	89
8.6. Fase de Implementación	97
8.6.1. Servidor Web	97
8.6.2. Servidor de Aplicaciones GlassFish	97
8.6.3. Motor de Base de Datos PostgreSQL	98

9. Cronograma de Actividades.....	99
10. Presupuesto	100
11. Conclusiones	101
Referencias.....	102

Lista de Tablas

Tabla 1 Población Involucrada Directamente.....	25
Tabla 2 Requerimiento Funcional RF01.....	32
Tabla 3 Requerimiento Funcional RF02.....	32
Tabla 4 Requerimiento Funcional RF03.....	33
Tabla 5 Requerimiento Funcional RF04.....	33
Tabla 6 Requerimiento Funcional RF05.....	34
Tabla 7 Requerimiento Funcional RF06.....	34
Tabla 8 Requerimiento Funcional RF07.....	35
Tabla 9 Requerimiento Funcional RF08.....	35
Tabla 10 Requerimiento No Funcional RNF01.....	36
Tabla 11 Requerimiento No Funcional RNF02.....	36
Tabla 12 Requerimiento No Funcional RNF03.....	37
Tabla 13 Requerimiento No Funcional RNF04.....	37
Tabla 14 Requerimiento No Funcional RNF05.....	38
Tabla 15 Requerimiento No Funcional RNF06.....	38
Tabla 16 Requerimiento No Funcional RNF07.....	39
Tabla 17 Requerimiento No Funcional RNF08.....	39
Tabla 18 Requerimiento No Funcional RNF09.....	40
Tabla 19 Caso de Uso C01.....	43
Tabla 20 Caso de Uso C02.....	44
Tabla 21 Caso de Uso C03.....	45
Tabla 22 Caso de Uso C04.....	46
Tabla 23 Caso de Uso C05.....	47
Tabla 24 Caso de Uso C06.....	48
Tabla 25 Caso de Uso C07.....	49
Tabla 26 Caso de Uso C08.....	51
Tabla 27 Criterios de Evaluación.....	87
Tabla 28 Test de Usabilidad	88
Tabla 29 Caso de Prueba CP01.....	90
Tabla 30 Caso de Prueba CP02.....	91
Tabla 31 Caso de Prueba CP03.....	94
Tabla 32 Caso de Prueba CP04.....	95
Tabla 33 Caso de Prueba CP05.....	96
Tabla 34 Presupuesto Global por Fuentes de Financiación.....	100
Tabla 35 Descripción del Personal Investigador	100
Tabla 36 Materiales e Insumos	100

Lista de Ilustraciones

Ilustración 1 Metodología Scrum.....	27
Ilustración 2 Lista de Producto (Product Backlog).....	29
Ilustración 3 Lista de Pendientes (Sprint Backlog).....	30
Ilustración 4 Planificación de un Sprint.....	31
Ilustración 5 Modelo 4 + 1 de Philippe Kruchten.....	41
Ilustración 6 Diagrama de Casos de Uso General	42
Ilustración 7 Caso de Uso C01.....	43
Ilustración 8 Caso de Uso C02.....	44
Ilustración 9 Caso de Uso C03.....	45
Ilustración 10 Caso de Uso C04.....	46
Ilustración 11 Caso de Uso C05.....	47
Ilustración 12 Caso de Uso C06.....	48
Ilustración 13 Caso de Uso C06.2.....	49
Ilustración 14 Caso de Uso C07.....	50
Ilustración 15 Caso de Uso C08.....	51
Ilustración 16 Diagrama de Clases	53
Ilustración 17 Diagrama de Secuencia DS01.....	54
Ilustración 18 Diagrama de Secuencia DS02.....	55
Ilustración 19 Diagrama de Comunicación DC01	56
Ilustración 20 Diagrama de Comunicación DC02	56
Ilustración 21 Diagrama de Paquetes.....	57
Ilustración 22 Diagrama de Componentes.....	58
Ilustración 23 Diagrama de Despliegue.....	58
Ilustración 24 Diagrama de Actividades DA01	59
Ilustración 25 Diagrama de Actividades DA02	60
Ilustración 26 Diagrama de Actividades DA03.....	61
Ilustración 27 Modelo Entidad Relación	63
Ilustración 28 Comunicación del Patrón (MVC).....	65
Ilustración 29 Patrón (MVC) Aplicación - Modelo.....	65
Ilustración 30 Patrón (MVC) Aplicación - Vista.....	66
Ilustración 31 Patrón (MVC) Aplicación - Controlador.....	66
Ilustración 32 Estándar Empleado en la Base de Datos.....	67
Ilustración 33 Estándar Empleado en la Codificación.....	68
Ilustración 34 Estándar Aplicado a Clases y Variables	69
Ilustración 35 Código Limpio - Aplicación Variables Globales	70
Ilustración 36 Código Limpio – Aplicación a los Métodos GET y SET	70
Ilustración 37 Código Limpio – Aplicación Método Llamada.....	71
Ilustración 38 Vista Registro Aspirante Parte 1.....	72
Ilustración 39 Vista Registro Aspirante Parte 2.....	73
Ilustración 40 Vista Registro Aspirante Parte 3.....	74
Ilustración 41 Vista Registro Aspirante Parte 4.....	75

Ilustración 42 Vista Registro Aspirante Parte 5.....	76
Ilustración 43 Controlador Registro Aspirante Parte 1.....	77
Ilustración 44 Controlador Registro Aspirante Parte 2.....	78
Ilustración 45 Modelo Registro Aspirante.....	79
Ilustración 46 Aplicación de POO Parte 1.....	81
Ilustración 47 Aplicación de POO Parte 2.....	82
Ilustración 48 Aplicación de POO parte 3.....	83
Ilustración 49 Aplicación de POO Parte 4.....	84
Ilustración 50 Aplicación de POO Parte 5.....	84
Ilustración 51 Aplicación de POO Parte 6.....	85
Ilustración 52 Inicio de Sesión y Acceso.....	90
Ilustración 53 Formulario Registro.....	93
Ilustración 54 Vista Registro.....	94
Ilustración 55 Vista Administrador - Rol y permisos.....	95
Ilustración 56 Vista Administrador- Eliminar Usuario.....	96
Ilustración 57 Cronograma de Actividades.....	99

1. Introducción

La Corporación Universitaria del Huila – Corhuila, es consciente del proceso de captación de estudiantes y pretende implementar herramientas tecnológicas para optimizar los procesos de atención personalizada que realiza el área de admisiones; capturando los datos importantes de los aspirantes desde su primer contacto de una manera más rápida y eficiente, a través de la cual se obtendrán estrategias de marketing ayudando a dar información específica sobre los intereses del aspirante según su información.

La optimización de procesos pretende lograr la reducción de tiempo y esfuerzo empleado para desarrollar cada una de las actividades, con el fin de generar una relación estable entre el aspirante y la universidad, así mismo alcanzar la fidelización de sus usuarios y mejorar la calidad de sus servicios enfocados a la educación superior.

Este amplio sector educativo, ha soportado grandes cambios sustanciales en lo que concierne al contexto competitivo del mercado al tener como objetivo la mayor captación de aspirantes a su institución. En la actualidad, el concepto de fidelización es nuevo en el sector educativo, el cual pretende la optimización de las relaciones con los aspirantes y con esto utilizar la información derivada de este primer contacto en la producción de diferentes estrategias de captación de aspirantes con la información anteriormente recolectada (Naranjo Hernández, 2011). Al usar conceptos como estos, se pretende evitar que los aspirantes abandonen su intención de pertenecer a una academia universitaria, brindando la ayuda pertinente durante el proceso de inscripción, y sus subprocesos para la consolidación de su matrícula y su proceso educativo dentro de la institución.

2. Justificación

La Corporación Universitaria del Huila – Corhuila, cuenta con un área de admisiones encargada de realizar todo el proceso de inscripción para el ingreso a los diferentes programas académicos, por eso se busca mecanismos que ayuden a la captación de nuevos aspirantes, pues es de vital importancia aplicar estrategias porque permiten el seguimiento en las etapas de la matrícula, dado esto se planteó incluir el desarrollo de un aplicativo web de fidelización de aspirantes, lo cual permitirá recabar la información necesaria para su tratamiento y permitir un primer contacto estable facultando la toma de decisiones oportunamente.

La pérdida de estos aspirantes se debe a la poca información suministrada, puesto que se está almacenando a través de un formulario y posteriormente exportado a un documento Excel, con una vista muy poco agradable que dificultan el proceso de monitoreo especializado al aspirante. Por lo tanto, se planteó el desarrollar un aplicativo web el cual genere informes detallados sobre el proceso realizado con un determinado aspirante, identificando cada una de las fases de su proceso. Esto permitirá la adecuada atención al aspirante convirtiendo ese primer contacto en un vínculo estable entre el aspirante y la Corporación Universitaria del Huila – Corhuila.

El beneficio generado con el uso de esta aplicación web es obtener un incremento de inscripciones en los periodos semestrales, significando un valor monetario para la institución.

3. Objetivo General

Desarrollar un aplicativo web para la fidelización de los aspirantes de la Corporación Universitaria del Huila – Corhuila.

3. 1. Objetivos Específicos

- † Analizar la información determinando las necesidades del área de admisiones, con el fin de definir los requisitos funcionales y no funcionales de la aplicación web.
- † Construir una Base de Datos que facilite la integración de la información proporcionada para su debido procesamiento por parte del área de admisiones.
- † Validar la aplicación web elaborando pruebas funcionales, en conjunto con la oficina de admisiones con el fin de verificar los requisitos definidos.

4. Planteamiento del Problema

La fidelización de aspirantes implica más que brindar información a un usuario para seguir una secuencia de pasos, correspondientes a un proceso interno cuyo objetivo final es quedar matriculado en la institución, identificando el nivel de interés por parte del aspirante, permitiendo realizar el respectivo seguimiento para así evitar la intención de abandono esto consisten en realizar llamadas telefónicas y envío de correos electrónicos con información pertinente al proceso de matrícula.

Al contar con más de 3.000 aspirantes promedio por cada periodo de inscripción semestral con base en los datos obtenidos por el área de admisiones, se generan altos volúmenes de información que se almacenan en la oficina de admisiones en documentos Excel, correspondiendo a los datos de cada uno de los aspirantes y la etapa en el cual se encuentra entre la trazabilidad del proceso, debido a esto se presenta una falencia en el momento de acceder a la misma, para realizar el monitoreo pertinente, debido a la organización de los datos no es la más adecuada para generar una vista dinámica y así poder establecer una metodología eficaz para mejorar la administración del tiempo empleado en cada uno de los procesos y lograr una atención personalizada de manera oportuna.

La dificultad se presenta porque los datos son almacenados en documentos de Excel que proporcionan una vista compleja de los registros de cada aspirante, a los cuales se les realiza el respectivo seguimiento; esto genera inconvenientes para la toma de decisiones, generación de reportes, y el desconocimiento de los perfiles o características de los aspirantes.

Con base en la problemática descrita anteriormente se plantea la siguiente pregunta de investigación:

¿Cómo se podrían mejorar los índices de fidelización de aspirantes de la Corporación Universitaria del Huila – Corhuila, implementando un aplicativo web?

5. Marco Teórico

El servicio ofrecido por una organización abarca diferentes aspectos como la calidad de servicio al cliente, publicidad, presentación y las características del servicio, la usabilidad y la confiabilidad otorgada al mismo, esto define la experiencia de forma directa e indirectamente (Andajani, 2015).

Al enfocarse en el sector educativo ocurre lo mismo, se puede observar la interacción con el usuario de una manera satisfactoria, así mismo, universidades como la Nacional sede Manizales tiene en cuenta y estudia los conceptos de la gestión de relaciones y la administración de la experiencia, para posteriormente utilizarlo en su beneficio al realizar estrategias de marketing que involucren estas ideologías, permitiendo ganar más consumidores y mantenerlos, al ofrecer un buen servicio cumpliendo con las expectativas establecidas, pues su exigencia es más exigentes con los servicios recibidos. (Naranjo Hernández, 2011).

Tomando como referencia que las organizaciones están implementando C.R.M (Customer Relationship Management), a un nivel estándar, se ha de entender que las estas estrategias cuentan con unas determinadas fases o alcances lo cual permiten definir el punto de corte al cual llegará el proyecto. Inicialmente se debe hacer una ejecución de las acciones de captación de nuevos clientes, el cual tiene como principal encargado al área de marketing de la institución donde se ejecutará el proyecto, la Universidad Católica San Antonio establece como objetivo la información suministrada, su orientación, y la ayuda en la captación respecto al plan de estrategia de marketing que maneja, ya sea con la promoción de la oferta de estudios de la Universidad realizada a través de los diferentes medios con los que cuenta: en su página Web, medios de comunicación y las

visitas programadas a instituciones o centros de educación secundaria (Universidad Católica San Antonio, 2012).

Sin embargo se deben simplificar los procesos de captación de usuarios de tal forma que no sea tarea difícil para la organización, es por eso que existen casos de estudio en donde la implementación de software especializado como Ellucian, una plataforma flexible que permite ayudas pertinentes en el proceso de gestión de relación con el cliente, especialmente para las instituciones educación superior (Ellucian, 2017), genera resultados realmente satisfactorios, en donde se refleja el beneficio al implementar esta herramienta tecnológica. La Universidad Americana de Kuwait reconoce que para realizar su visión global de mejorar el éxito estudiantil necesita abordar tres cuestiones principales. En primer lugar, necesitaba mejorar el asesoramiento de los estudiantes con tiempo para la graduación como un indicador clave de rendimiento.

Segundo, es necesario automatizar procesos manuales engorrosos. Finalmente es necesaria una eficaz toma de decisiones basadas en información en tiempo real, debido a estos cambios realizados a través de Ellucian, la universidad está experimentando una rápida expansión de la matrícula (Study Case: Provide students with better services and Support, 2016), como la Universidad Bautista de Houston (HBU) que durante los últimos cuatro años la inscripción en la universidad ha aumentado 29.9 por ciento de 2,432 estudiantes en otoño de 2011 a 3.160 estudiantes en el otoño de 2015 (Ellucian, 2016), con esto se espera reflejar que los casos implantados en las universidades tanto internacionales como nacionales con los procesos implementados y que son casos de éxito, enfocarlos de una manera regional en la Corporación Universitaria del Huila – Corhuila, brindando la ayuda que se necesita y que se requiere para mejorar los procesos anteriormente descritos en el área de admisiones y comunicaciones.

5.1. CRM (Customer Relationship Management)

Hay numerosas definiciones del CRM, claro que esta depende de las soluciones ofrecidas por las diferentes empresas de desarrollo, no todas las soluciones empleadas obtendrán un CRM completo y estable, pues las organizaciones solo necesitan ciertos aspectos coherentes centrados en sus usuarios, canales de comunicación y la organización de la empresa (Rajola, 2013).

Aunque el CRM, debería integrarse en todos los procesos de una empresa, algunas no lo hacen, pero otras empresas lo emplean incluso con sus proveedores. Cuando una empresa afirma que su objetivo es un excelente servicio de atención al cliente, debe significar la totalidad de la empresa. Este enfoque es obligatorio (Wali & Wright, 2016), enfocando en este punto un CRM requiere de la identificación, es decir una bitácora operacional del usuario donde se lleve los registros, productos, actividad diaria etc. Con información se podría hacer análisis de las actividades, reglas del comportamiento y la busca de nuevos clientes. Una vez integrado se gestiona el ciclo de vida de cada cliente respecto a la interacción con la empresa, lo cual marca la pauta de coordinar las acciones pertinentes en la busca de nuevos clientes o la fidelización de los mismos (Rajola, 2013).

5.1.1. En la Educación

La educación superior está experimentando cambios en la manera en que operan e interactúan con los estudiantes, padres, empresarios y funcionarios, reclaman atención y servicio instantáneo de manera más activa. Al observar esta situación las instituciones están recurriendo a la tecnología para la gestión de relaciones con el estudiante para satisfacer esta demanda de una manera eficaz (Neghina, 2016).

Con el fin de lograr resultados óptimos en la educación superior se están adoptado estrategias de CRM, con el fin de promover mejores prácticas y priorizar la relación con el estudiante y potenciales, es aquí donde la fidelización cobra un papel principal en la institución (Córdoba Filio & Julca Segovia, 2005), adoptando estrategias como esta, se busca el desarrollo del estudiante como cliente, cuyo valor principal es el aumento de estudiantes .(Rigo, Pedron, Caldeira, & Araújo, 2016).

La principal preocupación de las instituciones, es mantener la relación con sus estudiantes, en este caso se habla de que el mantenimiento de la relación es más bajo que la captación de un nuevo cliente, una de las claves es escuchar las diferentes sugerencias que estos puedan tener para ejecutarlas de una forma eficaz.(Quichimbo Sangurima & Vallejo Bojorque, 2012), con la implantación de un CRM educativo se busca el desarrollo utilizando herramientas libres, promoviendo la innovación y desarrollo tecnológico a través de las mismas(Córdoba Filio & Julca Segovia, 2005), para la gestión y automatización de los procesos que permita la comunicación con los actuales y los futuros aspirantes con la información básica como nombre del estudiante, género, antecedentes educativos, teléfono, correo electrónico u otro tipo de información la cual permita personalizar los mensajes y los canales utilizados para entregar el mensaje al estudiante en forma oportuna (Neghina, 2016).

Al utilizar tecnologías en las instituciones se da paso a una comunicación en ambos sentidos porque se estaría implantando modelos de relación con otros agentes externos vinculados con la institución, la evolución se ve reflejada en la relación con los estudiantes (Petrella, 2008).

5.1.2. Implantación en las Universidades

Durante los últimos años, la educación superior se ha sometido a un cambio constante, y es necesario que comprendan la utilidad de la implantación de nuevas herramientas tecnológicas ayudando a esta evolución (Petrella, 2008), debido a este cambio las ofertas ofrecidas por las diferentes instituciones, están obligando a las mismas a adoptar estrategias impulsando la captación de nuevos estudiantes y la retención de los existentes, en busca de la satisfacción de los mismos, buscando seguir el ciclo de vida del estudiante durante su carrera es decir, una evolución académica (Duque, 2017).

La necesidad de adaptarse a los diferentes entornos para incrementar el valor de sus servicios permite el diseño de ofertas que se ajusten a las diferentes situaciones en la cual se encuentra el estudiante, la oferta de las instituciones, deben contener elementos atractivos haciéndolas únicas y permitan la diferencia ante la competencia para la captación. (Marzo Navarro, Pedraja Iglesias, & Rivera Torres, 2004).

Al implantar estrategias CRM, es indispensable seguir una metodología que dependerá de los procesos en los cuales las organizaciones en efecto necesiten un aporte significativo que los beneficie (Pineda Rodríguez & Jaime Bohada, 2017), algunas posibilidades utilizadas es el envío masivo de correo electrónico el cual puede ser personalizado, también puede hacerse uso de mensajes de texto para socializar las diferentes actividades realizadas, existe una variedad de sistemas usados para la fidelización, de las cuales se pueden usar licencias online, o en su defecto desarrollado de acuerdo a las necesidades específicas de la institución es decir, a medida (SofDoit So easyto compare, 2018), de las cuales pueden alinearse a los objetivos tecnológicos del área de

TIC de la universidad el cual proveerá el mantenimiento necesario a las plataformas de atención al usuario pues, el mercado informático ofrece diferentes tipos de soluciones (Mancha, 2012). Ante esta causal, se pueden tomar diferentes soluciones como diseñar un desarrollo propio o la integración de algún existente.

5.2. Marketing Educativo

Relacionar el Marketing Relacional y el CRM en las universidades es algo que hasta hace algunos años era algo poco predecible, pero se ha notado su incursión ha sido muy bien recibida por parte de las instituciones en el mundo ayudando a sus diferentes tareas por medio de software especializado reforzando tareas en las cuales se obtenían dificultades a la hora de su ejecución (Vaquero, Portabales, & de la Fuente, 2018).

En estos últimos tiempos el Marketing Relacional y el CRM, se han enfocado en los diferentes mercados, pero han hecho una nueva incursión en el mercado educativo. Aunque el Marketing por su parte tiene su propio concepto el cual se puede definir como: *“El proceso de investigación de las necesidades de los estudiantes y el desarrollo de proyectos que satisfagan los mismos y que tiene por objetivo la formación integral de estudiantes en sus diferentes carreras a través de servicios de calidad, y ascenso de las instituciones educativas”* (Cataldo, Massaro, & Biglia, 2014), este tipo de servicio educativo no solo debe lograr la satisfacción del estudiante, si no de los padres a través del trabajo realizado (Franco Restrepo & Arrubla Zapata, 2014).

Cabe notar que el marketing educativo puede ser aplicado tanto a universidades privadas como públicas, para cumplir los objetivos pactados por las universidades, las cuales deben contar

con sistemas de información encargados de recoger, analizar, ordenar y la información necesaria de los prospectos para generar el tipo de oferta necesario para la demanda. (Cataldo et al., 2014).

Pocas llegan a incorporar el concepto completo del marketing en una institución, solo usan ciertas herramientas que este ofrece y de las cuales se pueden observar en funcionamiento en las bolsas de empleo, oficina de egresados y los demás beneficios de la institución con tenga en su momento, la implementación del marketing suele dar buenos resultados en empresas, pero no resulta muy bien implementada en las universidades, esto solo depende de los objetivos que quiera lograr (Naranjo Hernández, 2011).

Existe el temor al cambio por parte de las instituciones, y en algunos casos no se cree necesario y solo en algunas circunstancias se hace presente el fracaso vislumbrado por la falta de planificación que causa la toma de decisiones improvisadas las cuales no son analizadas y se presentan día con día (Solís Hurtado, 2004).

.2.1. Herramientas del Marketing Educativo

La comercialización de los servicios por parte de las instituciones educativas, es algo que se desarrolla día con día, por tanto son capaces de identificar qué funciona mejor y reunir esfuerzos en lo cual realmente atrae a los estudiantes (Taide, 2016), para esto hay numerosas actividades que se pueden incluirse dentro de las promociones institucionales para su efectividad contando con una planificación previa y los costos asociados a estas, se debe tener en cuenta que estos incentivos son acordes a la imagen de la institución y a su situación actual, y solo en algunos casos puede perjudicarla haciéndola poco atractiva, este tipo de eventos suelen promoverse en los tiempos adecuados para dar la información necesaria, cabe resaltar que al ser repetitivas, se cae en el cliché

de hacer repulsiva la institución y los incentivos allí ofrecidos, permitiendo caer en errores constantes, como la repetición de promociones de años anteriores, imposibilitando la creación de nuevas y originales, valorando positivamente la demanda que se desea, en este caso la retención y fidelización.(Solís Hurtado, 2004).

En casos como estos la creatividad es primordial, una imagen vale más que mil palabras a la hora de captar o fidelizar a un estudiante, algunas herramientas utilizadas en estos días son los códigos QR para conectar con los prospectos, a través de periódicos, carteles llevando a interactuar de forma diferente con la universidad, y redes como Facebook, Twitter entre otras, son redes sociales con la mayoría de suscriptores en el mundo y de la cual instituciones educativas están haciendo uso de estas para iniciar su reclutamiento por ser redes que se actualizan diariamente, otras redes utilizadas y con gran respuesta ante el público es la plataforma de videos YouTube una de las más visitada en el mundo, también es de gran ayuda para las instituciones promocionando los contenidos, controlando la cantidad de suscriptores, reproducciones permitiendo saber si la publicidad empleada ha sido exitosa. (Taide, 2016).

Podría decirse que las redes sociales son importantes a la hora de implementar estrategias de atracción, pero esto no es lo único que se puede hacer. Las instituciones por su parte también deben tener sus propias tácticas, invitando a colegios, promoviendo las carreras y los proyectos que se están desarrollando, este tipo de logística se debe hacer en los tiempos justos obteniendo respuestas positivas, un contacto personalizado, presentando el portafolio académico, recolectando datos del público interesado, y mostrando el apoyo necesario a carreras que se considere necesiten atención. Con sucesos como estos se busca terminar las brechas entre las instituciones y colegios, ayudando como consultoría a quienes necesitan apoyo en su último año, servicios como estos

demonstrarían que la institución está realmente comprometida con los aspirantes y el servicio que esta presta (BRIONES, 2017), algunos de los planes en estas presentaciones es informar las diferentes posibilidades de hacer parte de la institución, como como la Corporación Universitaria Adventista de Medellín la cual ha implementado el plan de referenciados para la atracción de nueva audiencia es algo que ha dado buenos resultados generando un incentivo entre el cuerpo estudiantil (Castillo Lorente, Martínez, & Marín Sánchez, 2013), logrando buenos resultados en los objetivos propuestos a la hora de la captación y retención dentro y fuera de las instalaciones generando posicionamiento entre las instituciones y la competencia sana entre las mismas.

5.3. Manejo de la Imagen Institucional

El manejo de la institucional es algo muy importante para las estrategias pues su influencia en la sociedad es de vital importancia, al promoverlas se facilita difundir información adicional la cual debe ser realista, una buena o la mala representación, siempre recae en las impresiones de la gente y de las ideas que se hacen respecto a una organización o un servicio como se establece en estos casos (Arroyo López, 2010).

Dar una buena impresión no es fácil, obtenerla proviene de un cambio interno, empezando por la comunidad estudiantil, son ellos quienes se convierten en voceros informando del progreso a su núcleo familiar y social, por lo tanto, estos comentarios suelen ser positivos o negativos hacia esta. La reputación de una organización se establece por la percepción que un grupo de personas tienen sobre estas (Solis Hurtado, 2004), con esta idea se plantea la idea de manejar su propio contenido visual algunas universidades consultadas para esta investigación manejan manuales sobre el manejo de la misma que establece diferentes directrices sobre la misma.

La universidad del Bosque sigue un plan de institucional en el cual manejan su propio lenguaje en términos de comunicación en la proyección de su imagen interna y externa fortaleciendo y mostrando un concepto fuerte de lo propone (Universidad del Bosque, 2009). Los manuales consultados a nivel internacional, nacional y regional abarcan la mayoría de las áreas visuales y está diseñado para ser modificado según las necesidades de la institución o su renovación según el caso.

5.3.1. El Aspirante Como Objeto Principal

Un factor clave en una institución es la relación con su comunidad estudiantil, es importante crear lazos emocionales entre la institución y el personal de la institución, en estos casos debe haber un equilibrio el cual su prioridad es escuchar (García Villalobos, 2006), si se quiere que la institución internamente sea estable es recomendable la creación de diferentes canales de comunicación de las cuales pueden tener diferentes métodos para desarrollar para que no se sientan olvidados, actividades como estas darán la confianza suficiente al estudiante para dar sugerencias puede que estas no se han solucionadas inmediatamente, pero se le tendrán en cuenta como prioridad (Solis Hurtado, 2004). El alumno puede dar su opinión y no solo el, si no la organización pero en algunos casos tanto los estudiantes como los funcionarios no suelen comunicarse pues no lo creen necesario o en algunos casos toman la mentalidad de que no vale la pena pues se piensa que nos son escuchados (Llorente Alonso, 2017).

Es en estos casos es donde se hace necesario el incentivo interno para que no solo la comunidad educativa y la organización respondan a los requerimientos, porque reconociendo el trabajo interno se fomenta la confianza, compromiso y la lealtad de la organización. (López Torres, 2017)

5.4 Presencia de las Instituciones en la Red

Las instituciones son campos de profundo conocimiento, que dan lugar a proporcionar su propio contenido a través de la red, mostrando una visión de su estructura organizacional, departamentos, y servicios en línea que esta contenga. Con presencias sólidas como esta, la comunicación será ágil, y de mayor provecho en casos en los cuales no se puedan visitar físicamente (Roca, 2006), lo que permite generar una imagen positiva, atrayendo nuevos prospectos, promocionando eventos e informando de su catálogo educativo.

La presencia en la red no es algo menor, la usabilidad de estas permiten que estén en continua expansión (Brito, Laaser, & Toloza, 2012), con la aparición de las redes sociales, la relaciones entre las empresas y los clientes es más dinámica, rápida y delicada (Assunção, Silva, & D'Amore, 2015), pero comenzar un primer contacto puede surgir de diferentes maneras, hacer este tipo de trato resulta difícil las primeras veces, se considera que las organizaciones deben hacer presencia en diferentes plataformas, para crear una buena imagen y posicionarla ante la sociedad, de acuerdo con el informe hecho por Hootsuite y We Are Social, en el mundo 3.028 millones de personas están activas en este tipo de medios, de las cuales 2.046 millones utilizan Facebook, como medio de comunicación principal, y se estima que un 40% de la población mundial está usando otra clase (Tecnósfera, 2017), algunas solo cuentan con Facebook institucional en el cual se encuentran ciertas actividades realizadas de las cuales se consignan algunos datos básicos y enlaces al sitio web para detallar la información presentada en Facebook, Aunque suele suceder que la mayor parte de sus seguidores son comunidad estudiantil y algunos interesados.

De acuerdo con esto podemos concluir que la mayoría ha dado el uso correcto de las mismas, dado a su presencia no es la más óptima, pues no incluye información, páginas de bienvenida, sección de videos, fotografías, transmisiones en vivo de eventos realizados por parte de la universidad, interacción en otras redes, y link a información específica, noticias, encuestas etc....(Grupo Educacion al Futuro, 2011), en casos como esto es donde se observa la poca información entregada por la institución resultan las reclamaciones que invaden las redes en segundos acerca de un servicio mal hecho o un producto incompleto y daña seriamente la imagen de la institución, y puede recibir un mensaje relacionado del mismo autor o de alguien secundario (Assunção et al., 2015).

A lo expresado, las redes sociales no se utilizan de la manera adecuada o en muchos de los casos no se da la importancia necesaria a las mismas, falta hacer conciencia de que estas puede medir el fracaso o el éxito de una institución ayudando a evaluar el impacto según su presencia y el manejo de la información relacionada en ella.(Brito et al., 2012).

6. Marco Conceptual

- **CRM (Customer Relationship Management)**

CRM es una estrategia empresarial que pretende establecer y desarrollar el valor de las relaciones con los clientes basada en el conocimiento previo. Usando como un facilitador el CRM que requiere un rediseño de la organización y orientar los procesos hacia el cliente, de modo que mediante la personalización de productos y servicios, la empresa puede satisfacer de manera óptima las necesidades de los clientes y, por ende, generar a largo plazo y mutuamente beneficiosas relaciones interpersonales” (Garrido-Moreno & Padilla-Meléndez, 2011).

- **Fidelización**

La fidelización de clientes consiste en que los compradores o usuarios de los servicios o productos de una empresa mantengan relaciones estables, continuas y de largo plazo con esta. La fidelidad se produce cuando existe una comunicación favorable entre el cliente y la organización y su habitualidad de compra o el uso de servicios lo que demuestra un nivel de satisfacción con la empresa (Mesén Figueroa, 2011).

- **Cliente**

El cliente se define como la persona que adquiere un producto o servicio, este es la parte central y más importante de cualquier empresa u organización, ya que todas las acciones de estas están dirigidas a él, además es el que realmente le da valor a la marca, por lo cual, es el motivo principal por el que se comercializan productos y servicios (Ruiz Correa & Villegas Quiroga, 2013).

- **Seguimiento**

Monitoreo al cliente de forma periódica para que las empresas pueden supervisar los diferentes patrones de interacción con los clientes y obtener una mejor comprensión de la experiencia que se está ofreciendo. Según la información que la compañía está buscando, puede optar por analizar las pautas anteriores, los modelos actuales, posibles patrones, o una combinación de ellos. Cada patrón requiere un método distinto de generar y analizar los datos que se producen en distintos tipos de conocimientos (Meyer & Schwager, 2007).

- **Java**

Es un lenguaje de programación y una plataforma informática comercializada por primera vez en 1995 por Sun Microsystems. Hay muchas aplicaciones y sitios web que no funcionarán a menos que tenga Java instalado y cada día se crean más. Java es rápido, seguro y fiable (Oracle Corporation, 2017).

- **NetBeans**

Es una herramienta que se utiliza para desarrollar aplicaciones Web, Móvil y de escritorio para diferentes lenguajes de programación como son Java, C++, Ruby y PHP entre otros. Es de código abierto, es multiplataforma, multilenguaje, contiene servidores web (Mendoza, 2015).

- **Java Server Faces**

la facilidad de uso, la arquitectura JavaServer Faces define claramente una separación entre la lógica de la aplicación y la presentación, al mismo tiempo que facilita la conexión de la capa de presentación con el código de la aplicación. Este diseño permite a cada miembro de un equipo de desarrollo de aplicaciones web enfocarse en su parte del proceso de desarrollo, y también proporciona un modelo de programación simple para unir las piezas. Por ejemplo, los

desarrolladores de páginas web sin experiencia en programación pueden usar las etiquetas de componentes de interfaz de usuario de JavaServer Faces para vincular el código de la aplicación dentro de una página web sin escribir ningún script (Oracle, 2017).

- **PrimeFaces**

Es una librería de componentes visuales de código abierto para el conjunto Java Server Faces 2.0 desarrollada y mantenida por Prime Technology. Su objetivo principal es ofrecer un conjunto de componentes para facilitar la creación y diseño de aplicaciones web. Los componentes de PrimeFaces cuentan con soporte nativo de Ajax, pero no se encuentra implícito, de tal manera que se tiene que especificar que componentes se deben actualizar al realizar una petición proporcionando así mayor control sobre los eventos. (Pech-May, Gomez-Rodriguez, Cruz-Diaz, & Lara-Jeronimo, 2010).

- **Base de Datos**

Es un conjunto de datos almacenados sin redundancias en un soporte informático y accesible simultáneamente por distintos usuarios y aplicaciones. Los datos deben estar estructurados y almacenados de forma independiente a las aplicaciones que la utilizan. (Cobo Yera, 2007).

- **PostgreSQL**

Es una base de datos objeto-relacional (ORDBMS) del sistema de gestión basado en Postgres, Versión 4.2 1, desarrollado en la Universidad de California en Berkeley, Departamento de Ciencias Informáticas. POSTGRES fue pionero en muchos conceptos que sólo estuvieron disponibles en algunos sistemas de bases de datos comerciales mucho más tarde (The PostgreSQL Global Development Group, 2017).

- **GlassFish**

Es un servidor de aplicaciones desarrollado por Sun Microsystems que implementa las tecnologías de la plataforma Java EE y permite ejecutar aplicaciones. La versión comercial es denominada Sun GlassFish Enterprise Server, de código libre, se distribuye bajo un licenciamiento dual a través de la licencia CDDL y la GNU GPL, está basado en el código fuente de Sun y Oracle Corporation, éste último proporcionó el módulo de persistencia TopLink. Tiene como base al servidor Sun Java System Application Server de Sun Microsystems, un derivado de Apache Tomcat, y que usa un componente adicional llamado Grizzly que usa Java NIO para escalabilidad y velocidad (Calsina Ponce & Cazorla Choque, 2012).

7. Metodología de Investigación

7.1. Investigación Descriptiva

Tiene como finalidad definir, clasificar, catalogar o caracterizar el objeto de estudio, su propósito es conseguir descripciones generales o de objetos específicos. Los métodos descriptivos pueden ser cualitativos o cuantitativos. Los métodos cualitativos son basados en el uso del lenguaje verbal y no recurren a la cuantificación. como principales métodos de la investigación se encuentran el observacional, el de encuestas y los estudios de caso (Universidad de Valencia, 2017).

7.1.1. Etapas del Método Descriptivo

Algunos de los principales pasos a seguir en el método son los siguientes:

-  **Identificación y delimitación del problema:** Es el primer paso, se decide que se va a investigar y qué tipo de preguntas se le buscará una respuesta. (Yanez, 2018)
-  **Construcción de los instrumentos:** de acuerdo con la problemática planteada se deben escoger los diferentes instrumentos y que estos sean adecuados para la recolección de información. Cabe anotar que estos elementos deben ser seleccionados con anticipación. (Yanez, 2018)
-  **Observación y registro de datos:** Implica observar la situación en condiciones naturales, sin interferir tomando la mayor cantidad de detalles posible. (Yanez, 2018)
-  **Categorización de la información:** La información se organiza según su importancia o en su defecto su significado, permitiendo procesar los datos en diferentes categorías sin confundirla. (Yanez, 2018)

- ✚ **Análisis:** Interpretar y analizar los datos, este análisis no debe establecer relaciones pues el método no lo permite. (Yanez, 2018)
- ✚ **Propuesta:** con los datos recabados surgen más preguntas en lo que se denomina normal y se propone una indagación a esta cuestión. (Yanez, 2018)

7.1.2. Características de la Investigación Descriptiva

Se presentan algunas características que determinaron la implementación de la metodología descriptiva en el desarrollo del aplicativo web:

- ✚ Permite obtener información precisa del estudio.
- ✚ La observación debe ser atenta y debe existir un registro de lo observado.
- ✚ Utiliza distintas técnicas de recolección de datos: documentación, observación, encuestas etc.
- ✚ Es un primer abordaje al objeto de estudio.

Referencia General: (Yanez, 2018).

7.2. Nivel de la Investigación

De acuerdo a lo planteado anteriormente la línea de investigación del proyecto es de tipo descriptivo pues con ella se describirá la realidad del objeto de estudio con el fin de esclarecer una verdad, o con el fin de representar situaciones, hechos, personas por medio de palabras (Niño Rojas, 2011), con un método descriptivo podremos contar el proceso de desarrollo del aplicativo web para ayudar al proceso de captación de aspirantes de la Corporación Universitaria del Huila - Corhuila.

7.3. Tipo de Investigación

El tipo de enfoque para la investigación será de tipo cuantitativo permitiendo analizar y recopilar los datos buscando factores o características de la fidelización estudiantil estructurando esquemas que en un futuro faciliten la toma de decisiones respecto a un aspirante en su debido proceso de matrícula.

7.4. Diseño de la Investigación

Para la investigación anteriormente propuesta es necesario diseñar un plan de trabajo que cubra todas las fases de la investigación en sus diferentes procesos para recolección de la información y su debido procesamiento, también se definen las fuentes primarias y secundarias que permiten estructurar la investigación y dar solución a las necesidades encontradas.

7.4.1. Diseño Transversal Descriptivo

Se enfoca en analizar fenómenos en un periodo de tiempo único, su objetivo es describir la incidencia en el lapso establecido (Publicaci & Cardiovasculares, 2013). Aplicando este tipo de diseño el objetivo es recolectar toda la información sobre la fidelización de estudiantes para conocer diferentes condiciones y examinar la tendencia sobre este tema en la educación.

7.5. Población y Muestra

Cuando se trata de la especificación de la población es necesario la identificación de esta, la cual está constituida por la totalidad de los elementos (personas, animales, objetos etc....) (Niño Rojas, 2011).

Descrito lo anterior la división de la población se define:

Tabla 1 Población Involucrada Directamente

Áreas Involucradas	N° de Participantes	Personal Implicado (%)
Coordinación de Marketing	3	80%
Coordinación de Comunicaciones	1	10%
Ventanilla Única	1	10%
TOTAL	5	100%

Nota Fuente: Adaptado de B. J. A. Mocada Morales, “DISEÑO DE UN SISTEMA CRM PARA MEJORAR EL PROCESO DE CAPTACIÓN DE ALUMNOS DE LA UNIVERSIDAD CÉSAR VALLEJO PIURA, 2014.” Universidad Católica de los Ángeles Chimbote, 2016.

La muestra es una porción de una población determinada con el fin de ser estudiada o medir sus características (Niño Rojas, 2011). Debido a la cantidad y nivel de implicación en los procesos de captación y fidelización, la muestra seleccionada es de tres (3) personas, los cuales se relacionan directamente con los procesos internos de monitoreo de la Corporación Universitaria del Huila, para la elección de dicha muestra no se usaron técnicas apropiadas, se tomaron debido a su experiencia en el área y sus necesidades.

La población indirecta que resulta beneficiada del uso de la aplicación, son de 7.540 según datos recolectados por el área de admisiones hasta el 12 de marzo del 2018 comprendido entre los periodos académicos 2017A Y 2017B.

7.6. Recolección de Datos

7.6.1. Datos Primarios

Son utilizados para extraer de forma directa los datos reales (Niño Rojas, 2011) para esta investigación se hicieron uso de dos que se describen a continuación:

7.6.1.1. Observación No Participante

Tiene lugar cuando el investigador queda fuera o es ajeno de la situación observada, sin inmiscuirse lo que facilita la imparcialidad en el asunto (Niño Rojas, 2011). Usando esta técnica se observa la problemática por el área de Marketing a la hora del monitoreo del estudiante debido a que la organización de la información no es la mejor y el tiempo de la ejecución es mayor.

7.6.1.2 Entrevista No Estructurada

Es una técnica de tipo oral, que permite recoger opiniones, puntos de vista y demás el tipo de entrevista utilizado es de tipo informal, no dirigida, adaptable y flexible que permite la libertad de responder.(Niño Rojas, 2011). Se utilizó este tipo de entrevista con el área encargada del monitoreo para recolectar la información necesaria para el desarrollo del aplicativo permitiendo fijar las áreas problemáticas durante su proceso para dar solución.

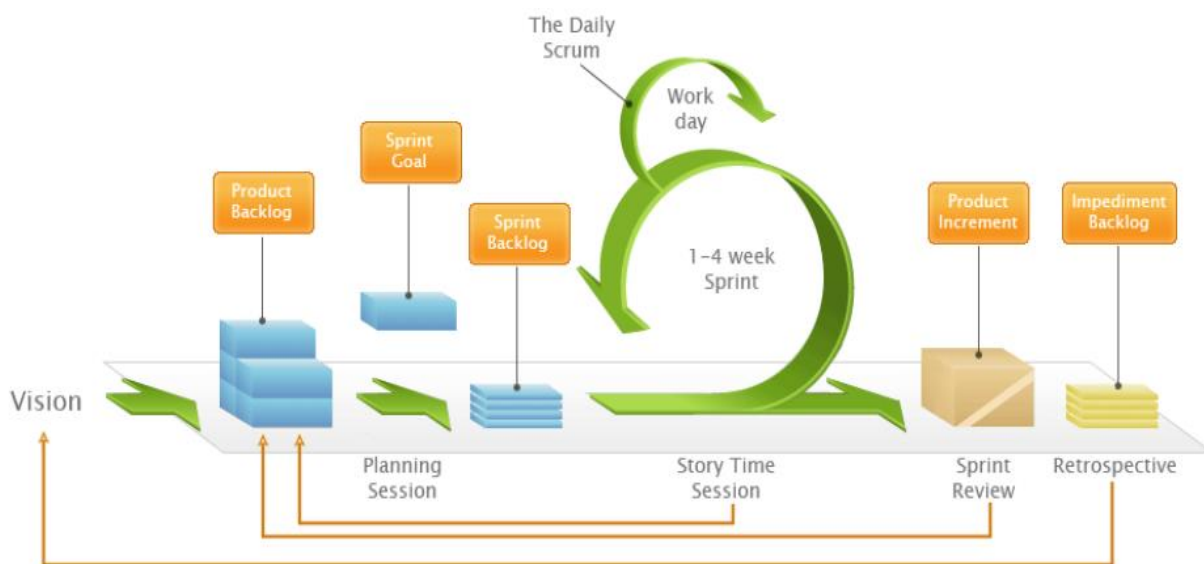
7.6.2. Datos Secundarios

Son los datos recogidos y reelaborados por otras personas, tiene gran variedad de fuentes como las auditivas, electrónicas, videográficas entre otras (Niño Rojas, 2011). La revisión de documentación se hizo a través de Bases de Datos abiertas de tipo académico, tesis, artículos de investigación, libros, manuales, proyectos de grado etc. Sobre el tema en cuestión para fundamentar los diferentes puntos de vista durante el proceso de investigación.

8. Metodología de Desarrollo

8.1. Scrum

Es un marco de referencia dentro de la metodología de desarrollo de software ágil que permite crear software y entregarlo a tiempo de una forma más sencilla. Scrum utiliza el concepto de equipo Scrum, donde los miembros juegan un papel específico. Es por eso que se propone realizar el trabajo en ciclos iterativos cortos de van desde una semana o un mes que generalmente es llamado iteración o sprint, el resultado de cada una de ellas deberá estar listo para entregar, una vez mostrado a los interesados se presenta para la retroalimentación con base a los comentarios se realiza una actualización el producto y los futuros sprint, el beneficio de trabajar con estos entregables es la que cada sprint es un producto listo de tal manera que si un sprint genera inconsistencias o múltiples errores, solamente se revierte a la última entrega evitando que se empiece desde cero (Dimes & Jimenez, 2015).



Nota Fuente: M. Trigas Gallego, "Gestión de Proyectos Informáticos. Metodología Scrum.," *Openaccess.Uoc.Edu*, p. 56, 2012

Ilustración 1 Metodología Scrum

8.1.1. Roles Principales

8.1.1.1. Propietario del Producto o (Product Owner)

Es la única persona responsable de la articulación de los requisitos del cliente y gestionar la lista del producto (Product Backlog o Cartera de Productos) (Trigas Gallego, 2012), su función principal es definir los objetivos y prioridades del programa (Satpathy, 2016). Para el desarrollo del proyecto el Product Owner fue la coordinadora de Marketing Katherine Bonilla Tocora quien proporcionó los requisitos funcionales y no funcionales de la aplicación web para su desarrollo.

8.1.1.2 Scrum Máster

Es el responsable de asegurarse de que el equipo se ajuste a las teorías y prácticas del scrum (Schwaber & Sutherland, 2013), facilitan la creación de entregables, determina el riesgos, cambios e impedimentos durante el proceso de cada iteración. (Satpathy, 2016). El Scrum Máster del equipo fue el ingeniero José Miguel Llanos Mosquera quien lideró los procesos y prácticas durante el desarrollo del aplicativo web.

8.1.1.3 Equipo de trabajo o (Scrum Team)

Consiste en los profesionales encargados de entregar el sprint terminado (Schwaber & Sutherland, 2013), el equipo está formado entre cinco (5) y nueve (9), personas autorizadas para organizar y tomar la mejor decisión para el proyecto (Trigas Gallego, 2012).

Los desarrolladores liderados por el scrum máster fueron: Henry Andrés Perdomo Roa y Martha Milena Vanegas quienes desarrollaron el aplicativo web bajo las teorías y prácticas del scrum en cada una de las fases del Sprint.

8.1.2. Elementos del Scrum

8.1.2.1. Lista de Producto (Product Backlog):

Como su nombre lo dice es una fuente de requisitos para cualquier cambio a realizarse, el Product Owner, es responsable de su contenido, orden y disponibilidad cuando se requiera, aunque nunca está completa solo se reflejan los requisitos conocidos, su evolución es dinámica pues su actualización viene cada vez que hay un sprint. (Schwaber & Sutherland, 2013).

 se presenta algunos de los requisitos solicitados y terminados en la versión V2.05.

TRAZABILIDAD DE CAMBIOS								
SOLICITUD					SOLUCIÓN			
Versión Cambios	Prioridad	N.º	Fecha	Descripción	% Ejecutado	Observaciones	Fecha	Testing
V2.05	Medio	1	15/02/2018	se solicita la vista de matrícula financiera y académica.	100%	La vista Admitidos ahora representa la vista Matricula Financiera, adicionalmente se implementó la vista Matricula Académica.	15-02-2018	SI
V2.05	Medio	3	15/02/2018	Se guarda un registro este registro aparece en aspirantes y en pre-inscritos, pero no aparece en registro aspirante.	100%	Faltaba un campo en el registro, por esto no ejecutaba la sentencia, se modificó el insert inicial para que este campo no se registre vacío.	15-02-2018	SI
V2.05	Medio	4	15/02/2018	falta los historiales de registros subidos desde el Excel, se solicita que se implemente el historial.	100%	Se están creando el historial inicial en la opción de actualizar.	15-02-2018	si
V2.05	Medio	5	15/02/2018	se solicita las actualizaciones del registro aspirante en el resto de vistas	100%	Todas las vistas con los requerimientos solicitados.	15-02-2018	si
V3.00	Medio	6	16/02/2018	Al descargar el reporte que aparezca con el nombre del reporte RepAspXContacto.pdf	100%	Cada reporte queda con su nombre correspondiente	17-02-2018	si

Nota Fuente: Los Autores

Ilustración 2 Lista de Producto (Product Backlog)

8.1.2.2. Lista de pendientes (Sprint Backlog)

Es una predicción hecha por el equipo de desarrollo, la cual cuenta con las funcionalidades del próximo incremento, esta lista se actualiza cada vez que hay un incremento, completando pendientes o añadiendo mas según lo decida el equipo de desarrollo (Schwaber & Sutherland, 2013).

 A continuación, lista de pendientes para la versión V3.09

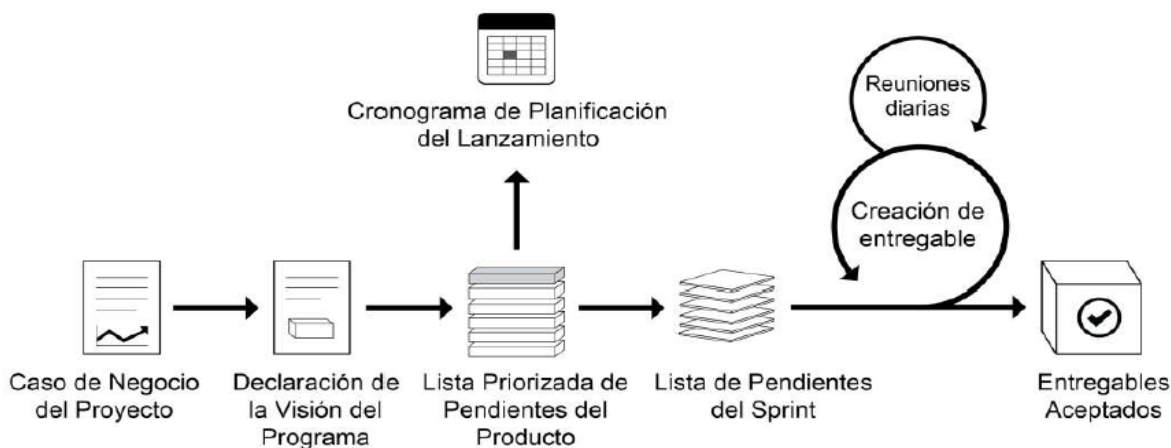
V 3.09		57	20/03/18	Que los que se agregan aparezcan de primero con alerta roja				
V 3.09		58	20/03/18	Que los intereses sean obligatorios a la hora de guardar				
V 3.09		59	20/03/18	Poner diplomados, talleres, cursos y seminarios en checkbox separados de las carreras de pregrado bajo el título de educación continuada y las carreras con título de programa académico				
V 3.09		60	20/03/18	Organizar las cajas de texto de la siguiente forma: Contacto Nombre cedula Teléfono dirección Celular ciudad Correo periodo				
V 3.09		61	20/03/18	Que se pueda editar las fechas del periodo académico				

Nota Fuente: Los Autores

Ilustración 3 Lista de Pendientes (Sprint Backlog)

8.1.3. Incremento (Sprint)

Es una iteración de tiempo asignado entre una y seis semanas en la cual se trabaja en el entregable (Satpathy, 2016), cada nuevo sprint inicia terminado el anterior estos constan de una planificación, un objetivo, reuniones diarias para la sincronización de actividades y una revisión que cuenta con la inspección de lo que se hizo durante el sprint, una vez hecho los pasos anteriores el equipo crea nuevas formas para mejorar en el próximo sprint. (Schwaber & Sutherland, 2013).



Nota Fuente: T. Satpathy, *Una guía para el cuerpo de conocimiento de scrum (Guía SBOK)*. SCRUMstudy™, una marca de VMEdU, Inc, 2016.

Ilustración 4 Planificación de un Sprint

Los sprint para el aplicativo web se dividieron en dos partes la primera parte del sprint consiste en reuniones con el cliente que presenta su lista de requisitos de la aplicación registrando funciones prioritarias. Una vez examinada la lista por el equipo se resuelven dudas que el cliente pueda tener dando soporte y satisfacción. La segunda parte consta de una reunión del equipo la cual prioriza los requisitos y aplica una estrategia para el desarrollo, definiendo tareas entre el equipo creando la lista de Spring Backlog. El tiempo de ejecución de un sprint del aplicativo, consta de una semana y media debido a que los requerimientos solicitados son de baja complejidad.

8.2. Fase de Análisis

8.2.1. Requerimientos Funcionales del Aplicativo Web

UHCRM es el nombre que se le da al aplicativo web para conocimiento interno de las áreas involucradas en el proyecto.

Tabla 2 Requerimiento Funcional RF01

Código de requisito	RF01
Nombre de requisito	Usuario Nuevo
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	El sistema deberá mediante el administrador crear usuarios que interactúen con el aplicativo.
Pre-requisito	Ninguno
Importancia del requisito	Aprobado
Urgencia del requisito	Alto
Estabilidad	Alto
Estado	Alto
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

Tabla 3 Requerimiento Funcional RF02

Código de requisito	RF02
Nombre de requisito	Rol y Permisos
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	El sistema deberá definir roles y permisos a los usuarios directos que sean creados, Es necesario definir los roles a los usuarios nuevos para generar restricciones de los módulos del software.
Pre-requisito	RF01
Importancia del requisito	Aprobado
Urgencia del requisito	Alto
Estabilidad	Alto
Estado	Alto
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

Tabla 4 Requerimiento Funcional RF03

Código de requisito	RF03
Nombre de requisito	Modificar Usuario
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	El sistema permitirá realizar modificaciones a los usuarios creados, cambiar permisos, y eliminar los usuarios no activos, es necesario alternar los usuarios que estén en uso, para generar seguridad del sistema.
Pre-requisito	RF01
Importancia del requisito	Aprobado
Urgencia del requisito	Alto
Estabilidad	Alto
Estado	Alto
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

Tabla 5 Requerimiento Funcional RF04

Código de requisito	RF04
Nombre de requisito	Ingreso al sistema
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	El sistema deberá validar (usuario - contraseña) y registrar los ingresos que realicen los usuarios, Es necesario definir un control al ingreso al sistema y llevar un registro de los ingresos por tema de seguridad.
Pre-requisito	RF01
Importancia del requisito	Aprobado
Urgencia del requisito	Alto
Estabilidad	Alto
Estado	Alto
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

Tabla 6 Requerimiento Funcional RF05

Código de requisito	RF05
Nombre de requisito	Nuevo Registro
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	El sistema deberá permitir crear registros de los aspirantes que solicitan información de los programas académicos, los registros son necesarios para realizar el respectivo seguimiento y apoyo en el proceso de inscripción.
Requisito Derivado	RF05.1 El sistema deberá identificar los aspirantes repetidos para notificar al usuario y definir si crear un registro adicional.
Importancia del requisito	Aprobado
Urgencia del requisito	Alto
Estabilidad	Alto
Estado	Alto
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

Tabla 7 Requerimiento Funcional RF06

Código de requisito	RF06
Nombre de requisito	Actualizar Registro
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	El sistema deberá actualizar los registros de los aspirantes con información suministrada por CORHUILA PLUS cada vez que el administrador lo requiera, es necesario que los registros se encuentren en una constante actualización para que se trabaje con datos reales y no trabajar con información antigua.
Pre-requisito	RF06.1 El sistema deberá actualizar las etapas de los registros, para identificar el proceso en el que se encuentra. RF06.2 El sistema deberá actualizar las alertas de inactividad de los registros que no han sido contactados. RF06.3 El sistema deberá actualizar las alertas de inactividad de los registros que no ha presentado movimiento en los últimos días.
Importancia del requisito	Aprobado
Urgencia del requisito	Alto
Estabilidad	Alto
Estado	Alto
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

Tabla 8 Requerimiento Funcional RF07

Código de requisito	RF07
Nombre de requisito	Modificar Registro
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	El sistema deberá permitir modificar los registros de forma especializada por el usuario encargado, Es necesario mantener los registros en constante actualización y determinar los datos de contacto del aspirante.
Pre-requisito	RF01
Importancia del requisito	Aprobado
Urgencia del requisito	Alto
Estabilidad	Alto
Estado	Alto
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

Tabla 9 Requerimiento Funcional RF08

Código de requisito	RF08
Nombre de requisito	Trazabilidad Registro (Historial)
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	El sistema deberá almacenar los movimientos del registro hasta llegar a la fase final, Es necesario llevar una trazabilidad por las determinadas fases para poder hacer un seguimiento especializado.
Pre-requisito	RF01
Importancia del requisito	Aprobado
Urgencia del requisito	Alto
Estabilidad	Alto
Estado	Alto
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

8.2.2. Requerimientos No Funcionales del Aplicativo Web

Tabla 10 Requerimiento No Funcional RNF01

Código de requisito	RNF01
Nombre de requisito	Identidad de la aplicación
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	La interfaz de usuario debe manejar los colores institucionales de la Corporación Universitaria del Huila (Verde y Blanco)
Importancia del requisito	Aprobado
Urgencia del requisito	Bajo
Estabilidad	Medio
Estado	Moderado
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

Tabla 11 Requerimiento No Funcional RNF02

Código de requisito	RNF02
Nombre de requisito	Navegación
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	El sistema debe ser accesible en sus diferentes vistas
Importancia del requisito	Aprobado
Urgencia del requisito	Alto
Estabilidad	Alto
Estado	Alto
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

Tabla 12 Requerimiento No Funcional RNF03

Código de requisito	RNF03
Nombre de requisito	Compatibilidad
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	El sistema debe ser accesible en diferentes navegadores para observar que su funcionalidad es la misma que en el navegador predeterminado.
Importancia del requisito	Aprobado
Urgencia del requisito	Alto
Estabilidad	Alto
Estado	Alto
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

Tabla 13 Requerimiento No Funcional RNF04

Código de requisito	RNF04
Nombre de requisito	Rendimiento
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	La aplicación tendrá un tiempo de respuesta no mayor a los 10 segundos ante una petición (generar vistas y actualización de los datos).
Importancia del requisito	Aprobado
Urgencia del requisito	Alto
Estabilidad	Alto
Estado	Alto
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

Tabla 14 Requerimiento No Funcional RNF05

Código de requisito	RNF05
Nombre de requisito	Retroalimentación
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	Al ser un sistema alimentable de corhuila plus los datos obtenidos serán actualizados de forma remota al aplicativo.
Importancia del requisito	Aprobado
Urgencia del requisito	Alto
Estabilidad	Alto
Estado	Alto
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

Tabla 15 Requerimiento No Funcional RNF06

Código de requisito	RNF06
Nombre de requisito	Escalable
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	Al ser escalable los módulos empleados en un futuro serán adicionados sin ningún problema.
Importancia del requisito	Aprobado
Urgencia del requisito	Alto
Estabilidad	Alto
Estado	Alto
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

Tabla 16 Requerimiento No Funcional RNF07

Código de requisito	RNF07
Nombre de requisito	Usabilidad
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	Al ser de fácil uso permitirá el ahorro de tiempo en procesos como registro y seguimiento.
Importancia del requisito	Aprobado
Urgencia del requisito	Alto
Estabilidad	Alto
Estado	Alto
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

Tabla 17 Requerimiento No Funcional RNF08

Código de requisito	RNF08
Nombre de requisito	Disponibilidad
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	La aplicación tendrá una disponibilidad mínima de 11 personas trabajando al tiempo, al ser un sistema con altos volúmenes información a mostrar se decidió que el uso de la aplicación sea de pocas personas para evitar caídas de la aplicación
Importancia del requisito	Aprobado
Urgencia del requisito	Alto
Estabilidad	Alto
Estado	Alto
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

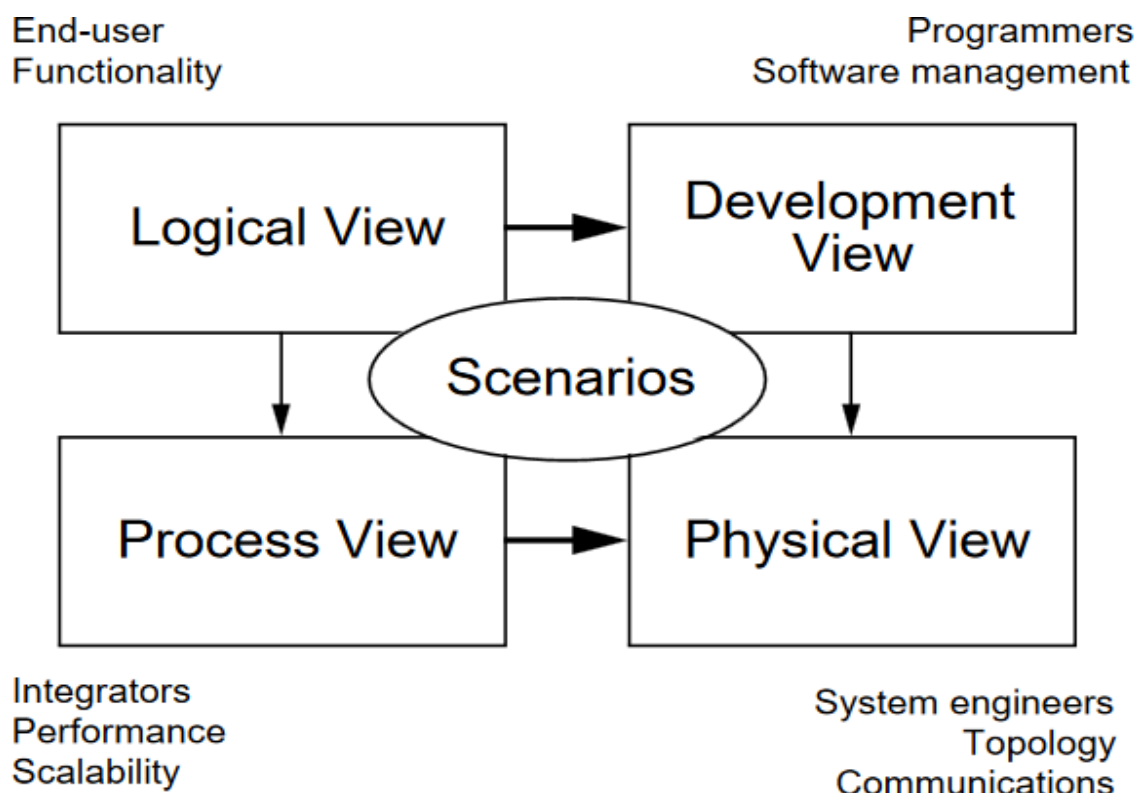
Tabla 18 Requerimiento No Funcional RNF09

Código de requisito	RNF09
Nombre de requisito	Manual de Usuario
Responsable	Henry Andrés Perdomo Roa
Descripción del requisito	La aplicación debe contar con un manual en caso de que el usuario olvide alguno de los procesos en la aplicación web.
Importancia del requisito	Aprobado
Urgencia del requisito	Alto
Estabilidad	Alto
Estado	Alto
Comentarios	Ninguno
Autor	Martha Milena Vanegas Capera
Fecha	23-06-2017

8.3. Fase de Diseño

8.3.1. Modelo 4 + 1 de Philippe Kruchten

La siguiente arquitectura de software trabaja en el diseño y ejecución de estructuras, el resultado consiste en combinar un cierto número de elementos arquitectónicos para satisfacer los principales requisitos de funcionalidad y rendimiento, así como algunos otros requisitos no funcionales, tales como la fiabilidad, escalabilidad, portabilidad y disponibilidad (Krunten, 1995), en el desarrollo del aplicativo se utilizó un modelo compuesto de múltiples puntos de vista o perspectivas, a fin de poder abordar grandes arquitecturas y desafiantes, el modelo que se propuso compone de cinco vistas principales:

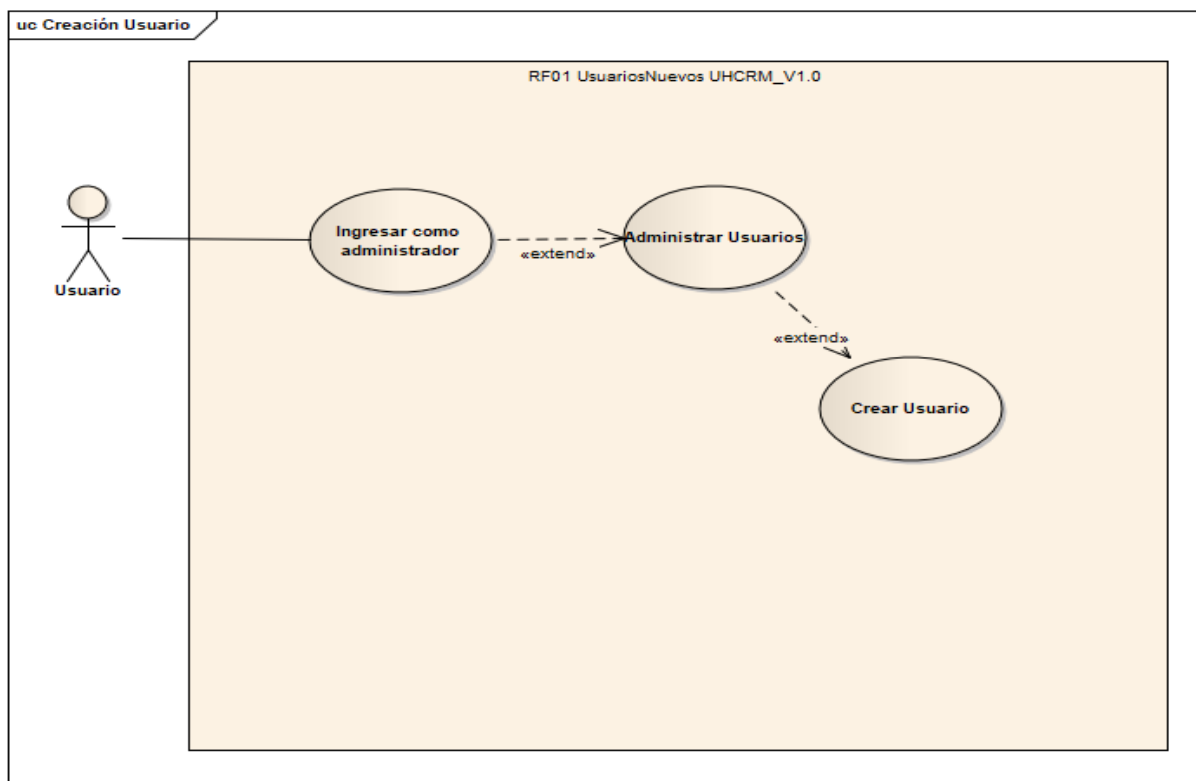


Nota Fuente: P. Kruntchen, "Architectural blueprints—the 4+ 1" view model of software architecture," *IEEE Softw.*, vol. 12, no. November, pp. 42–50, 1995.

Ilustración 5 Modelo 4 + 1 de Philippe Kruchten

Tabla 19 Caso de Uso C01

C01	Usuario Nuevo
[Versión]	<UHCRM_V1.0> (<23-06-2017 >)
[Dependencias]	RF01. Usuario Nuevo
Precondición	Para poder crear un nuevo usuario, es necesario realizar esta acción desde el administrador, ya que es el único que tiene permisos otorgados para hacerlo.
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso [Nuevos Usuarios] cuando El administrador requiera desde la vista Control Usuarios y a través del módulo usuarios, sobre el indicador de “Nuevo Usuario”. estos usuarios nuevos serán necesarios para la realización de otros casos de uso.
[Importancia]	Es importante disponer de usuarios que controlen las funcionalidades del Software.

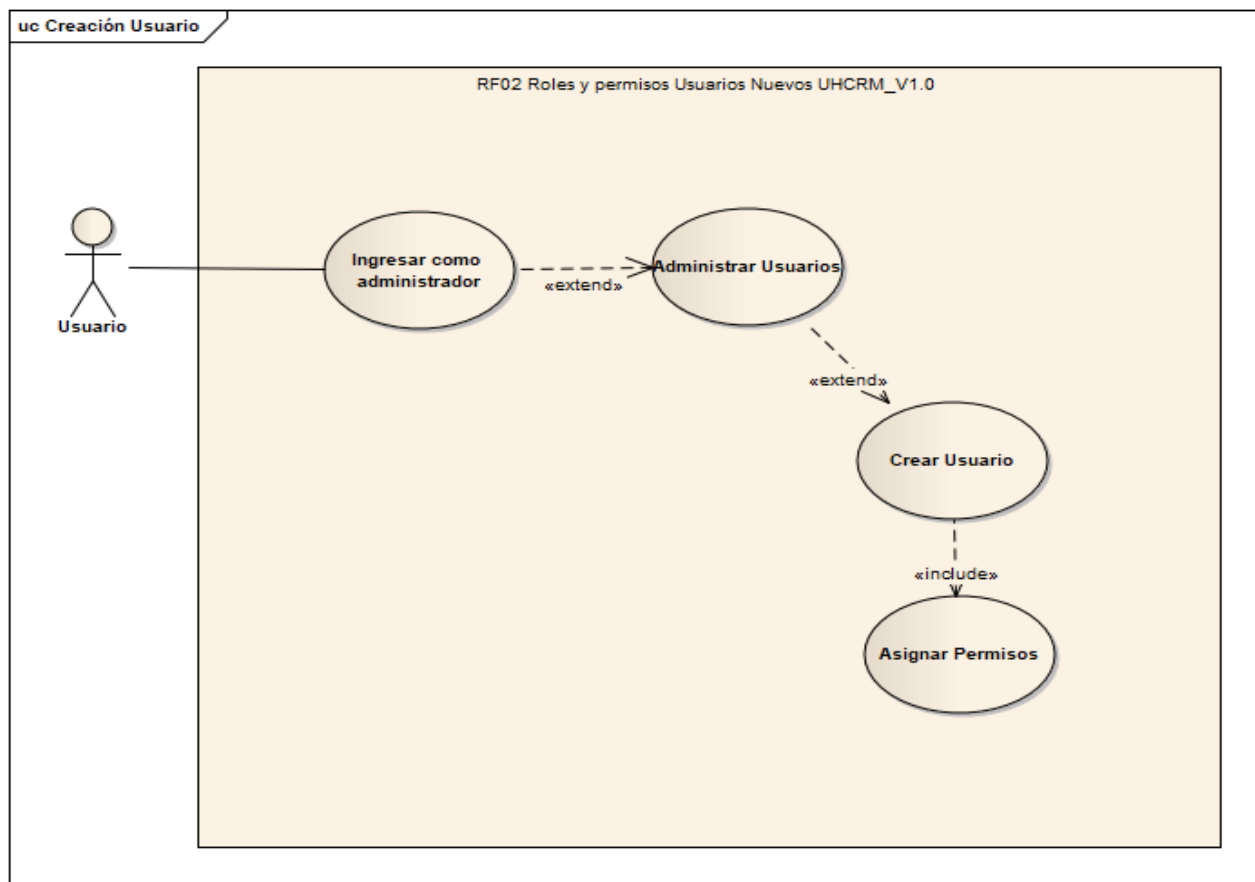


Nota Fuente: Los Autores

Ilustración 7 Caso de Uso C01

Tabla 20 Caso de Uso C02

C02	Roles y Permisos
[Versión]	<UHCRM_V1.0> (<23-06-2017 >)
[Dependencias]	RF02. Roles y permisos Usuarios Nuevos.
Precondición	Para poder definir el rol de un usuario, es necesario realizar esta acción desde el administrador e igualmente contar con el usuario ya creado.
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso [Roles y permisos Usuarios Nuevos] cuando El administrador requiera desde la vista Control Usuarios y a través del módulo usuarios, sobre el indicador de “Nuevo Usuario” o la modificación de uno
[Importancia]	Es importante asignar permisos mediante los roles de los usuarios para poder controlar las acciones que se realicen.

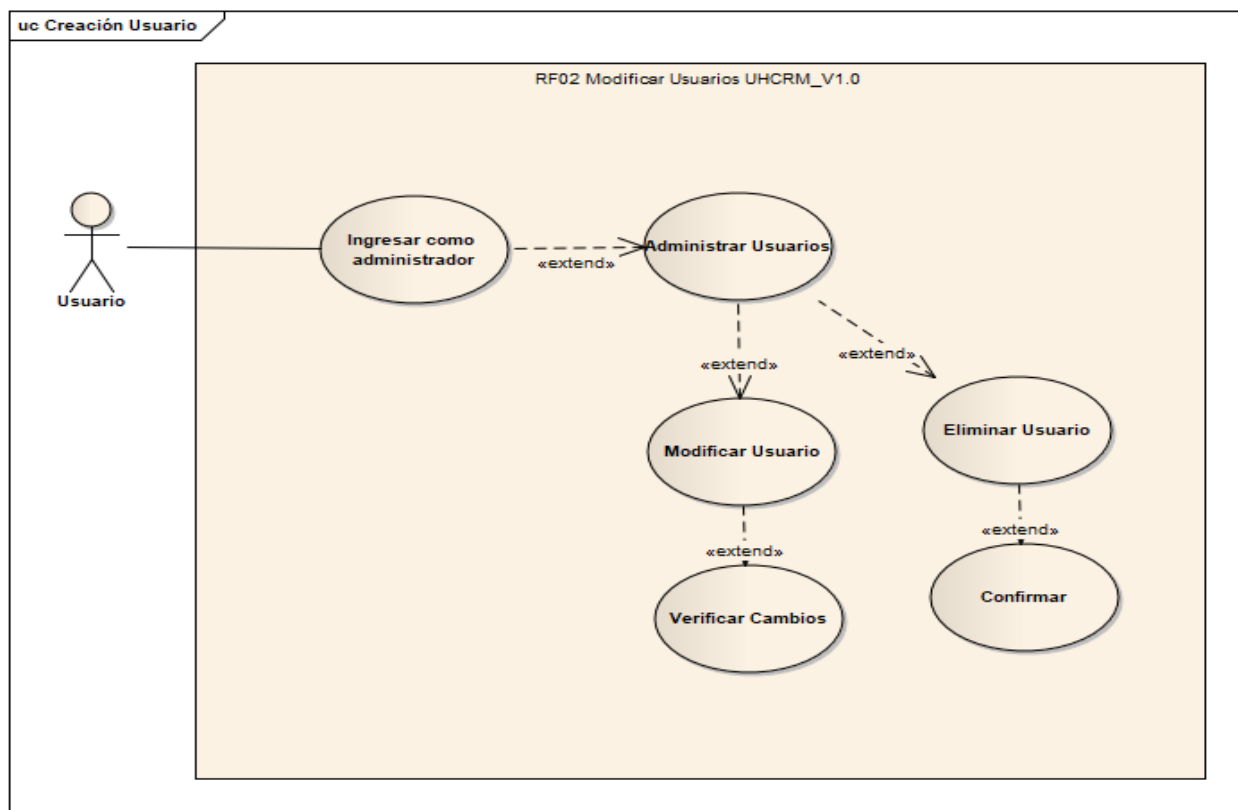


Nota Fuente: Los Autores

Ilustración 8 Caso de Uso C02

Tabla 21 Caso de Uso C03

C03	Modificar Usuarios
[Versión]	<UHCRM_V1.0> (<23-06-2017 >)
[Dependencias]	RF03. Modificar Usuarios.
Precondición	Para poder modificar un usuario, es necesario realizar esta acción desde el administrador e igualmente contar con el usuario ya creado.
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso [Modificar Usuarios] cuando El administrador requiera desde la vista Control Usuarios y a través del módulo usuarios, al editar los datos ya existentes del usuario.
[Importancia]	Es importante mantener los usuarios actualizados y alternar los permisos mediante los roles de los usuarios para poder controlar las acciones que se realicen.

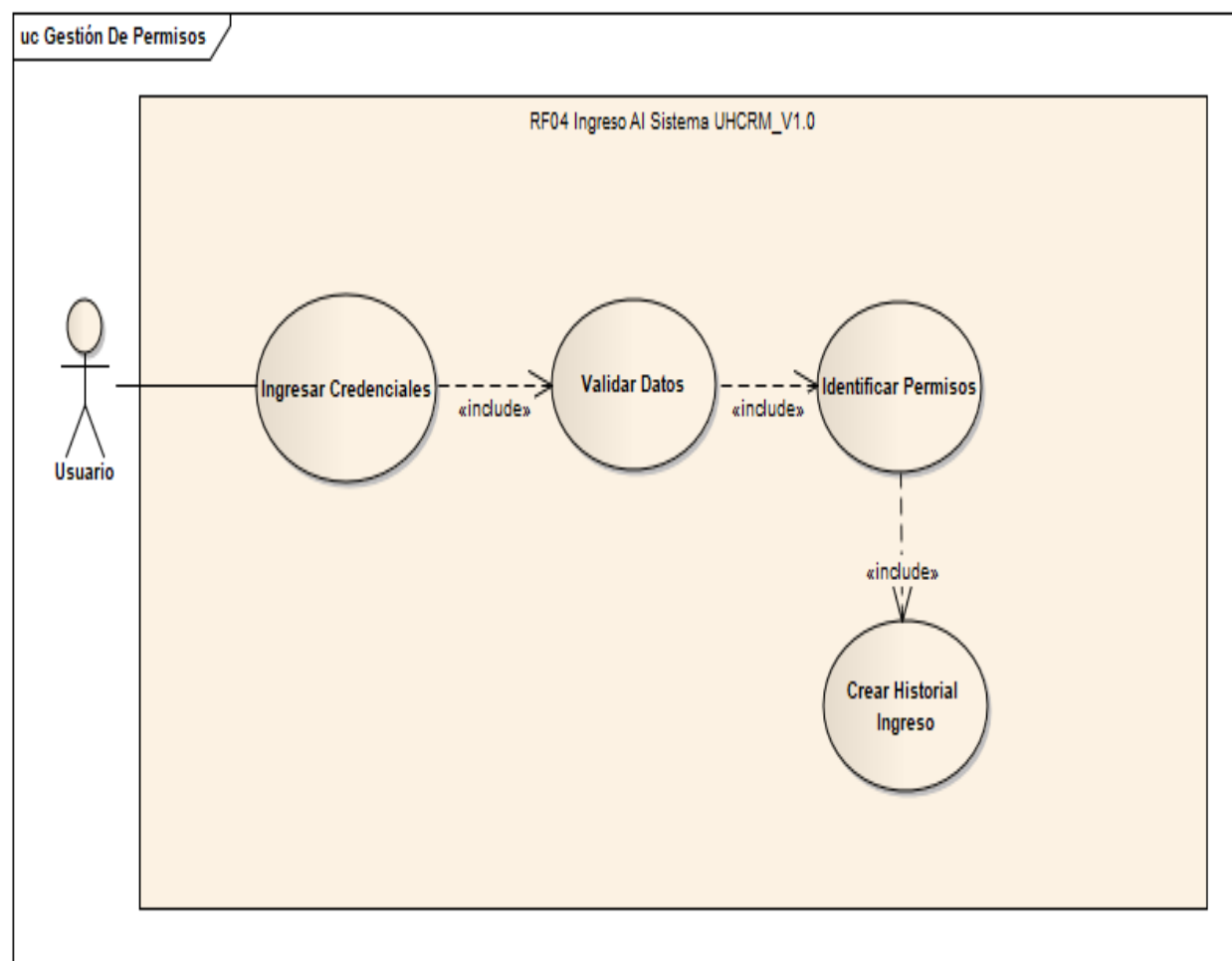


Nota Fuente: Los Autores

Ilustración 9 Caso de Uso C03

Tabla 22 Caso de Uso C04

C04	Ingreso al sistema.
[Versión]	<UHCRM_V1.0> (<23-06-2017 >)
[Dependencias]	RF04. Ingreso al sistema.
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso [Ingreso al sistema.] cuando Él Usuario lo requiera desde la vista Ingreso. Para esto debe ingresar las credenciales y el sistema lo validará, posterior a esto le asignará sus permisos y creará un registro historial
[Importancia]	Es necesario definir un control al ingreso al sistema y llevar un registro de los ingresos por tema de seguridad.

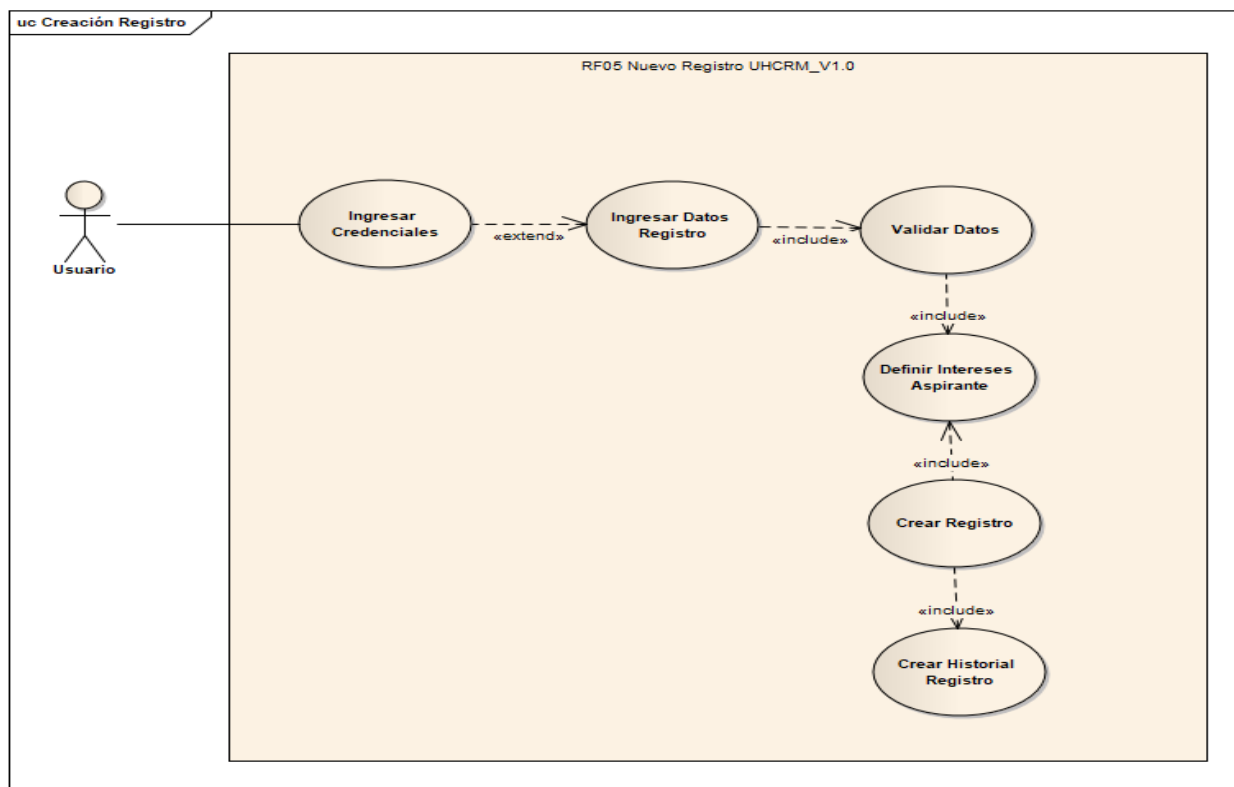


Nota Fuente: Los Autores

Ilustración 10 Caso de Uso C04

Tabla 23 Caso de Uso C05

C05	Nuevo Registro.
[Versión]	<UHCRM_V1.0> (<23-06-2017 >)
[Dependencias]	RF05. Nuevo Registro.
Precondiciones	Para poder crear un nuevo registro es necesario haber iniciado sesión con cual sea de los usuarios previamente registrados.
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso [Nuevo Registro] cuando Él Usuario lo requiera desde la vista Registro. Para esto debe ingresar los datos del registro, el sistema los validará tomando de referencia el interés del aspirante y validando que no exista con antiguos registros. Posteriormente se creará un registro junto a un nuevo elemento de historial que está ligado al registro.
[Importancia]	Los registros son necesarios para realizar el respectivo seguimiento y apoyo en el proceso de inscripción.

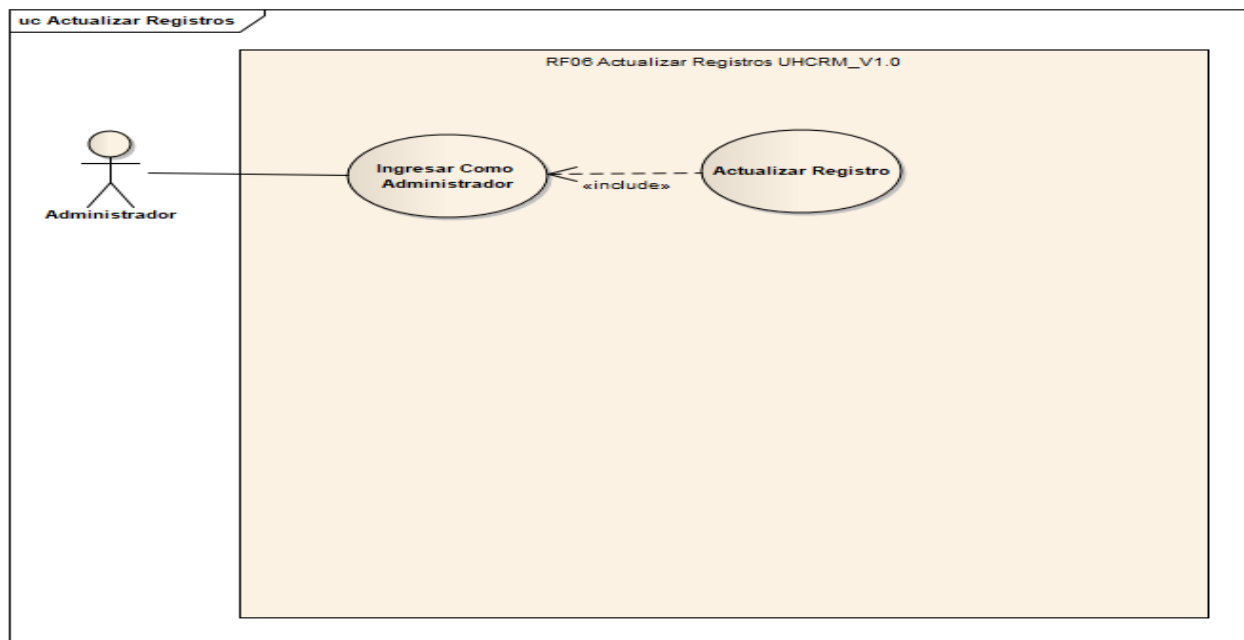


Nota Fuente: Los Autores

Ilustración 11 Caso de Uso C05

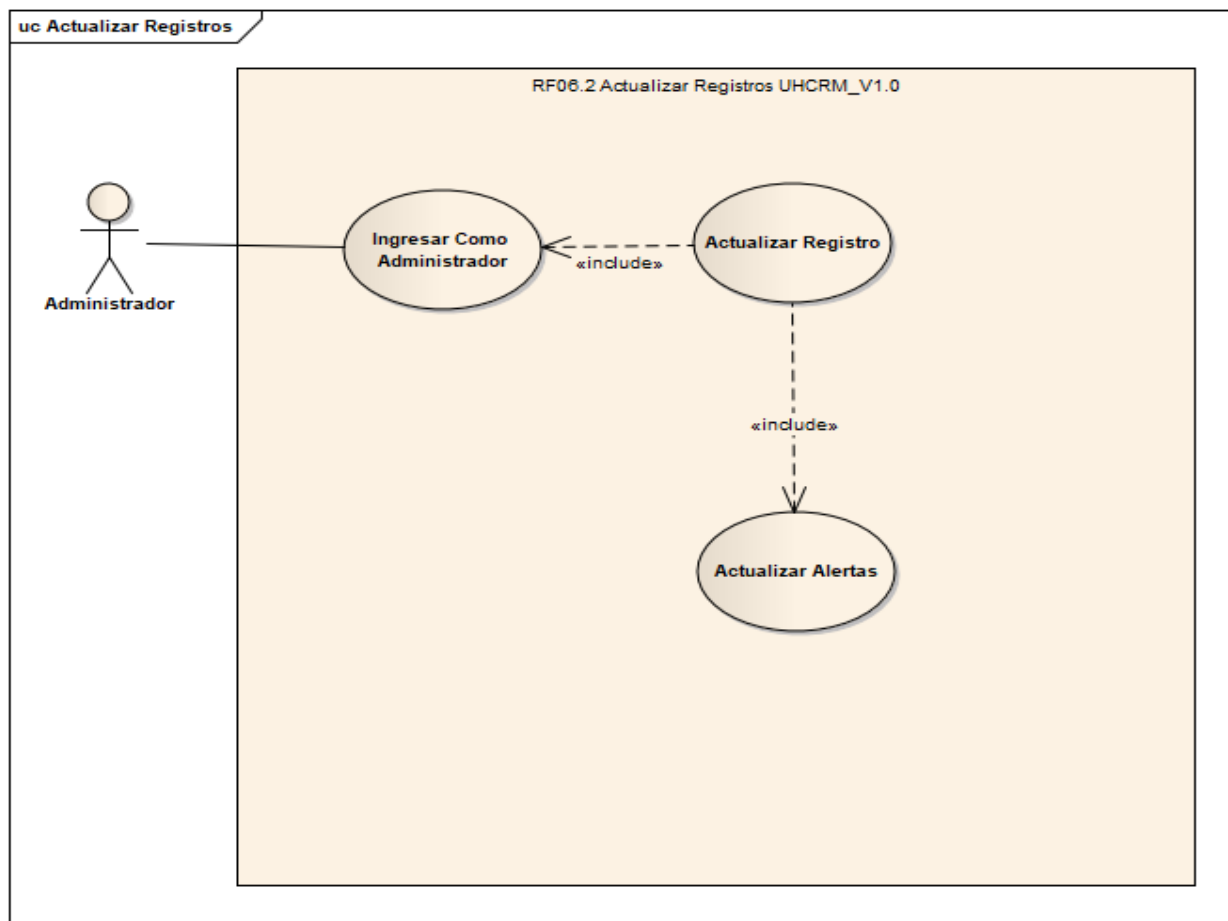
Tabla 24 Caso de Uso C06

C06	Actualizar Registros y Alertas
[Versión]	<UHCRM_V1.0> (<23-06-2017 >)
[Dependencias]	RF06.2 Actualizar Alertas
Precondiciones	Para poder crear un nuevo registro es necesario haber iniciado sesión con cual sea de los usuarios previamente registrados.
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso [Actualizar Registros] cuando Él Usuario lo requiera desde la vista Registros Aspirantes. Para esto el sistema deberá actualizar los registros de los aspirantes con información suministrada por CORHUILA PLUS donde actualizará los datos del aspirante, la etapa del registro y las alertas cada vez que el administrador lo requiera.
[Importancia]	Es necesario que los registros se encuentren en una constante actualización para que se trabaje con datos reales y no trabajar con información antigua.



Nota Fuente: Los Autores

Ilustración 12 Caso de Uso C06

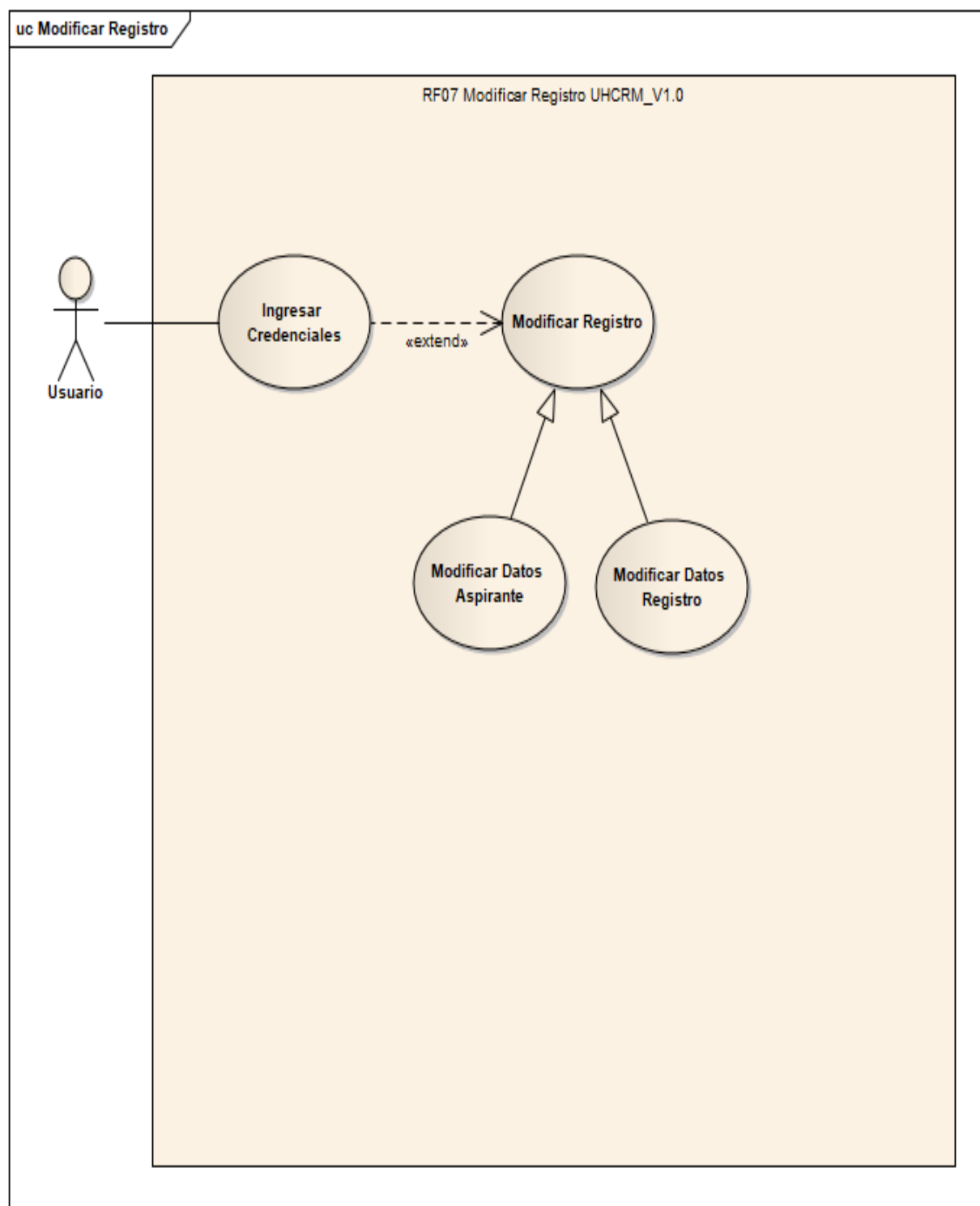


Nota Fuente: Los Autores

Ilustración 13 Caso de Uso C06.2

Tabla 25 Caso de Uso C07

C07	Modificar Registro
[Versión]	<UHCRM_V1.0>(<23-06-2017 >)
[Dependencias]	RF07. Modificar Registro.
Precondiciones	Para poder crear un nuevo registro es necesario haber iniciado sesión con cual sea de los usuarios previamente registrados.
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso [Modificar Registro] cuando Él Usuario lo requiera desde la vista Registros Aspirantes. Para esto el sistema podrá modificar los datos del aspirante o los datos del registro.
[Importancia]	Es necesario mantener los registros en constante actualización y determinar los datos de contacto del aspirante.

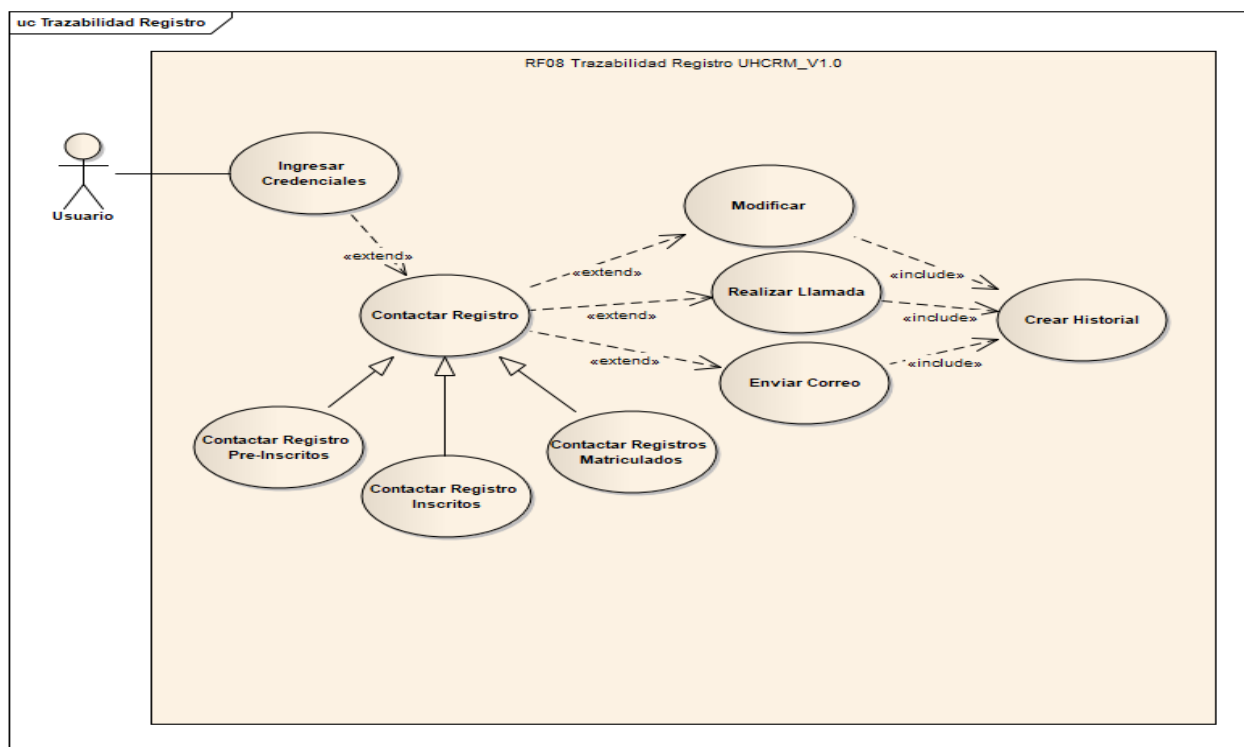


Nota Fuente: Los Autores

Ilustración 14 Caso de Uso C07

Tabla 26 Caso de Uso C08

C08	Trazabilidad Registro (Historial)
[Versión]	<UHCRM_V1.0> (<23-06-2017 >)
[Dependencias]	RF08. Trazabilidad Registro (Historial).
Precondiciones	Para poder crear un nuevo registro es necesario haber iniciado sesión con cual sea de los usuarios previamente registrados. Haber realizado alguna acción sobre el registro (Modificación, actualización)
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso [Trazabilidad Registro] cuando Él Usuario lo requiera desde la vista Registros Aspirantes. Para esto el sistema podrá generar un elemento del historial del registro una vez que el usuario haya realizado una llamada, enviado un correo, o una modificación.
[Importancia]	Es necesario llevar una trazabilidad por las determinadas fases para poder hacer un seguimiento especializado.



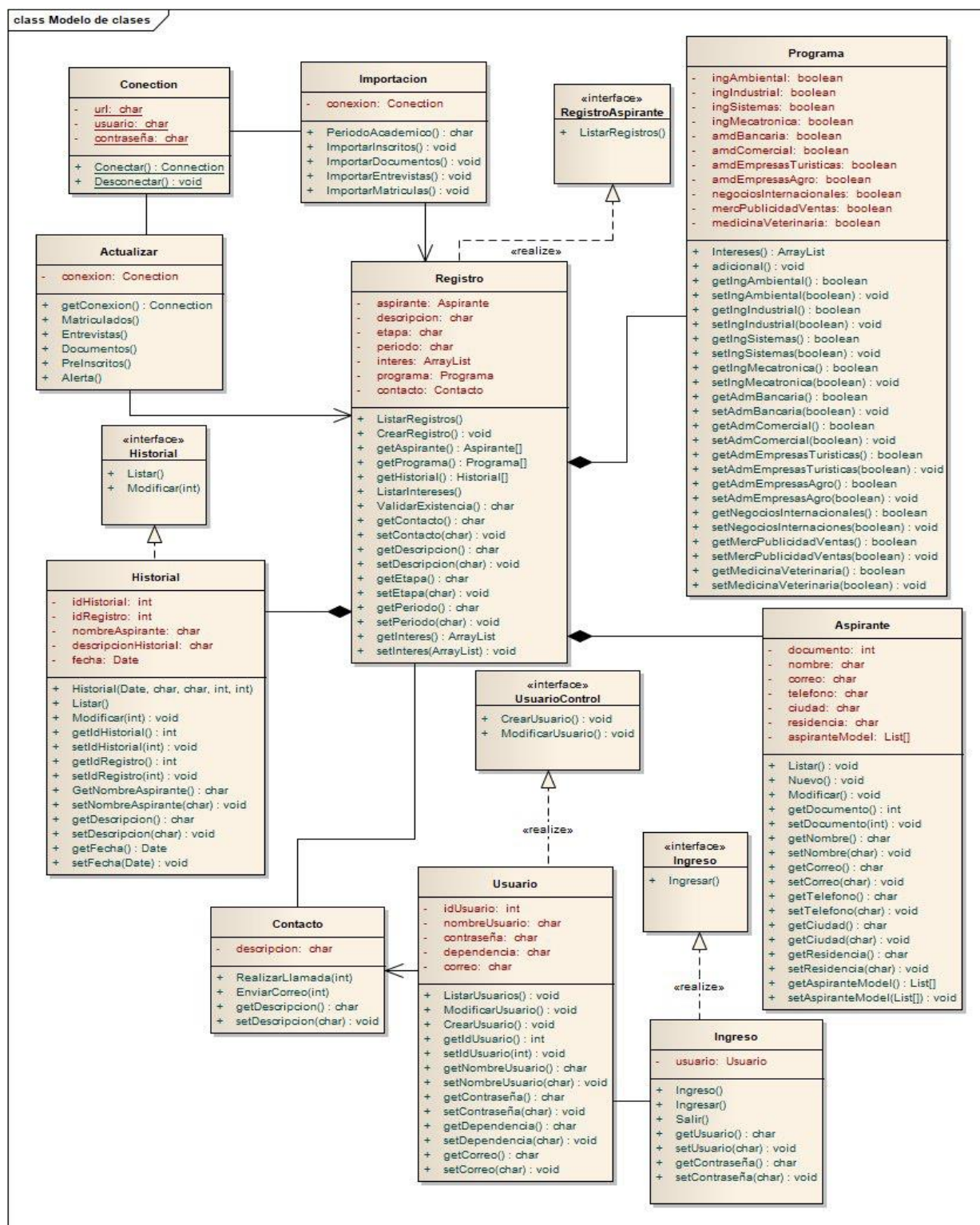
Nota Fuente: Los Autores

Ilustración 15 Caso de Uso C08

8.3.1.2. Vista Lógica o (Logical View)

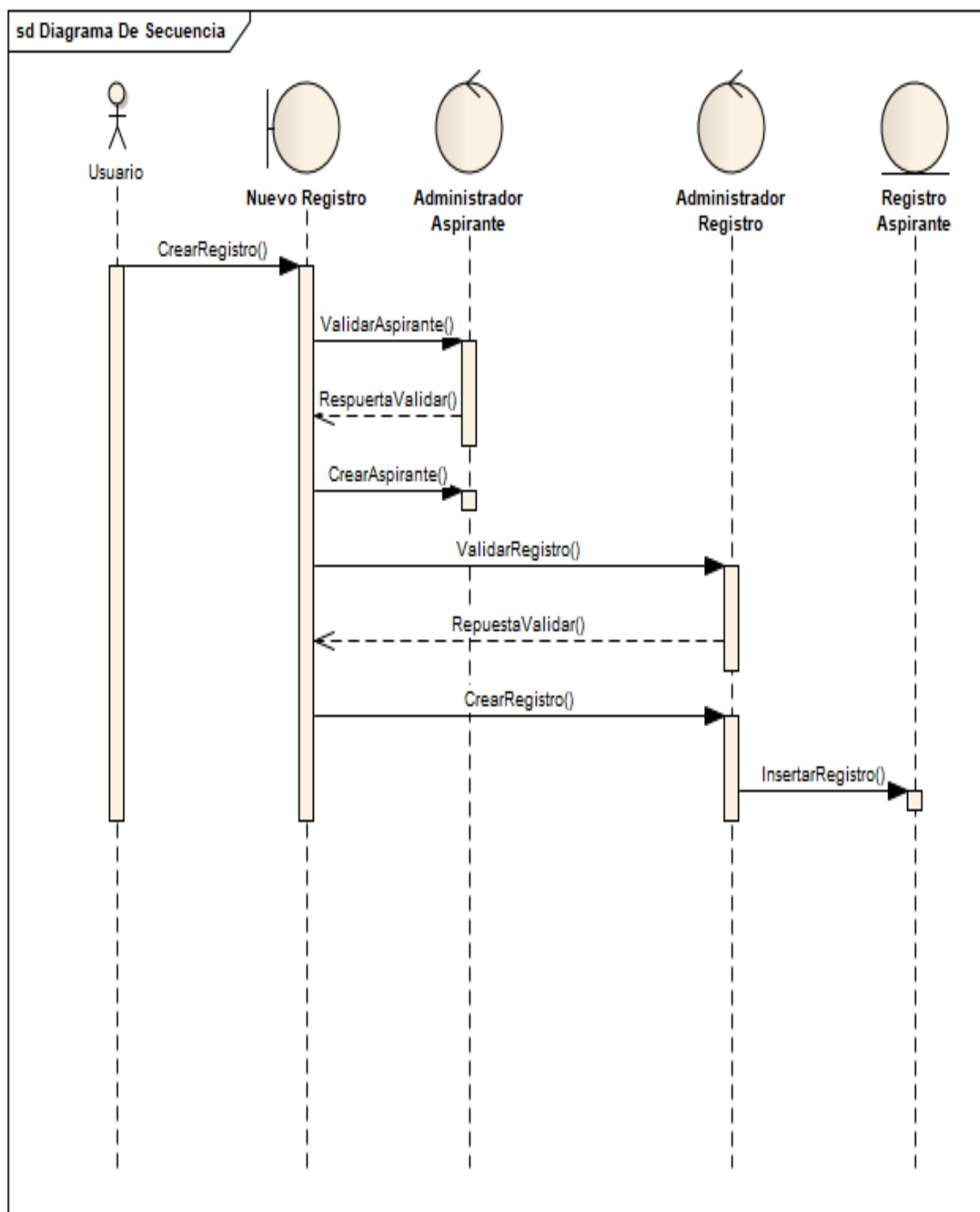
la arquitectura se apoya en los requisitos funcionales de lo que el sistema debe proporcionar en términos de servicios. El sistema se descompone en un conjunto de abstracciones clave tomadas desde el dominio del problema, en forma de objetos o clases de objetos, lo cual permite identificar mecanismos comunes y elementos de diseño a través de las diversas partes del sistema. (Kruntschen, 1995).

 Diagrama de clases



Nota Fuente: Los Autores

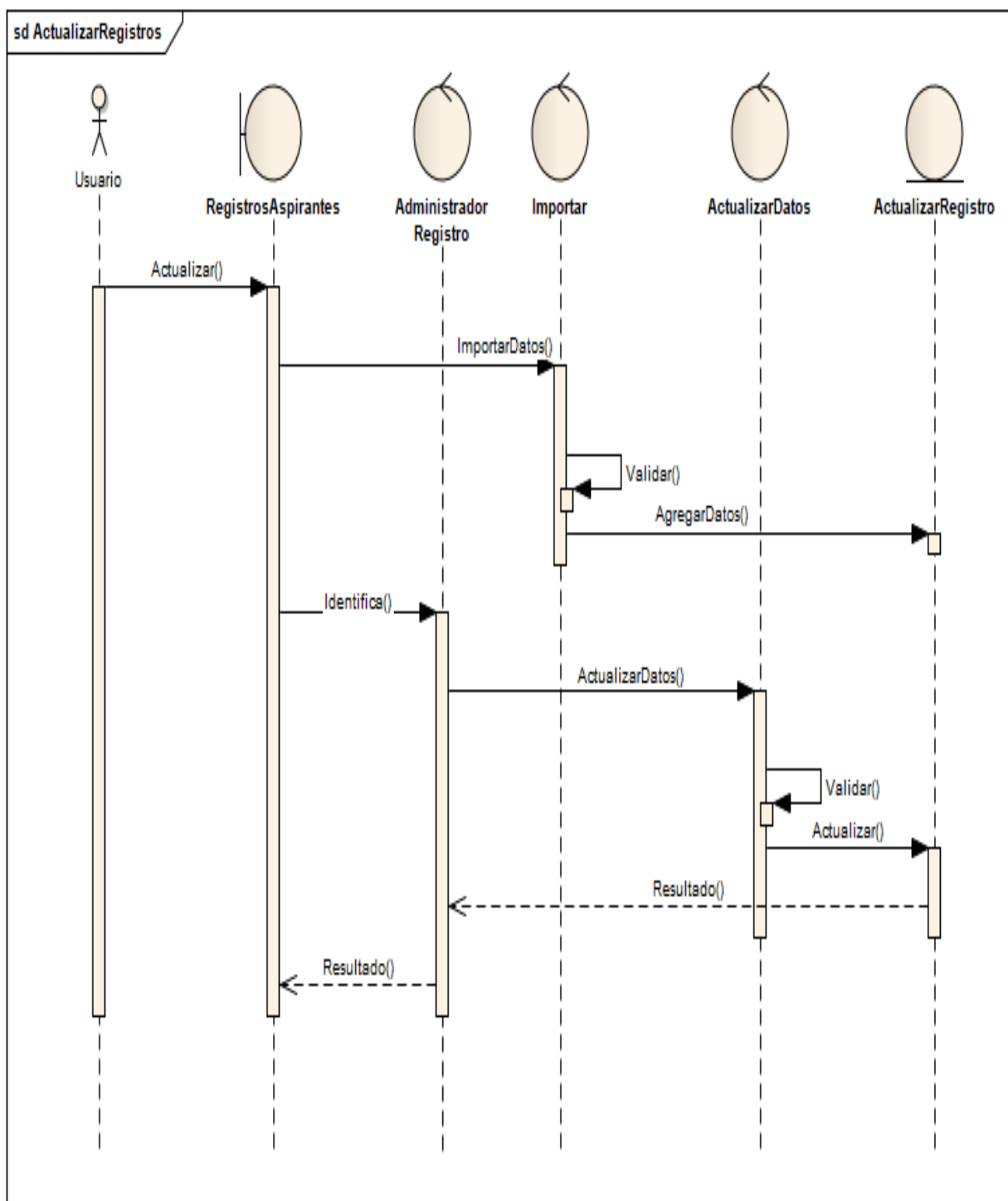
Ilustración 16 Diagrama de Clases

 Diagrama de secuencia Crear Registro

Nota Fuente: Los Autores

Ilustración 17 Diagrama de Secuencia DS01

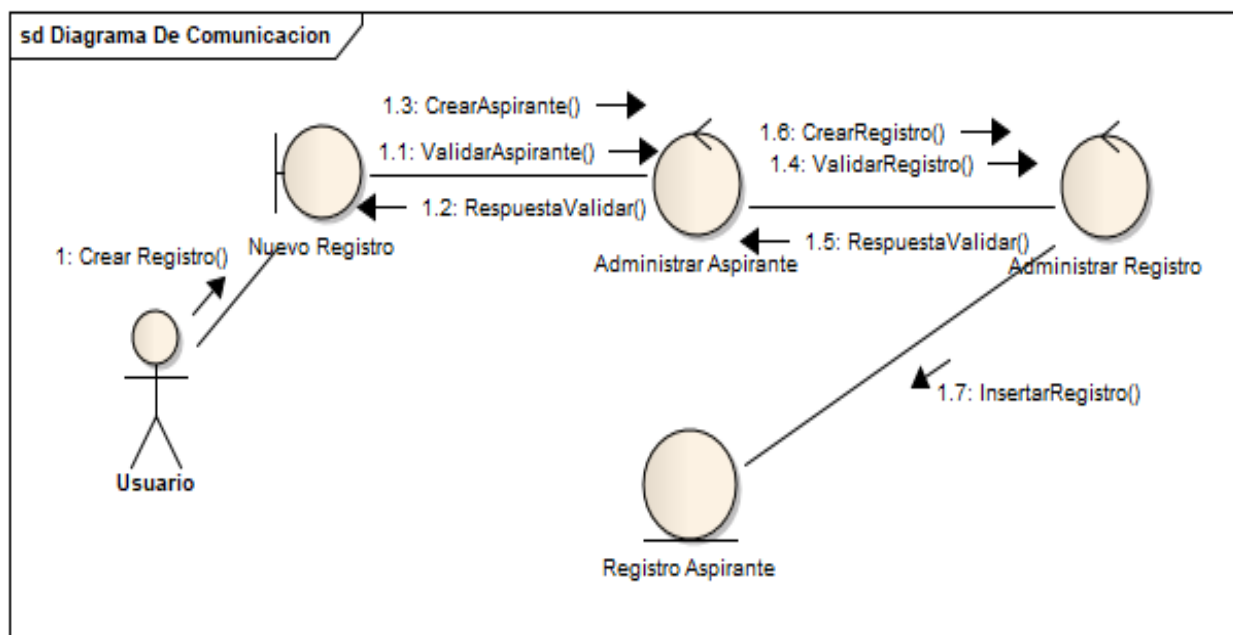
 Diagrama de Secuencia Actualizar Registro



Nota Fuente: Los Autores

Ilustración 18 Diagrama de Secuencia DS02

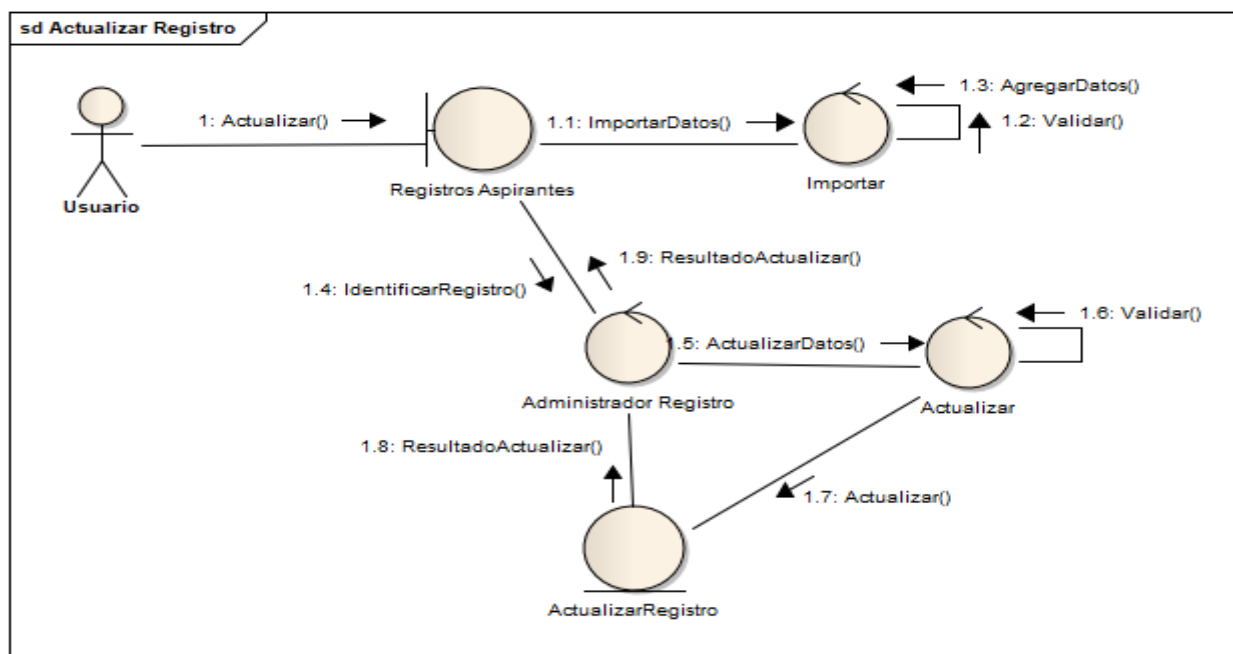
Diagrama de Comunicación Crear Registro



Nota Fuente: Los Autores

Ilustración 19 Diagrama de Comunicación DC01

Diagrama de Comunicación Actualizar Registro

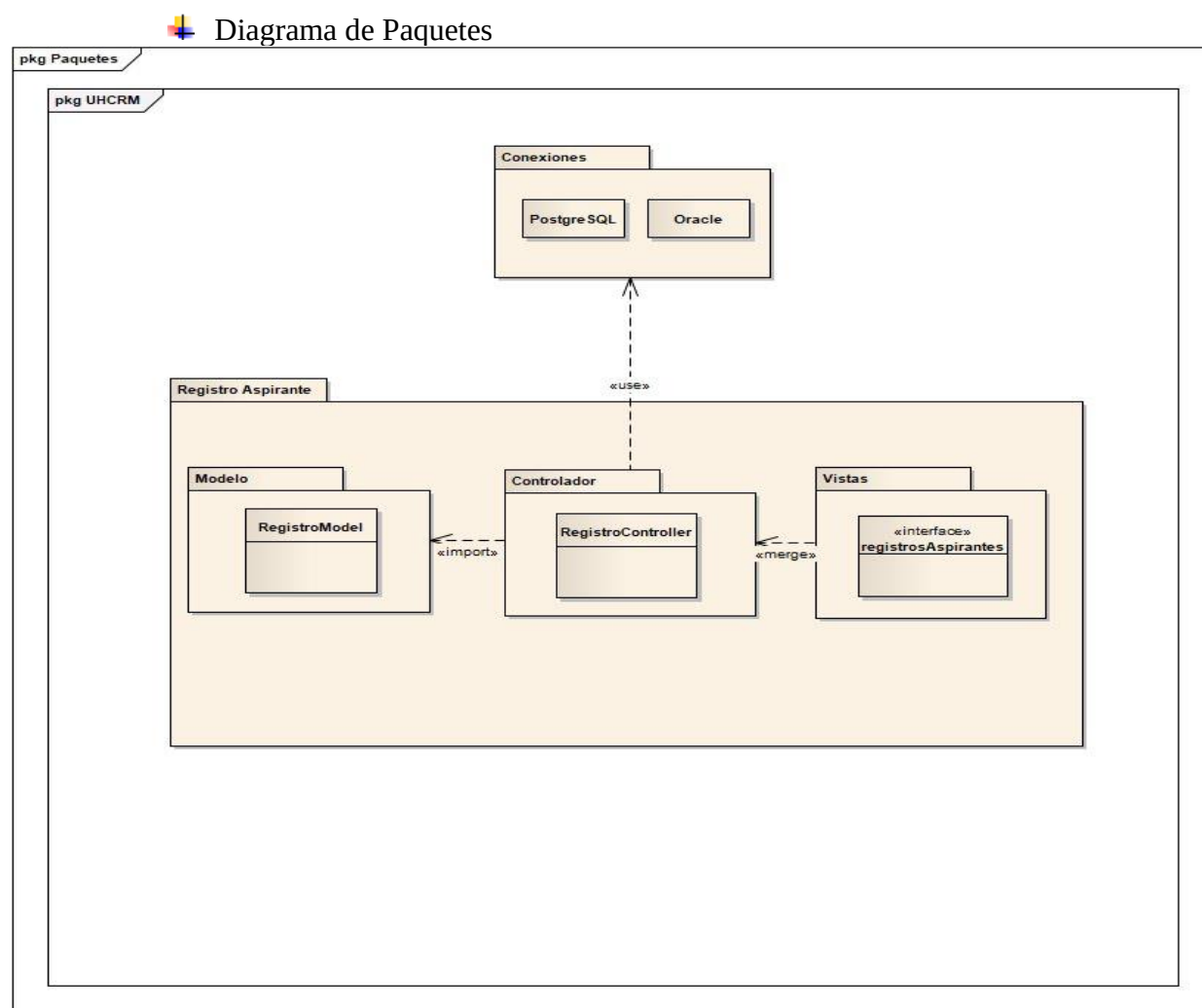


Nota Fuente: Los Autores

Ilustración 20 Diagrama de Comunicación DC02

8.3.1.3. Vista del Desarrollo o (Development View)

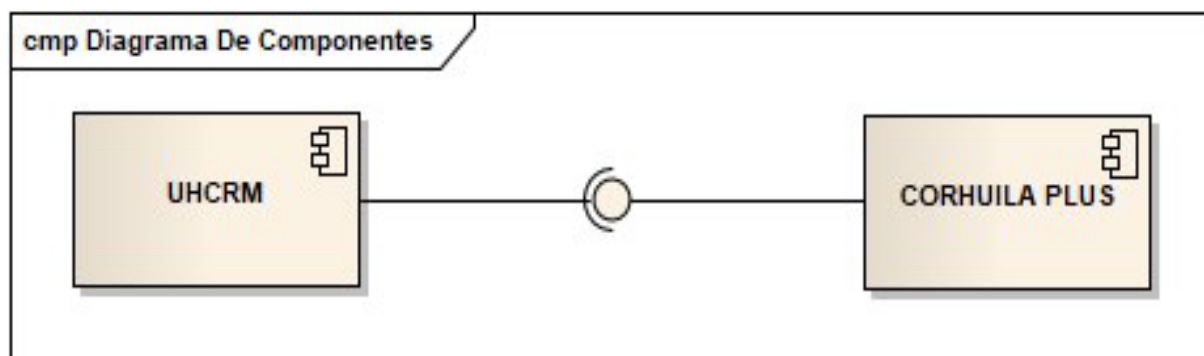
Se centra en la organización del software en el entorno de desarrollo, el cual se encuentra empaquetado pequeñas porciones del programa de bibliotecas, o subsistemas que puede ser desarrollado por uno o más desarrolladores. Los subsistemas están organizados en una jerarquía de capas, cada capa proporciona un estrecho y bien definido de interfaz a las capas superiores, el sistema es representado por módulo y diagramas del subsistema, mostrando la "exportación" e "importación" de relaciones. (Krunten, 1995).



Nota Fuente: Los Autores

Ilustración 21 Diagrama de Paquetes

Diagrama de Componentes



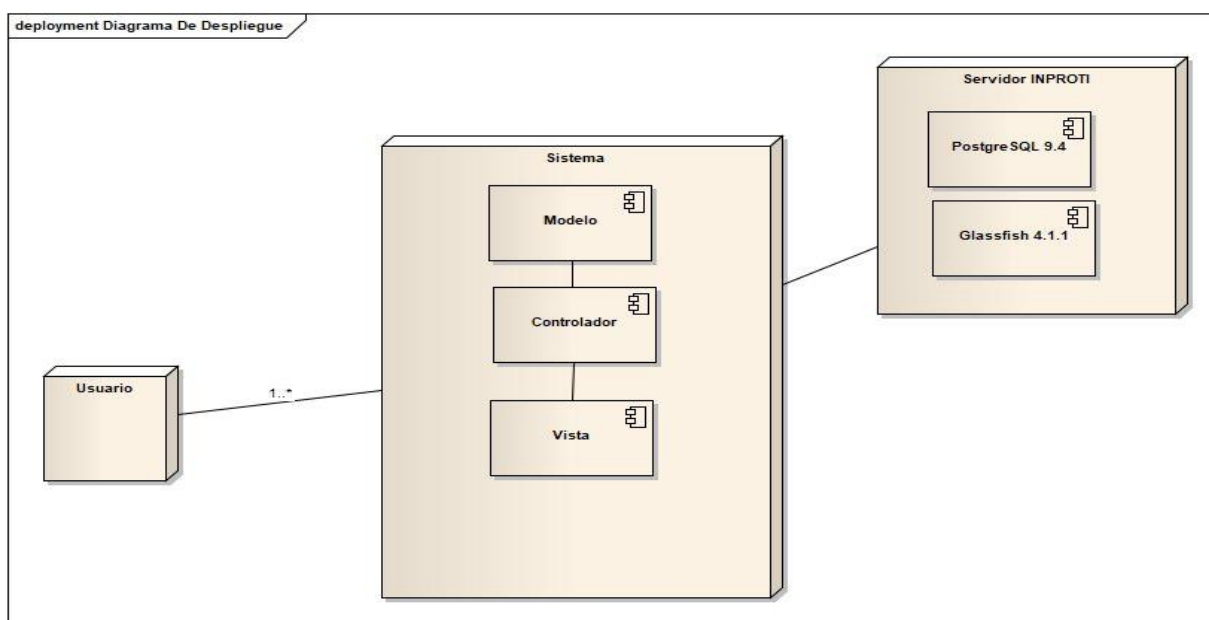
Nota Fuente: Los Autores

Ilustración 22 Diagrama de Componentes

8.3.1.4. Vista Física o (Physical View)

Asignación del software para el hardware de la arquitectura física tiene en cuenta principalmente los requisitos no funcionales del sistema, como la disponibilidad, la fiabilidad (tolerancia a fallos), Performance (Rendimiento) y escalabilidad (Kruntschen, 1995)

Diagrama de Despliegue

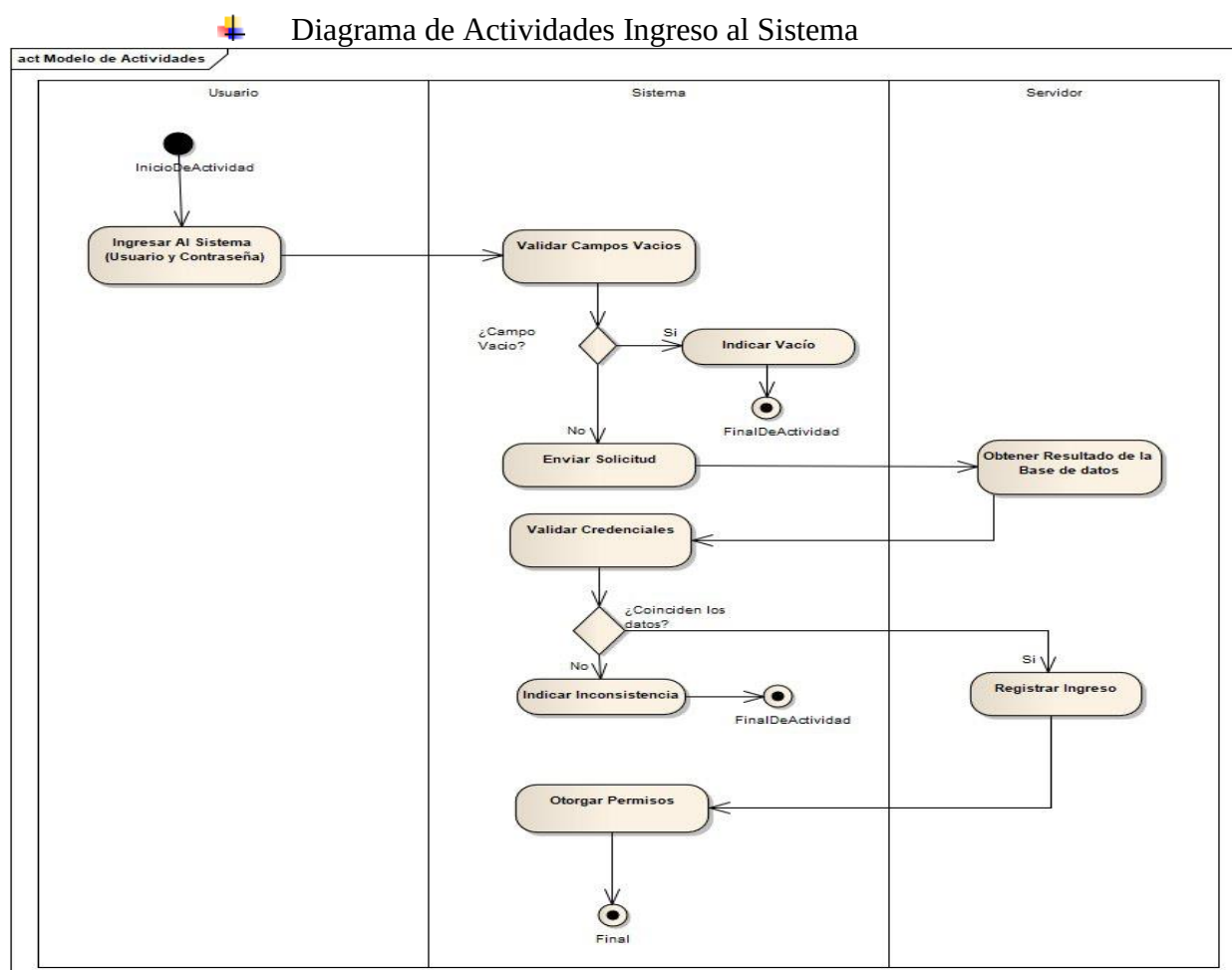


Nota Fuente: Los Autores

Ilustración 23 Diagrama de Despliegue

8.3.1.5. Vista de procesos o (Process View)

Se toman algunos requisitos no funcionales, como el rendimiento y disponibilidad, que permiten solucionar problemas de concurrencia y distribución, de la integridad del sistema como tolerancia a fallos, pueden ser vistos como un conjunto independiente de redes lógicas de programas de comunicación (denominados "procesos"), distribuidos en un conjunto de hardware, los procesos pueden ser replicados para el aumento de la distribución de la carga de procesamiento, o para mejorar la disponibilidad. El software está dividido en un conjunto de tareas independientes (Kruntschen, 1995).

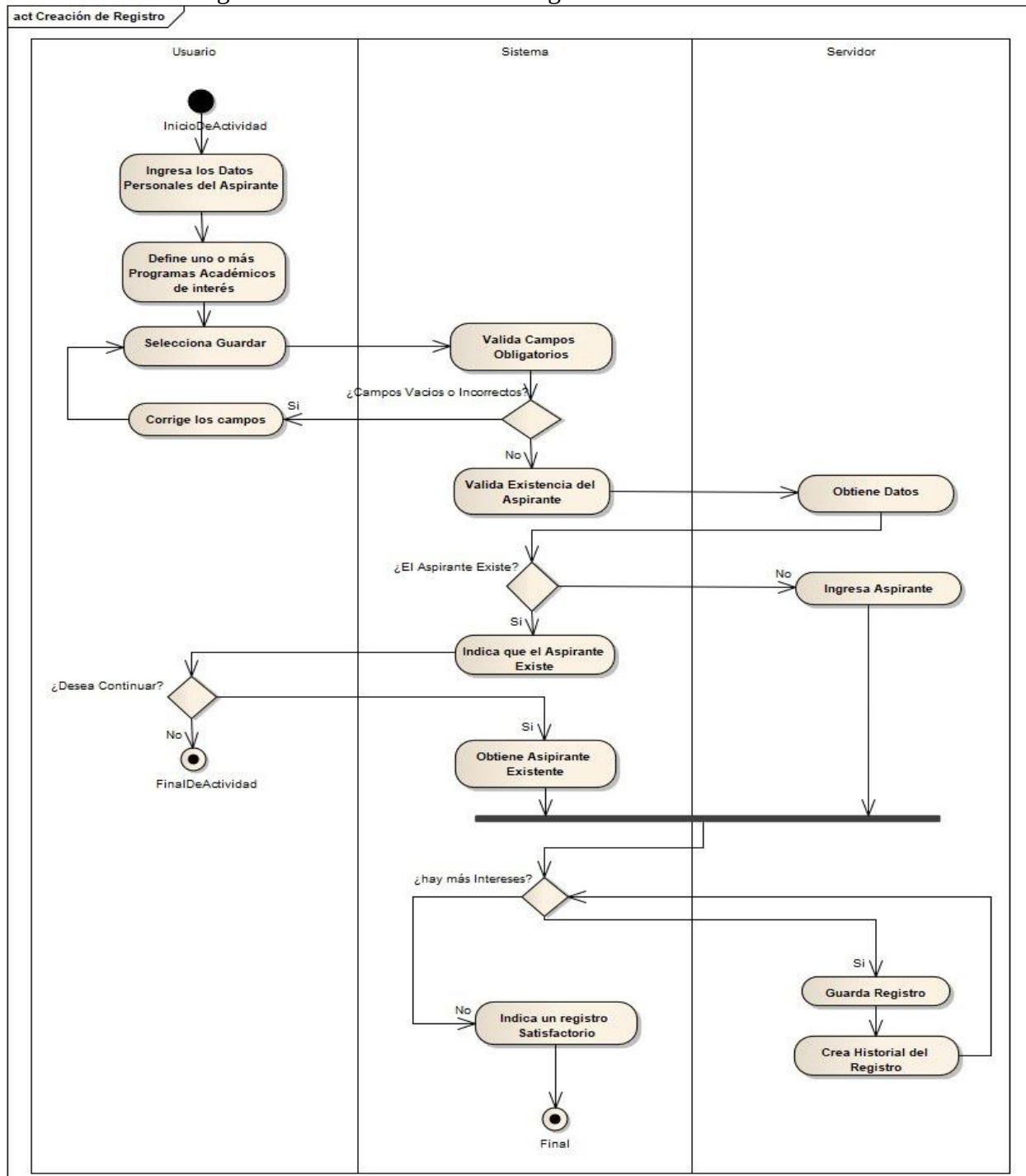


Nota Fuente: Los Autores

Ilustración 24 Diagrama de Actividades DA01



Diagrama de Actividades Crear Registro

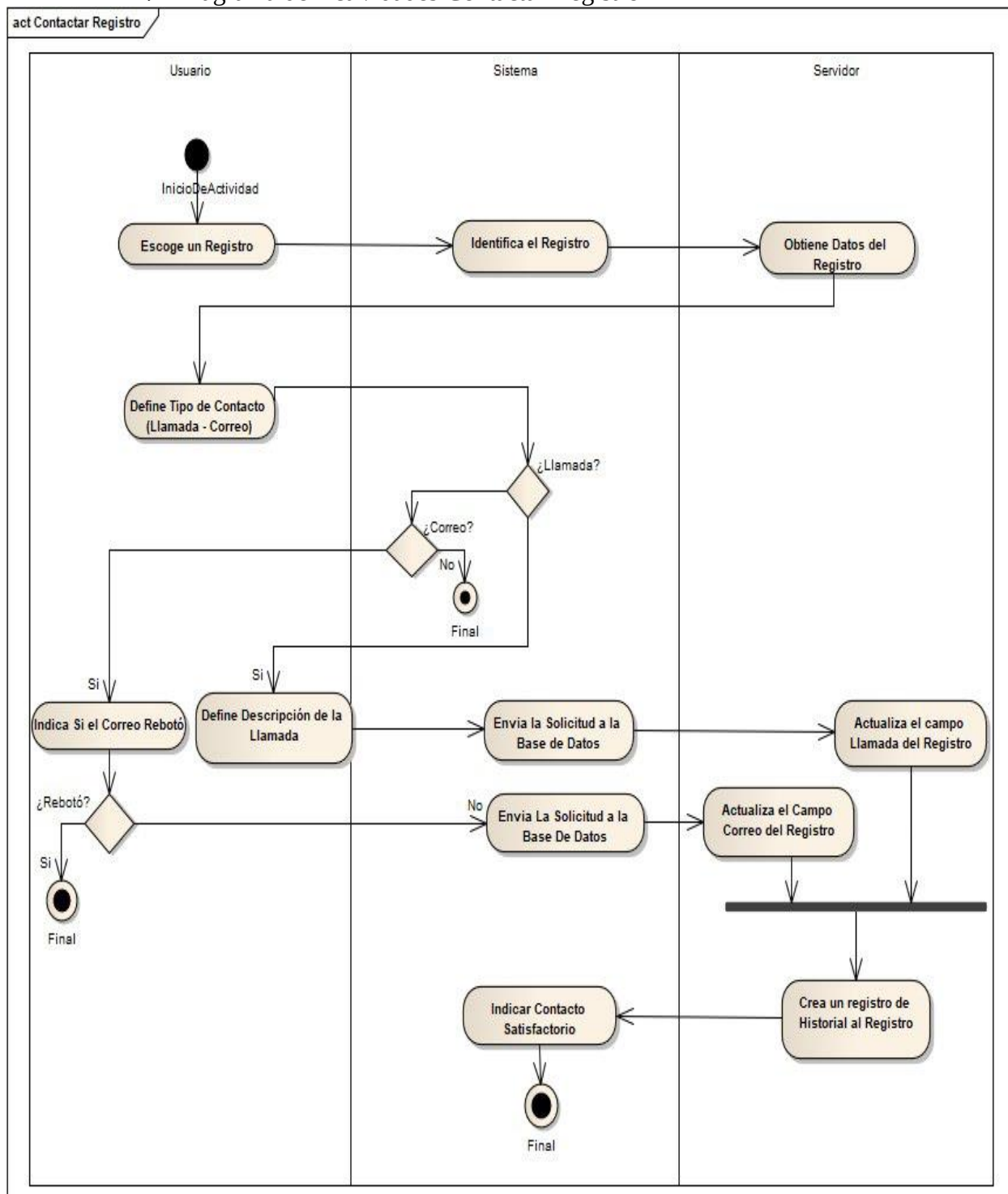


Nota Fuente: Los Autores

Ilustración 25 Diagrama de Actividades DA02



Diagrama de Actividades Contactar Registro



Nota Fuente: Los Autores

Ilustración 26 Diagrama de Actividades DA03

8.3.2. Modelo Relacional de la Base de Datos

Representa la realidad con un número de elementos semánticos (Taboada González & Cotos Yáñez, 2005), que consiste en un conjunto llamado entidades y de unas relaciones entre estos objetos, se usa para esquematizar la estructura lógica de la base de datos (Rivera, 2008).

Entidad:

Es una cosa u objeto del cual se quiere conocer o almacenar información, las entidades se representan mediante cajas, en la cuales se coloca su nombre que va de forma singular y en mayúscula, el tamaño de las cajas debe poseer el espacio necesario dispuesto hacer un diagrama legible permitiendo que las líneas de relación se crucen (Rocha, 2017).

Relación:

Una relación se representa mediante una línea la cual es una asociación significativa entre dos entidades, cada entidad debe ser identificable de forma separada, o consigo misma.

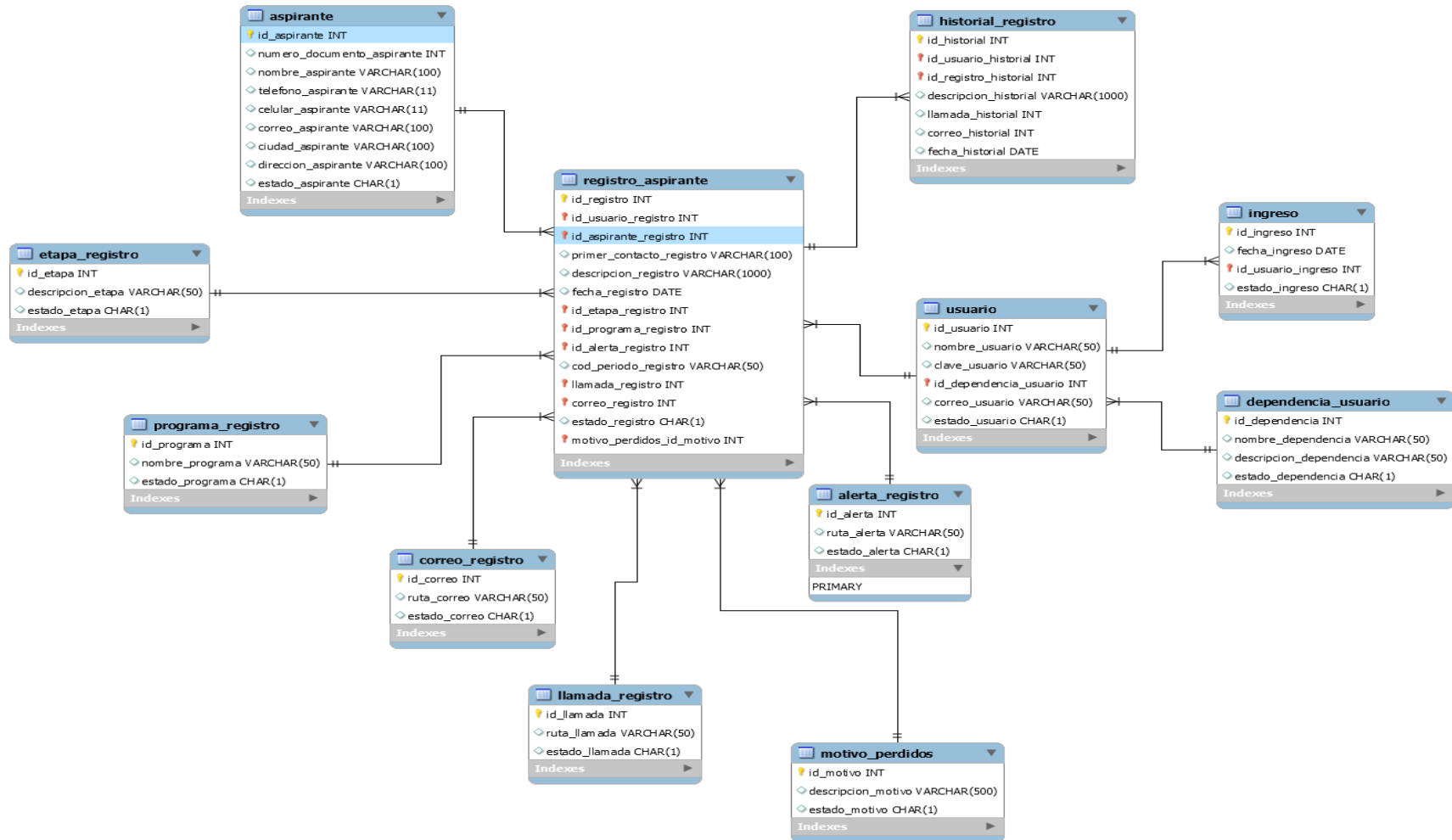
- Una leyenda
- Cardinalidad uno o muchos
- Una opcionalidad o obligatoria

Referencia General: (Rocha, 2017)

Atributo:

Un atributo puede ser un texto descriptivo que dé a entender su función para representar un atributo este se escribe en minúscula y singular, un atributo se transforma en una entidad cuando tiene significado completo en sí mismo con relaciones y atributos propios (Rocha, 2017).

Modelo Entidad Relación



Nota Fuente: Los Autores

Ilustración 27 Modelo Entidad Relación

8.3.3. Modelo Vista Controlador (MVC)

Es una guía para el diseño de arquitecturas que ofrecen una alta interacción con los usuarios, su funcionamiento consiste en la separación de la aplicación de forma organizada, en tres partes, primero los datos de la aplicación, segundo las vistas, es decir formularios de entrada y salida de la información y el tercero son los controladores que procesan las peticiones del usuario y la ejecución del sistema (Gutiérrez, 2006).

Modelo:

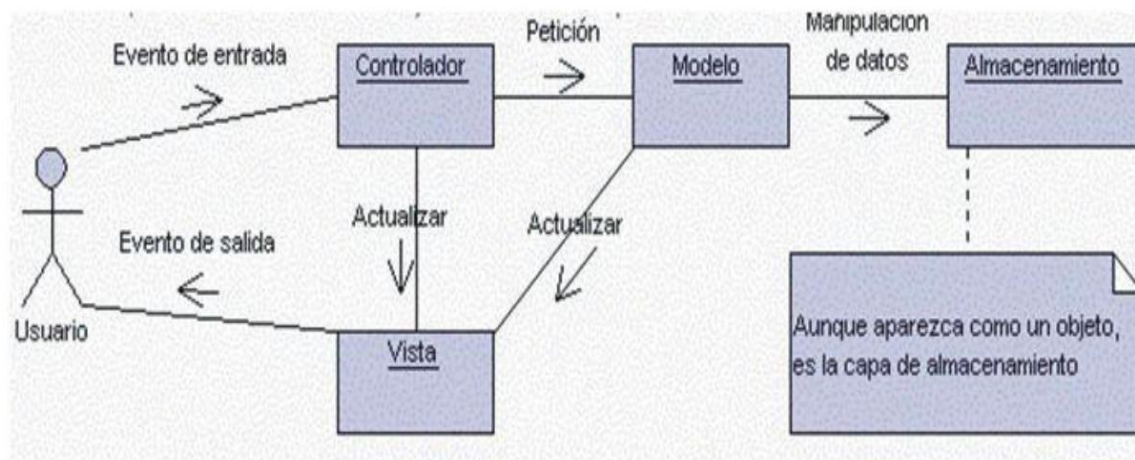
Representa los datos del programa, no tiene conocimiento del controlador ni de las vistas, su función principal es mantener el enlace entre las mismas, notificar cuando el modelo es actualizado (Fernández Romero & Díaz Gonzáles, 2012).

Vista:

Maneja la presentación de los datos por el modelo, y da una vista a los usuarios, su interacción es con el controlador, pero es posible que su comunicación sea a través de una referencia del modelo (Fernández Romero & Díaz Gonzáles, 2012).

Controlador:

Es el quien da significado a las órdenes del usuario, actuando sobre la información presentada por el modelo. su interacción principal es entre la vista y el modelo en cuanto se realizan actualizaciones en el modelo o la vista (Fernández Romero & Díaz Gonzáles, 2012).



Nota Fuente: Y. Fernández Romero and Y. Díaz Gonzáles, “Patrón Modelo-Vista-Controlador,” *Rev. Telemática*, vol. 11, no. 1, pp. 1–11, 2012.

Ilustración 28 Comunicación del Patrón (MVC)

De acuerdo con lo anterior se expone el empleo del modelo MVC en el aplicativo web de la siguiente manera:

Modelo:

```

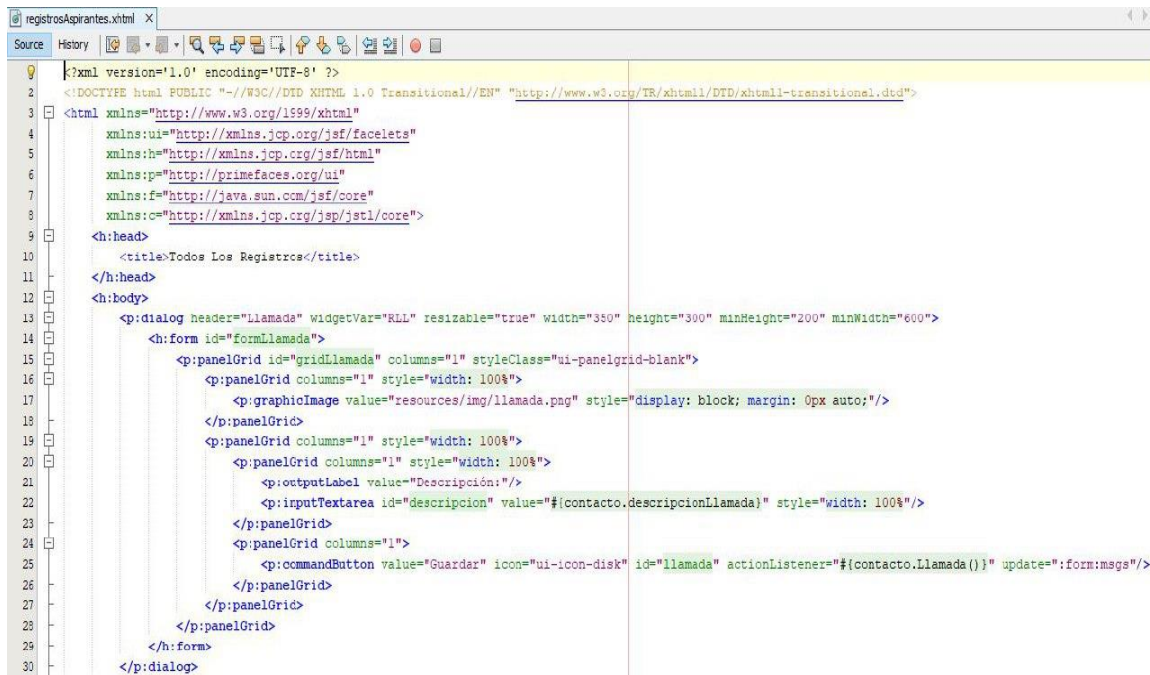
RegistroAspiranteModel.java
Sources History
6 public class RegistroAspiranteModel implements Serializable{
7     private int idRegistro;
8     private String numDocAspiranteRegistro;
9     private String nombreAspirante;
10    private String contacto;
11    private String nombrePrograma;
12    private String telefono;
13    private String celular;
14    private String email;
15    private String descripcionRegistro;
16    private String fecha;
17    private String etapa;
18    private String estado;
19    private String ruta;
20    private String llamada;
21    private String correo;
22    private String motivo;
23
24    public int getIdRegistro() {
25        return idRegistro;
26    }
27
28    public void setIdRegistro(int idRegistro) {
29        this.idRegistro = idRegistro;
30    }
31
32    public String getNumDocAspiranteRegistro() {
33        return numDocAspiranteRegistro;
34    }
35
36    public void setNumDocAspiranteRegistro(String numDocAspiranteRegistro) {
37        this.numDocAspiranteRegistro = numDocAspiranteRegistro;
38    }

```

Nota Fuente: Los Autores

Ilustración 29 Patrón (MVC) Aplicación - Modelo

Vista:



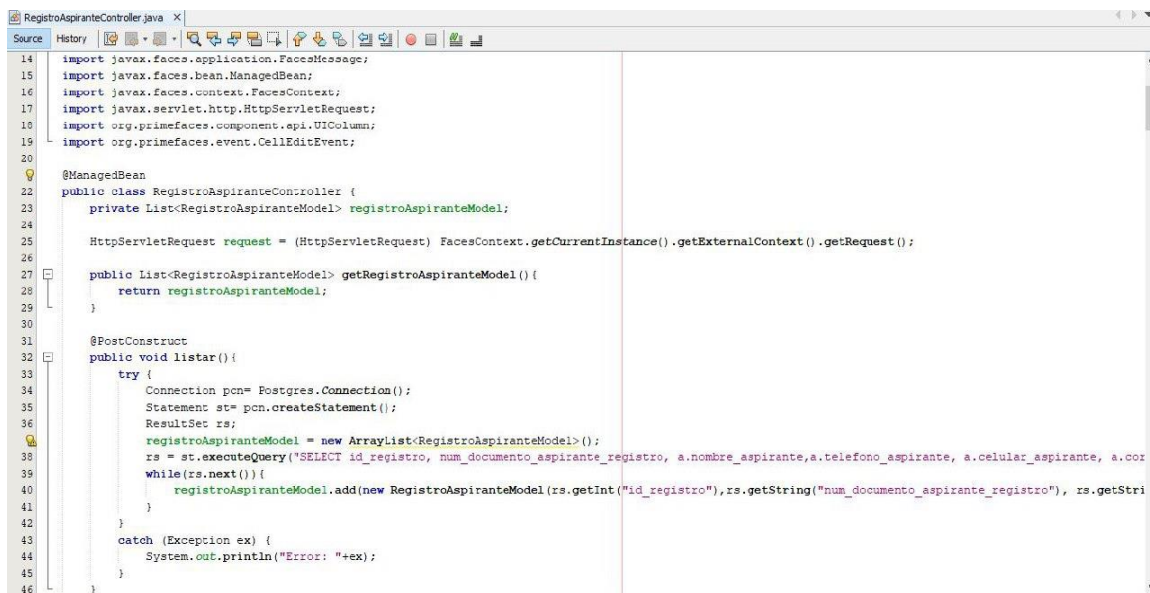
```

1 <?xml version='1.0' encoding='UTF-8' ?>
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml"
4       xmlns:ui="http://xmlns.jcp.org/jsf/facelets"
5       xmlns:h="http://xmlns.jcp.org/jsf/html"
6       xmlns:p="http://primefaces.org/ui"
7       xmlns:f="http://java.sun.com/jsf/core"
8       xmlns:c="http://xmlns.jcp.org/jsp/jstl/core">
9   <h:head>
10    <title>Todos Los Registros</title>
11  </h:head>
12  <h:body>
13    <p:dialog header="Llamada" widgetVar="RLL" resizable="true" width="350" height="300" minHeight="200" minWidth="600">
14      <h:form id="formLlamada">
15        <p:panelGrid id="gridLlamada" columns="1" styleClass="ui-panelgrid-blank">
16          <p:panelGrid columns="1" style="width: 100%">
17            <p:graphicImage value="resources/img/llamada.png" style="display: block; margin: 0px auto;"/>
18          </p:panelGrid>
19          <p:panelGrid columns="1" style="width: 100%">
20            <p:panelGrid columns="1" style="width: 100%">
21              <p:outputLabel value="Descripción:"/>
22              <p:inputText id="descripcion" value="#{contacto.descripcionLlamada}" style="width: 100%"/>
23            </p:panelGrid>
24            <p:panelGrid columns="1">
25              <p:commandButton value="Guardar" icon="ui-icon-disk" id="llamada" actionListener="#{contacto.Llamada()}" update=":form:mags"/>
26            </p:panelGrid>
27          </p:panelGrid>
28        </p:panelGrid>
29      </h:form>
30    </p:dialog>
  
```

Nota Fuente: Los Autores

Ilustración 30 Patrón (MVC) Aplicación - Vista

Controlador:



```

14 import javax.faces.application.FacesMessage;
15 import javax.faces.bean.ManagedBean;
16 import javax.faces.context.FacesContext;
17 import javax.servlet.http.HttpServletRequest;
18 import org.primefaces.component.api.UIColumn;
19 import org.primefaces.event.CellEditEvent;
20
21 @ManagedBean
22 public class RegistroAspiranteController {
23     private List<RegistroAspiranteModel> registroAspiranteModel;
24
25     HttpServletRequest request = (HttpServletRequest) FacesContext.getCurrentInstance().getExternalContext().getRequest();
26
27     public List<RegistroAspiranteModel> getRegistroAspiranteModel() {
28         return registroAspiranteModel;
29     }
30
31     @PostConstruct
32     public void listar() {
33         try {
34             Connection con= Postgres.Connection();
35             Statement st= con.createStatement();
36             ResultSet rs;
37             registroAspiranteModel = new ArrayList<RegistroAspiranteModel>();
38             rs = st.executeQuery("SELECT id_registro, num_documento_aspirante_registro, a.nombre_aspirante, a.telefono_aspirante, a.celular_aspirante, a.cor");
39             while(rs.next()){
40                 registroAspiranteModel.add(new RegistroAspiranteModel(rs.getInt("id_registro"),rs.getString("num_documento_aspirante_registro"), rs.getStri
41             }
42         } catch (Exception ex) {
43             System.out.println("Error: "+ex);
44         }
45     }
46 }
  
```

Nota Fuente: Los Autores

Ilustración 31 Patrón (MVC) Aplicación - Controlador

8.4. Fase de Desarrollo

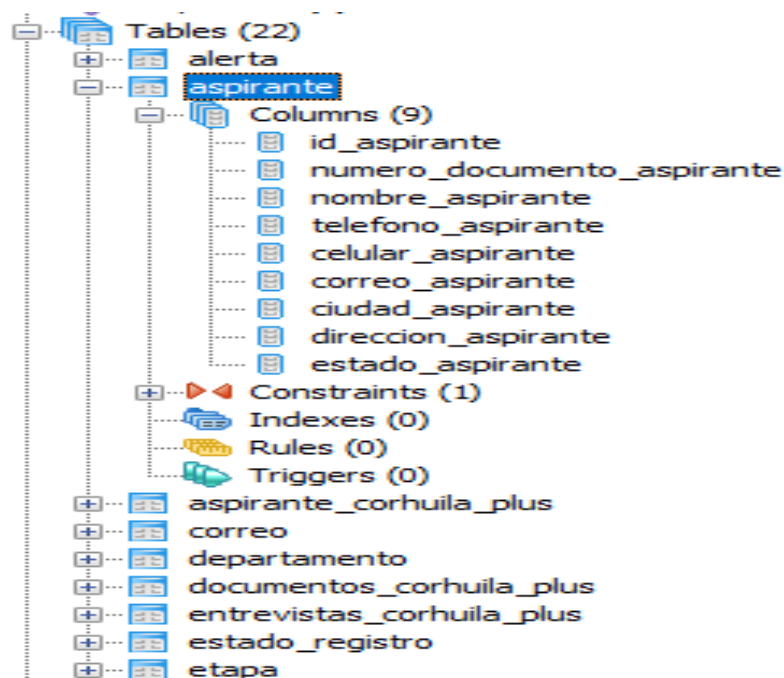
8.4.1. Estándares

Son parte de las llamadas buenas o mejores prácticas, son un conjunto de reglas no formales que distintas comunidades han empleado en sus codificaciones para una mejor lectura de código, y que pueden ser aplicadas a diferentes metodologías de desarrollo que con el tiempo y bien aplicadas incrementan la calidad del código. (Programming guidelines, 2017)

✚ Estándar utilizado en de la Base de Datos:

- ◆ El nombre de las tablas en minúscula (lowercase)
- ◆ Los campos de la tabla se escriben en minúscula (lowercase), seguido de un guion bajo y el nombre de la tabla a la que pertenece.

✚ Aplicación del estándar a la Base de Datos:



Nota Fuente: Los Autores

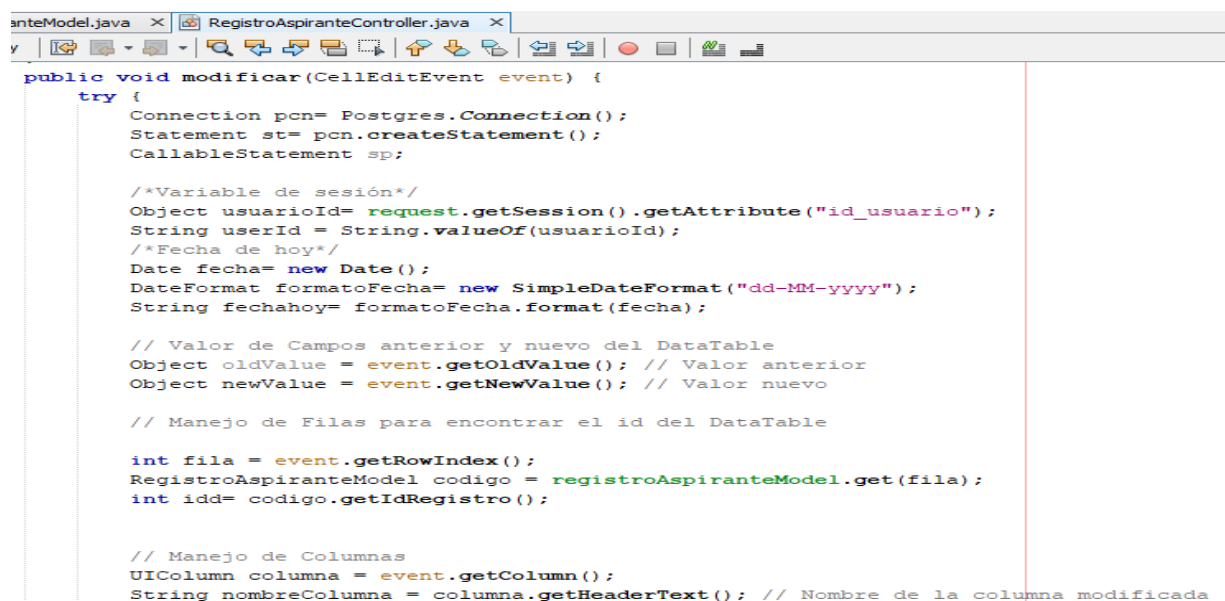
Ilustración 32 Estándar Empleado en la Base de Datos

▪ Estándar utilizado en la Codificación

Los identificadores como PascalCase se utilizan en clases, propiedades, métodos y otros miembros de clase. Al igual que en el nombre, PascalCase, el identificador de cada palabra es mayúscula, incluida la primera. El identificador Camelcase es para los campos privados, parámetros y variables. El formato Camelcase pone en mayúscula la primera palabra y los posteriores en minúscula (Mayo, 2017).

De acuerdo con lo anterior se especifica los estándares utilizados en la codificación de la aplicación:

- ✚ El nombre de los paquetes en formato PascalCase
- ✚ Las clases deben estar en formato PascalCase.
- ✚ Las variables estarán definidas en camelCase.
- ✚ El código debe estar debidamente comentado para seguir el funcionamiento de esta por parte de otro compañero.



```

public void modificar(CellEditEvent event) {
    try {
        Connection pcn= Postgres.Connection();
        Statement st= pcn.createStatement();
        CallableStatement sp;

        /*Variable de sesión*/
        Object usuarioId= request.getSession().getAttribute("id_usuario");
        String userId = String.valueOf(usuarioId);
        /*Fecha de hoy*/
        Date fecha= new Date();
        DateFormat formatoFecha= new SimpleDateFormat("dd-MM-yyyy");
        String fechahoy= formatoFecha.format(fecha);

        // Valor de Campos anterior y nuevo del DataTable
        Object oldValue = event.getOldValue(); // Valor anterior
        Object newValue = event.getNewValue(); // Valor nuevo

        // Manejo de Filas para encontrar el id del DataTable

        int fila = event.getRowIndex();
        RegistroAspiranteModel codigo = registroAspiranteModel.get(fila);
        int idd= codigo.getIdRegistro();

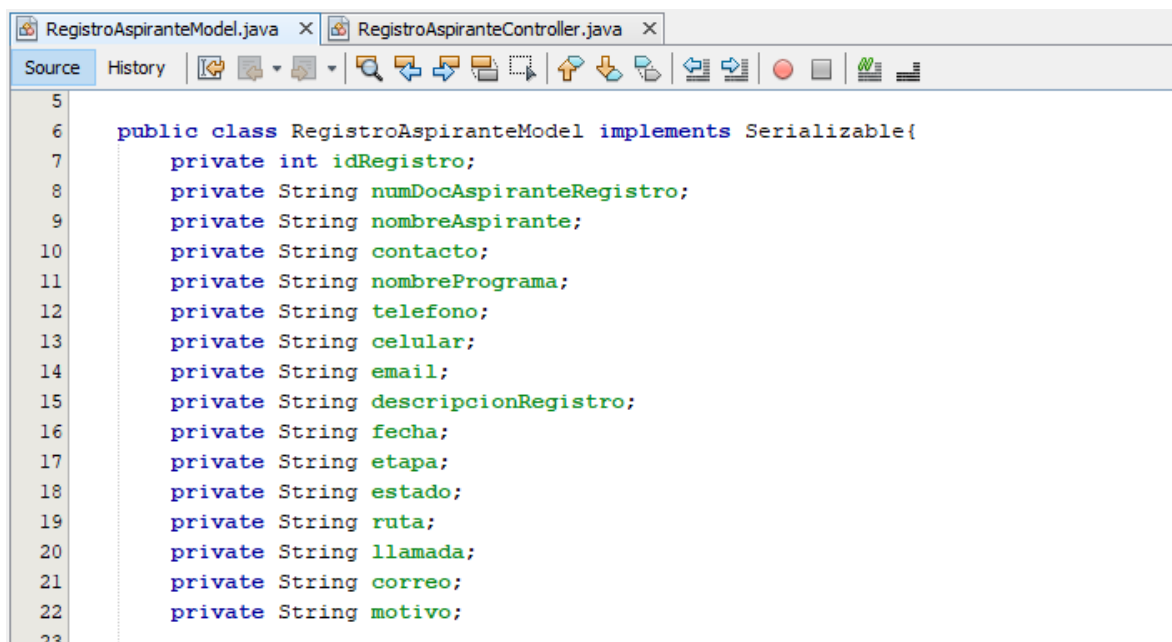
        // Manejo de Columnas
        UIColumn columna = event.getColumn();
        String nombreColumna = columna.getHeaderText(); // Nombre de la columna modificada
    }
}

```

Nota Fuente: Los Autores

Ilustración 33 Estándar Empleado en la Codificación

✚ Estandar aplicado a los paquetes, clases, etc.



Nota Fuente: Los Autores

Ilustración 34 Estándar Aplicado a Clases y Variables

8.4.2. Código Limpio

Los principios aplicados al código de la aplicación web se basaron en tres factores esenciales:

- ✚ Legibilidad de la fuente es decir que sea capaz de ser explicado sin nada más que el código.
- ✚ Organización del elemento, la estructura del código debe ser correcta.
- ✚ Funcionamiento, se obtiene de diferentes pruebas que verifiquen el funcionamiento correcto de la aplicación.

Referencia General: (Gozales, 2015)

Descrito lo anterior se presenta un fragmento de código demostrando los principios de código limpio:

Las variables están definidas globalmente para su uso en cualquier método de la clase.

```

1
2 package RegistrosAspirantes;
3
4 import Conexiones.Postgres;
5 import java.sql.CallableStatement;
6 import java.sql.Connection;
7 import java.sql.Statement;
8 import java.text.DateFormat;
9 import java.text.SimpleDateFormat;
10 import java.util.Date;
11 import javax.faces.application.FacesMessage;
12 import javax.faces.bean.ManagedBean;
13 import javax.faces.context.FacesContext;
14 import javax.servlet.http.HttpServletRequest;
15
16 @ManagedBean
17 public class Registro {
18     HttpServletRequest request = (HttpServletRequest) FacesContext.getCurrentInstance().getExternalContext().getRequest();
19     //Variables Globales
20     private int numeroDocumentoAspiranteRegistro;
21     private String nombreAspirante;
22     private String contacto;
23     private String nombrePrograma;
24     private String descripcionRegistro;
25     private String etapa;
26     private String estado;
27

```

Nota Fuente: Los Autores

Ilustración 35 Código Limpio - Aplicación Variables Globales

Encapsulamiento de las Variables

```

28 //Get & Set
29 public int getNumeroDocumentoAspiranteRegistro() {
30     return numeroDocumentoAspiranteRegistro;
31 }
32
33 public void setNumeroDocumentoAspiranteRegistro(int numeroDocumentoAspiranteRegistro) {
34     this.numeroDocumentoAspiranteRegistro = numeroDocumentoAspiranteRegistro;
35 }
36
37 public String getNombreAspirante() {
38     return nombreAspirante;
39 }
40
41 public void setNombreAspirante(String nombreAspirante) {
42     this.nombreAspirante = nombreAspirante;
43 }
44
45 public String getContacto() {
46     return contacto;
47 }
48
49 public void setContacto(String contacto) {
50     this.contacto = contacto;
51 }
52
53 public String getNombrePrograma() {
54     return nombrePrograma;
55 }
56
57 public void setNombrePrograma(String nombrePrograma) {
58     this.nombrePrograma = nombrePrograma;
59 }

```

Nota Fuente: Los Autores

Ilustración 36 Código Limpio – Aplicación a los Métodos GET y SET

Es un método utilizado para todos los registros y no necesariamente definir o crear un método especializado para indicar la realización de la llamada.

```
//Método para definir llamadas a los registros
public void Llamada(String ref){
    try{
        Connection pcn= Postgres.Connection();
        Statement st= pcn.createStatement();
        Object usuarioId= request.getSession().getAttribute("idUsuario");
        String userId = String.valueOf(usuarioId);
        Date fecha= new Date();
        DateFormat formatoFecha= new SimpleDateFormat("dd-MM-yyyy");
        String fechahoy= formatoFecha.format(fecha);

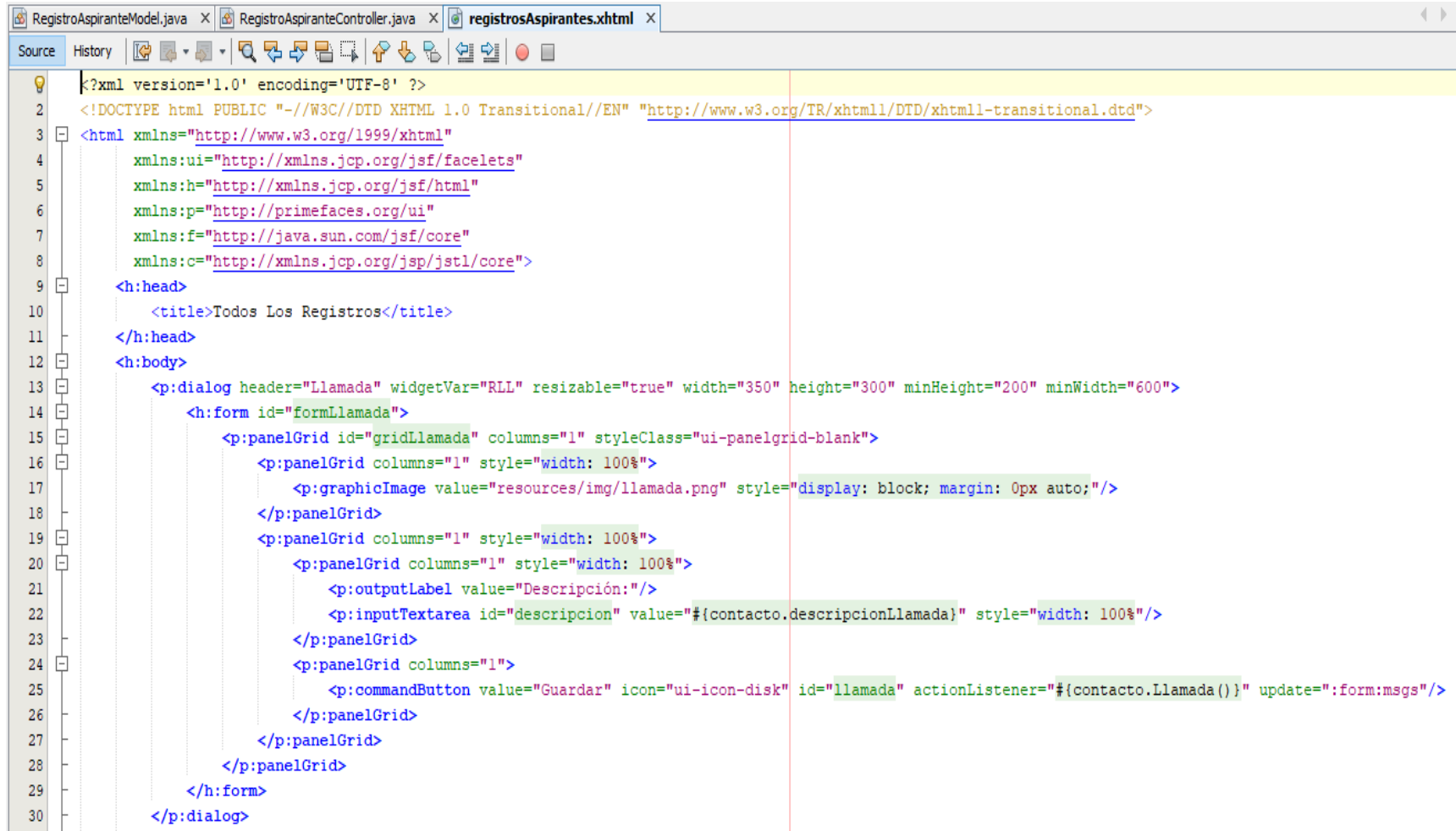
        int resultado= st.executeUpdate("UPDATE registro_aspirante SET descripcion_registro='"+descripcionRegistro+"' WHERE id_registro="+ref+"");
        CallableStatement sp_historial;
        sp_historial= pcn.prepareCall("SELECT sp_historial_registro("+userId+", "+ref+", 'Llamada= -'+descripcionRegistro+'-', '"+fechahoy+"',1,0)");
        if(resultado == 1 && sp_historial.execute()){
            FacesContext.getCurrentInstance().addMessage(null, new FacesMessage(FacesMessage.SEVERITY_INFO, "Se registró la llamada",""));
        }
    }
    catch(Exception ex){
        FacesContext.getCurrentInstance().addMessage(null, new FacesMessage(FacesMessage.SEVERITY_INFO, "Error", "Conexión Perdida"));
    }
}
```

Nota Fuente: Los Autores

Ilustración 37 Código Limpio – Aplicación Método Llamada

Explicación del formulario Registro Aspirante en sus tres formas Vista, Modelo y Controlador.

La vista interactuará con el usuario final, con la intención de suministrar todos los servicios que sean solicitados.



```

1  <?xml version='1.0' encoding='UTF-8' ?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3  <html xmlns="http://www.w3.org/1999/xhtml"
4        xmlns:ui="http://xmlns.jcp.org/jsf/facelets"
5        xmlns:h="http://xmlns.jcp.org/jsf/html"
6        xmlns:p="http://primefaces.org/ui"
7        xmlns:f="http://java.sun.com/jsf/core"
8        xmlns:c="http://xmlns.jcp.org/jsp/jstl/core">
9  <h:head>
10     <title>Todos Los Registros</title>
11 </h:head>
12 <h:body>
13     <p:dialog header="Llamada" widgetVar="RLI" resizable="true" width="350" height="300" minHeight="200" minWidth="600">
14         <h:form id="formLlamada">
15             <p:panelGrid id="gridLlamada" columns="1" styleClass="ui-panelgrid-blank">
16                 <p:panelGrid columns="1" style="width: 100%">
17                     <p:graphicImage value="resources/img/llamada.png" style="display: block; margin: 0px auto;"/>
18                 </p:panelGrid>
19                 <p:panelGrid columns="1" style="width: 100%">
20                     <p:panelGrid columns="1" style="width: 100%">
21                         <p:outputLabel value="Descripción:"/>
22                         <p:inputTextarea id="descripcion" value="#{contacto.descripcionLlamada}" style="width: 100%"/>
23                     </p:panelGrid>
24                     <p:panelGrid columns="1">
25                         <p:commandButton value="Guardar" icon="ui-icon-disk" id="llamada" actionListener="#{contacto.Llamada()}" update=":form:msgs"/>
26                     </p:panelGrid>
27                 </p:panelGrid>
28             </p:panelGrid>
29         </h:form>
30     </p:dialog>

```

Nota Fuente: Los Autores

Ilustración 38 Vista Registro Aspirante Parte 1

- Hace el uso de diálogos para realizar funciones ajenas de la vista pero que se relacionan con los registros para así mantener la fluidez de trabajo al evitar el uso de otra vista para dicha función.

```

55 <c:if test="#{sessionScope.id_usuario != null }">
56   <p:panel header="Todos Los Registros" style="margin-bottom:1px;">
57     <h:form id="form">
58       <p:panelGrid id="grid" columns="2" styleClass="ui-panelgrid-blank">
59         <f:facet name="header">
60           <p:messages id="msgs" />
61         </f:facet>
62         <p:dataTable id="regAsp" var="aspirante" style="table-layout: auto" value="#{registroAspiranteController.registroAspiranteModel}" editable="true">
63           <p:ajax event="cellEdit" listener="#{registroAspiranteController.modificar}" update=":form:msgs"/>
64           <f:facet name="header">
65             <p:commandButton id="toggler" type="button" value="Ver" style="float:right;width:8%" icon="ui-icon-calculator"/>
66             <p:columnToggler datasources="regAsp" trigger="toggler"/>
67           </f:facet>
68           <p:column headerText="Historial" style="width: 7%; padding-left: 3px; padding-right: 1px">
69             <h:commandLink target="_blank" id="buscar" value="#{aspirante.idRegistro}" actionListener="#{registroAspiranteController.historial(aspirante.idRegistro)}"/>
70           </p:column>
71           <p:column headerText="Contacto" style="width: 7%; padding-left: 3px; padding-right: 1px">
72             <p:commandButton icon="#{aspirante.llamada}" title="Llamada" style="width: 45%">
73               <p:ajax listener="#{registroAspiranteController.Llamada(aspirante.idRegistro)}" update=":form:grid"/>
74             </p:commandButton>
75             <p:commandButton icon="#{aspirante.correo}" title="Correo" style="width: 45%">
76               <p:ajax listener="#{registroAspiranteController.Correo(aspirante.idRegistro)}" update=":form:grid"/>
77             </p:commandButton>
78           </p:column>
79           <p:column headerText="Inscripción" style="width: 10%; padding-left:1px">
80             <h:outputText value="#{aspirante.fecha}"/>
81           </p:column>
82           <p:column headerText="Documento" style="width: 12%">
83             <h:outputText value="#{aspirante.numDocAspiranteRegistro}"/>
84           </p:column>
85           <p:column headerText="Nombre" style="width: 20%">
86             <h:outputText value="#{aspirante.nombreAspirante}"/>
87           </p:column>

```

Nota Fuente: Los Autores

Ilustración 39 Vista Registro Aspirante Parte 2

```

103 <p:column headerText="Contacto" style="width: 26%" >
104   <h:outputText value="#{aspirante.email}"/>
105   <br></br>
106   <h:outputText value="#{aspirante.telefono}"/>
107   <br></br>
108   <h:outputText value="#{aspirante.celular}"/>
109 </p:column>
110 <p:column headerText="Interés" style="width: 10%">
111   <p:cellEditor>
112     <f:facet name="output">
113       <h:outputText value="#{aspirante.nombrePrograma}"/>
114     </f:facet>
115     <f:facet name="input">
116       <h:selectOneMenu id="interes" value="#{aspirante.etapa}" required="true" requiredMessage="Programa Null">
117         <f:selectItem itemLabel="- Seleccione -" itemValue="" noSelectionOption="true" />
118         <f:selectItem itemLabel="Adm. de empresas agropecuarias." itemValue="10" />
119         <f:selectItem itemLabel="Adm. de empresas turísticas." itemValue="15" />
120         <f:selectItem itemLabel="Adm. Bancaria y financiera" itemValue="30" />
121         <f:selectItem itemLabel="Adm. Comercial." itemValue="60" />
122         <f:selectItem itemLabel="Ing. Ambiental." itemValue="80" />
123         <f:selectItem itemLabel="Ing. Industrial" itemValue="40" />
124         <f:selectItem itemLabel="Ing. Sistemas" itemValue="90" />
125         <f:selectItem itemLabel="Ing. Mecatrónica" itemValue="17" />
126         <f:selectItem itemLabel="Mercadeo Publicidad y ventas" itemValue="70" />
127         <f:selectItem itemLabel="Negocios internacionales." itemValue="12" />
128         <f:selectItem itemLabel="Veterinaria y Zootecnia" itemValue="50" />
129         <f:selectItem itemLabel="Diplomado" itemValue="200" />
130         <f:selectItem itemLabel="Taller" itemValue="300" />
131         <f:selectItem itemLabel="Curso" itemValue="400" />
132         <f:selectItem itemLabel="Seminario" itemValue="500" />
133       </h:selectOneMenu>
134     </f:facet>
135   </p:cellEditor>

```

Nota Fuente: Los Autores

Ilustración 40 Vista Registro Aspirante Parte 3

```

137 <p:column headerText="Etapa" style="width: 10%" >
138   <p:cellEditor>
139     <f:facet name="output">
140       <h:outputText value="#{aspirante.etapa}" />
141     </f:facet>
142     <f:facet name="input">
143       <h:selectOneMenu id="etapa" value="#{aspirante.etapa}" required="true" requiredMessage="Etapa Null">
144         <f:selectItem itemLabel="- Seleccione -" itemValue="" noSelectionOption="true" />
145         <f:selectItem itemLabel="Pre-Inscrito Sin Pago" itemValue="1" />
146         <f:selectItem itemLabel="Pre-Inscrito Con Pago" itemValue="2" />
147         <f:selectItem itemLabel="Inscrito" itemValue="3" />
148         <f:selectItem itemLabel="Documentación" itemValue="4" />
149         <f:selectItem itemLabel="Entrevista" itemValue="5" />
150         <f:selectItem itemLabel="Admitido" itemValue="6" />
151         <f:selectItem itemLabel="Matricula Académica" itemValue="7" />
152         <f:selectItem itemLabel="Matricula Financiera" itemValue="8" />
153         <f:selectItem itemLabel="Inducción" itemValue="9" />
154         <f:selectItem itemLabel="Finalizado" itemValue="10" />
155       </h:selectOneMenu>
156     </f:facet>
157   </p:cellEditor>
158 </p:column>
159 <p:column headerText="Alerta" style="width: 5%; padding-left: 1px">
160   <p:graphicImage value="#{aspirante.ruta}" style="display: block; margin: 0px auto;"/>
161 </p:column>

```

Nota Fuente: Los Autores

Ilustración 41 Vista Registro Aspirante Parte 4

```

162 <p:column headerText="Inactivar" style="width: 9%" >
163   <p:cellEditor>
164     <f:facet name="output">
165       <h:outputText value="#{aspirante.motivo}" />
166     </f:facet>
167     <f:facet name="input">
168       <h:selectOneMenu id="motivo" value="#{aspirante.motivo}" required="true" requiredMessage="Estado Null">
169         <f:selectItem itemLabel="Seleccione" itemValue='0' />
170         <f:selectItem itemLabel="No tiene Correo/ Teléfono" itemValue="1" />
171         <f:selectItem itemLabel="Perdida de interés" itemValue="2" />
172         <f:selectItem itemLabel="Salud" itemValue="3" />
173         <f:selectItem itemLabel="Carácter económico" itemValue="4" />
174         <f:selectItem itemLabel="Familiares" itemValue="5" />
175         <f:selectItem itemLabel="Otra universidad" itemValue="6" />
176         <f:selectItem itemLabel="Vive lejos" itemValue="7" />
177         <f:selectItem itemLabel="Busca Carrera a Distancia" itemValue="8" />
178         <f:selectItem itemLabel="Laboral" itemValue="9" />
179         <f:selectItem itemLabel="No existe el Programa" itemValue="10" />
180         <f:selectItem itemLabel="Fuera de servicio" itemValue="11" />
181         <f:selectItem itemLabel="Numero equivocado" itemValue="12" />
182         <f:selectItem itemLabel="Sin motivo" itemValue="13" />
183         <f:selectItem itemLabel="Deceso" itemValue="14" />
184       </h:selectOneMenu>
185     </f:facet>
186   </p:cellEditor>
187 </p:column>
188 </p:dataTable>
189 </p:panelGrid>
190 </h:form>
191 </p:panel>
192 </c:if>

```

Nota Fuente: Los Autores

Ilustración 42 Vista Registro Aspirante Parte 5

La vista tiene la capacidad de permitir o no la visualización de su contenido al utilizar una sentencia donde se otorga permisos según la variable de sesión que haya sido asignada en el ingreso del usuario, si cumple con los permisos, será mostrada la tabla con las respectivas columnas que contendrá los datos de los registros.

✚ El **Controlador** se encargará de gestionar las acciones que sean necesarias o solicitadas por el usuario a través de la interface gráfica, pero es necesario que se haga uso del componente **@ManagedBean** para la comunicación entre la vista y el controlador.

```
package RegistrosAspirantes;

import Conexiones.Postgres;
import java.sql.CallableStatement;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.Statement;
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
import java.util.List;
import javax.annotation.PostConstruct;
import javax.faces.application.FacesMessage;
import javax.faces.bean.ManagedBean;
import javax.faces.context.FacesContext;
import javax.servlet.http.HttpServletRequest;
import org.primefaces.component.api.UIColumn;
import org.primefaces.event.CellEditEvent;

@ManagedBean
public class RegistroAspiranteController {
    private List<RegistroAspiranteModel> registroAspiranteModel;

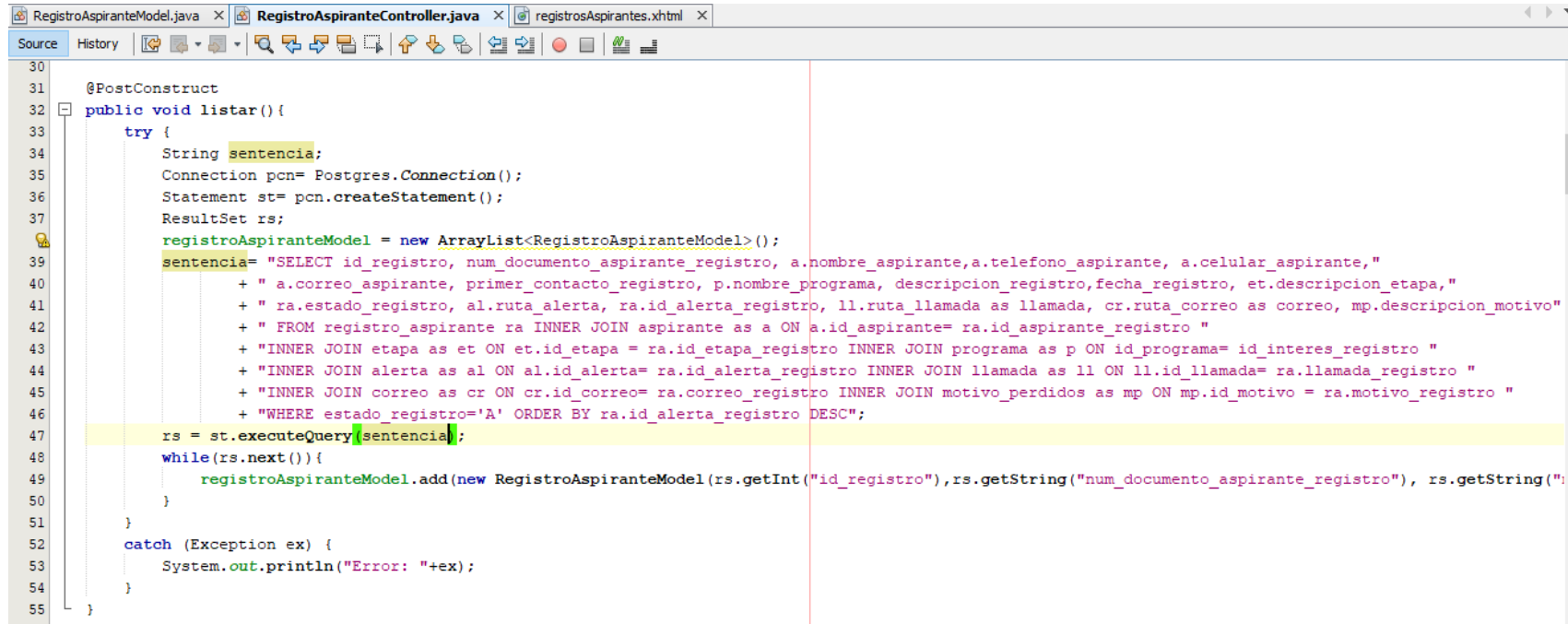
    HttpServletRequest request = (HttpServletRequest) FacesContext.getCurrentInstance().getExternalContext().getRequest();

    public List<RegistroAspiranteModel> getRegistroAspiranteModel() {
        return registroAspiranteModel;
    }
}
```

Nota Fuente: Los Autores

Ilustración 43 Controlador Registro Aspirante Parte 1

✚ Para este ejemplo podemos observar el método utilizado para obtener los registros existentes en la base de datos, los cuales serán mostrados en la vista correspondiente.



```

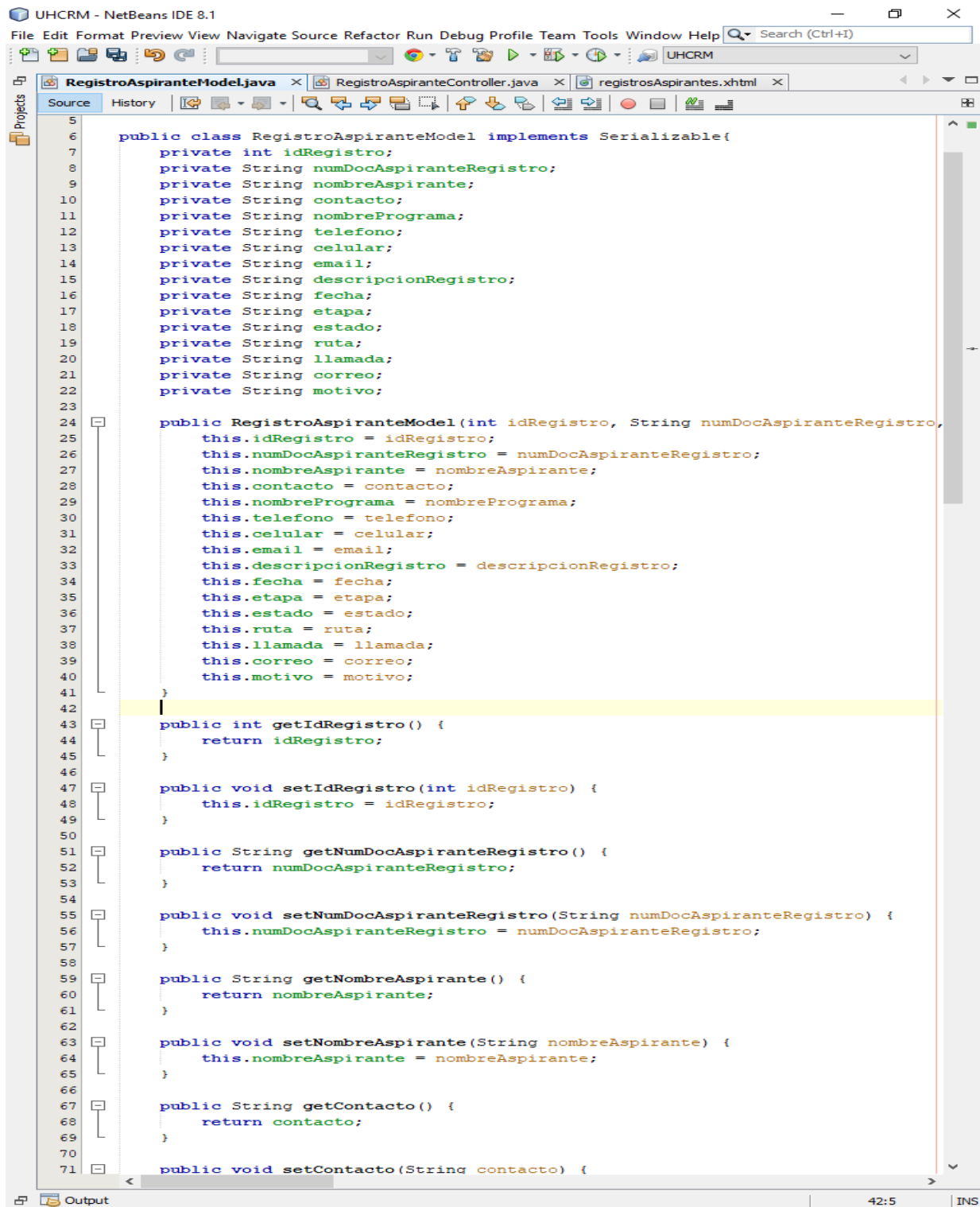
30
31 @PostConstruct
32 public void listar() {
33     try {
34         String sentencia;
35         Connection pcn= Postgres.Connection();
36         Statement st= pcn.createStatement();
37         ResultSet rs;
38         registroAspiranteModel = new ArrayList<RegistroAspiranteModel>();
39         sentencia= "SELECT id_registro, num_documento_aspirante_registro, a.nombre_aspirante, a.telefono_aspirante, a.celular_aspirante, "
40             + " a.correo_aspirante, primer_contacto_registro, p.nombre_programa, descripcion_registro, fecha_registro, et.descripcion_etapa, "
41             + " ra.estado_registro, al.ruta_alerta, ra.id_alerta_registro, ll.ruta_llamada as llamada, cr.ruta_correo as correo, mp.descripcion_motivo"
42             + " FROM registro_aspirante ra INNER JOIN aspirante as a ON a.id_aspirante= ra.id_aspirante_registro "
43             + " INNER JOIN etapa as et ON et.id_etapa = ra.id_etapa_registro INNER JOIN programa as p ON id_programa= id_interes_registro "
44             + " INNER JOIN alerta as al ON al.id_alerta= ra.id_alerta_registro INNER JOIN llamada as ll ON ll.id_llamada= ra.llamada_registro "
45             + " INNER JOIN correo as cr ON cr.id_correo= ra.correo_registro INNER JOIN motivo_perdidos as mp ON mp.id_motivo = ra.motivo_registro "
46             + " WHERE estado_registro='A' ORDER BY ra.id_alerta_registro DESC";
47         rs = st.executeQuery(sentencia);
48         while(rs.next()){
49             registroAspiranteModel.add(new RegistroAspiranteModel(rs.getInt("id_registro"),rs.getString("num_documento_aspirante_registro"), rs.getString("
50         }
51     }
52     catch (Exception ex) {
53         System.out.println("Error: "+ex);
54     }
55 }

```

Nota Fuente: Los Autores

Ilustración 44 Controlador Registro Aspirante Parte 2

La clase **Modelo** se utiliza para almacenar de manera temporal los datos provenientes del resultado de la sentencia ejecutada en el controlador para esto se hace uso del constructor con todos los atributos necesarios mientras a su vez se va cargando los datos en la lista **registroAspiranteModel** instanciada anteriormente del Modelo correspondiente para almacenar todos los registros existentes.



Nota Fuente: Los Autores

Ilustración 45 Modelo Registro Aspirante

8.4.3. Programación Orientada a Objetos (POO)

POO es una técnica para desarrollar soluciones utilizando objetos de software, hace referencia a las teorías, estándares, modelos y métodos que permiten organizar el conocimiento, proporcionando una visualización del problema y desplegar la solución con un lenguaje de programación, la programación orientada a objetos está basada en el modelo objeto, donde el principal es el objeto, el cual contiene las características y comportamientos, hace que todas las partes independientes se relacionen con objetos de su misma clase o de otras clase (Canchala Fernández, 2018).

Objeto:

Componente que contiene sus características (campos) como sus comportamientos (métodos) y se accede a través de su interfaz.

Campo:

Es la característica del objeto, define su estructura y permite diferenciarlo de otros, se define con un identificador y un tipo, indicando los tipos que puede almacenar

Método:

Es la implementación de una operación o función que un objeto realiza, el conjunto de métodos determina el comportamiento del objeto.

Referencia General: (Canchala Fernández, 2018)

Entre sus características principales podemos encontrar las siguientes:

 **Abstracción:** Cada objeto del sistema sirve como modelo de un "agente" que realiza el trabajo de informar, cambiar su estado, y comunicarse con otros objetos sin revelar sus características (WikiBooks, 2018).

- ✚ **Encapsulamiento:** Reúne todos los elementos que pertenecen a una misma entidad, aumenta la cohesión de los elementos del sistema.
- ✚ **Modularidad:** Permite subdividir una aplicación en partes más pequeñas (llamadas módulos), cada una de las cuales debe ser tan independiente como sea posible.
- ✚ **Polimorfismo:** Comportamiento diferente, asociados a diferentes objetos, pueden compartir el nombre; pero al llamarlo por el nombre se utilizará el comportamiento correspondiente al objeto que se esté usando.
- ✚ **Herencia:** las clases no se aíslan, solo se relacionan entre sí, formando una jerarquía.

Referencia General:(WikiBooks, 2018).

Aplicación de la Programación Orientada a Objetos en el código del aplicativo web de la siguiente manera: **Aplicando Modularidad**

```
@ManagedBean
public class Actualizar implements Serializable{
    HttpServletRequest request = (HttpServletRequest) FacesContext.getCurrentInstance().getExternalContext().getRequest();
    private boolean retorno=false;
    //Método controlado por administrador
    public void ActualizarDatos(){
        try{
            Importar imp= new Importar();
            imp.ImportarDatos();
            CedulaAspirantes();
            TelefonoAspirantes();
            CrearHistorial();
            Matriculados();
            Entrevistas();
            Documentos();
            Preinscritos();
            Alerta();
            FacesContext.getCurrentInstance().addMessage(null, new FacesMessage(FacesMessage.SEVERITY_INFO, "Correcto", "Registros Actualizados"));
        }
        catch(Exception ex){
            FacesContext.getCurrentInstance().addMessage(null, new FacesMessage(FacesMessage.SEVERITY_INFO, "Error", "Registros no Actualizados"));
        }
        System.out.println("Ok alerta");
    }
}
```

Nota Fuente: Los Autores

Ilustración 46 Aplicación de POO Parte 1

Se hace uso de instancia de un objeto **Importar** implementando la recursividad y una mejor organización del código al distribuir los métodos de tal manera que sean clases especializadas.

```
@ManagedBean
public class Importar {
    HttpServletRequest request = (HttpServletRequest) FacesContext.getCurrentInstance().getExternalContext().getRequest();
    private boolean retorno=false;
    private String periodoAcademico=PeriodoAcademico();

    //Método para saber el periodo académico actual
    public String PeriodoAcademico(){...24 lines }

    //Método controlado por Administrador
    public void ImportarDatos(){
        try{
            if(ImportarAspirantes() && ImportarInscripciones() && ImportarDocumentacion() && ImportarEntrevistas() && ImportarMatriculas()){
                FacesContext.getCurrentInstance().addMessage(null, new FacesMessage(FacesMessage.SEVERITY_INFO, "Satisfactorio", "Datos Importados"));
                System.out.println("Correcto");
            }
            else{
                System.out.println("Error errores");
            }
            CrearRegistro();
        }
        catch(Exception ex){
            System.out.println("Error: "+ex);
        }
    }
}
```

Nota Fuente: Los Autores

Ilustración 47 Aplicación de POO Parte 2

De esta manera se obtiene un mejor control sobre los métodos que sean utilizados por las clases del aplicativo, para poder detectar posibles errores que sean presentados.

Encapsulamiento:

```
@ManagedBean
public class NuevoRegistro {

    private final HttpServletRequest httpServletRequest;
    HttpServletRequest request = (HttpServletRequest) FacesContext.getCurrentInstance().getExternalContext().getRequest();
    private ArrayList<String> cod = new ArrayList<>();
    private String ndocumento;
    private String contacto;
    private String nombre;
    private String ciudad;
    private String email;
    private String telefono;
    private String celular;
    private String direccion;
    private String descripcionRegistro;
    private String etapa;
    private String periodo;
    private String programa;
    private String opcion;
    private ArrayList ciudades;
    private ArrayList otrosIntereses;
    private ArrayList periodos;
```

Nota Fuente: Los Autores

Ilustración 48 Aplicación de POO parte 3

Los atributos de todas las clases especializadas son de tipo privado por seguridad para acceder a ellas es necesario implementar los métodos **Get** y **Set** para que a través de estas se haga uso de los atributos por los métodos que la clase implemente.

```

public String getPrograma() {
    return programa;
}

public void setPrograma(String programa) {
    this.programa = programa;
}

public String getNdocumento() {
    return ndocumento;
}

public void setNdocumento(String ndocumento) {
    this.ndocumento = ndocumento;
}

public String getContacto() {
    return contacto;
}

public void setContacto(String contacto) {
    this.contacto = contacto;
}

public String getNombre() {
    return nombre;
}

public void setNombre(String nombre) {
    this.nombre = nombre;
}

```

Nota Fuente: Los Autores

Ilustración 49 Aplicación de POO Parte 4

Método que hace uso de los atributos:

```

public void Registrar(String fecha) {
    try {
        int ultimoAsp = 0;
        int ultimoReg = 0;
        int ultimoHist = 0;
        int idAspiranteExiste = 0;
        String existente;

        //Traer la variable de sesión del usuario logueado
        Object usuario_id = request.getSession().getAttribute("idUsuario");
        String user_id = String.valueOf(usuario_id);

        //Verifico el contenido de documento
        if (ndocumento.equals("")) {
            //Si es null lo dejamos por defecto en 0
            ndocumento = "0";
        }

        //Verifico el contenido de ciudad
        if (ciudad == null) {
            //Meiva por defecto
            ciudad = "665";
        }

        //verifico el contenido de telefono y EMAIL
        if (telefono.equals("") && email.equals("") && celular.equals("")) {
            //Si están vacios, se requiera al menos uno
            FacesContext.getCurrentInstance().addMessage(null, new FacesMessage("Teléfono, Celular y Correo vacios [Registrar al menos uno]"));
        } else {
            //Genero la conexión con Postgres
            Connection pcon = Postgres.Connection();
            //Cargo el número de Intereses en el ArrayList opciones
            ArrayList opciones = intereses();

```

Nota Fuente: Los Autores

Ilustración 50 Aplicación de POO Parte 5

```

if(opciones.isEmpty()){
    FacesContext.getCurrentInstance().addMessage(null, new FacesMessage("No has seleccionado un interés"));
}
else{
    Iterator<String> interes = opciones.iterator();
    //Verifico si el Aspirante ya se encuentra registrado
    existente = ValidarExistencia();
    if (existente.equals("")) {
        //Agrego un nuevo aspirante
        ultimoAsp = UltimoAspirante();

        sentencia="SELECT * FROM sp_nuevo_aspirante('" + ultimoAsp + "','" + ndocumento.trim() + "','"
            + nombre.toUpperCase().trim() + "','" + telefono.trim() + "','" + celular.trim()+ "','"
            + email.trim() + "','" + ciudad.trim() + "','" + direccion.toUpperCase().trim() + "','" + 'A'");

        CallableStatement spAspirante = pcn.prepareStatement(sentencia);
        spAspirante.execute();

        //Recorro el array con los Intereses [Nº Registros = Nº Intereses]
        while (interes.hasNext()) {

            //obtengo el último registro
            ultimoReg = UltimoRegistro();

            //obtengo el último registro del Historial
            ultimoHist = UltimoHistorial();

            //Agrego el registro
            sentencia="SELECT sp_nuevo_registro('" + ultimoReg + "','" + user_id + "','" + ultimoAsp + "','"
                + ndocumento + "','" + contacto.toUpperCase() + "','" + descripcionRegistro.toUpperCase().trim() + "','"
                + fecha + "','" + '1' + "','" + interes.next() + "','" + periodo + "','" + '0' + '0');";
            CallableStatement spRegistro = pcn.prepareStatement(sentencia);
            CallableStatement spHistorial;

```

Nota Fuente: Los Autores

Ilustración 51 Aplicación de POO Parte 6

8.5. Fase de Prueba o Testing

Se destaca de manera importante que no se estableció una metodología de pruebas para la aplicación web por lo cual se diseñaron casos de prueba, en las funcionalidades principales de la aplicación, y los resultados serán plasmados en fotos visualizando su resultado, aunque no se está siguiendo de forma estricta ninguna metodología se están tomando algunos pasos de esta para el desarrollo de las pruebas.

8.5.1. Test de Usabilidad

El objetivo es dar apoyo al desarrollo de interfaces que faciliten el trabajo que cualquier persona deba hacer ante una pantalla, permitiendo interactuar, con la información que ofrece, debido a que las especificaciones son concretas es posible medir su cumplimiento (Almazán & Camus, 2016).

De acuerdo con lo anterior se presenta el test de usabilidad implementado en el área de admisiones:

Objetivo del Test de Usabilidad

Estamos trabajando en el diseño de un aplicativo web para la fidelización de estudiantes de la Corporación Universitaria del Huila- Corhuila y durante el proceso de desarrollo se intenta realizar algunas funciones para revisar su comportamiento.

El test va a durar aproximadamente 20 minutos. Cabe resaltar que se testea el aplicativo, no las habilidades. La tarea principal es mejorarlo para hacerlo más usable.

Instrucciones: Complete el siguiente cuestionario de acuerdo con la percepción que usted tiene del aplicativo web, su tarea es indicar el grado de satisfacción con cada pregunta los criterios para esta evaluación son los siguientes:

Tabla 27 Criterios de Evaluación

Puntuación	Criterios de Evaluación
1	Malo
2	Regular
3	Bueno
4	Muy Bueno
5	Excelente

Nota Fuente: Los Autores

Los criterios por evaluar en este Test de Usabilidad:

Identidad: Se evaluarán los rasgos, atributos o las características propias del aplicativo web asociadas a la institución.

Contenido: Validar que el contenido de la aplicación web concuerde con los requerimientos funcionales solicitados. Ejemplo (registros, etapas, historial etc.).

Navegación: Busca que el usuario encuentre la información necesaria en las diferentes vistas de una manera fácil.

Acceso: indicar la capacidad de ingreso a todas las vistas del aplicativo web.

De acuerdo con lo descrito anteriormente ingrese al siguiente enlace <http://inproti.corhuila.edu.co:8080/UHCRM/> y complete las preguntas en cada una de las categorías marcando con una (X) los criterios de evaluación.

Tabla 28 Test de Usabilidad

Nº	IDENTIDAD	1	2	3	4	5
1	Con la información presentada en el inicio de sesión es posible saber a qué institución corresponde la aplicación.					X
2	Presenta algún elemento grafico o texto que le de indicios claros a que institución pertenece.					X
3	Relaciona los colores del aplicativo web con la institución					X
4	Distingue la imagen y el nombre de la institución dentro del aplicativo web					X
5	Relaciona la dirección web del aplicativo con la dirección de la institución					X
Nº	CONTENIDO	1	2	3	4	5
1	La aplicación web ofrece los contenidos necesarios para las actividades realizadas por la institución.				X	
2	Los textos usados en el contenido son los suficientemente descriptivos para acceder a través de ellos.				X	
3	Es fácil distinguir la actualización de estados y alertas ofrecidos por el aplicativo web.				X	
4	La información ofrecida por los archivos adjuntos ofrecidos por el aplicativo web contiene la información necesaria					X
5	El historial ofrecido en el aplicativo web ofrece la suficiente información				X	
Nº	NAVEGACION	1	2	3	4	5
1	El diseño de las interfaces de la aplicación web es consistente				X	
2	Se marcan los campos obligatorios en el formulario registro aspirante con algún texto o signo que indique la obligatoriedad.				X	
3	La agrupación de los campos del formulario registro aspirante se agrupan en varios grupos para facilitar la introducción de los datos.				X	
4	Se muestra los errores que el usuario comete cuando llena el formulario.					X

- 5 El menú del aplicativo web está en un lugar destacado **X**

Nº	ACCESO	1	2	3	4	5
1	La letra es del tamaño adecuado en la aplicación web.				X	
2	Todos los formularios cuentan con un título que indique la actividad presente en la interfaz.				X	
3	Los vínculos del aplicativo web son visibles al usuario			X		
4	Es fácil llegar a otras secciones del aplicativo web desde el menú				X	
5	El diseño del Menú está en una parte visible y de fácil acceso en la vista				X	

Nota Fuente: Los Autores

Una vez se realizó el test de usabilidad, se concluye que la aplicación web tiene un buen recibimiento entre los usuarios, la forma en que se muestra la información es de forma ordenada y por secciones según la etapa cada uno de los módulos está separado según su función.

Hecha la primera parte de la prueba se iniciará la segunda fase implementando las pruebas unitarias por medio de casos de prueba para una conclusión más estable sobre el aplicativo web.

8.5.2. Pruebas Unitarias

Se utiliza para comprobar que un método en concreto del código funciona, este tipo de pruebas se pueden repetir tantas veces como se quiera, estas deben cubrir la totalidad del código (Probar, Unitarias, & Test, 2006).

A continuación, se presentan las pruebas unitarias del aplicativo web del siguiente formato:

Nota Fuente Adaptado: Admin. (2016). ¡Un Caso de Prueba de ejemplo! - Testing Colombia. Retrieved April 16, 2018, from <http://www.testingcolombia.com/un-caso-de-prueba-de-ejemplo/>

Tabla 29 Caso de Prueba CP01

Caso de Prueba CP01			
Objetivo del caso de prueba	Validar el inicio de sesión de un Usuario y agregar un registro de ingreso.		
Identificador	CP-01		
Nombre del caso	Inicio Usuario		
Precondiciones	El Usuario debe estar creado con anterioridad y poseer permisos de ingreso.		
Pasos	Resultado Esperado	Resultado Real	
1) En la vista 'Ingresar' Se deben ingresar las credenciales del usuario.	N/A		
2) Hacer clic sobre el botón “Ingresar” para validar credenciales.	Si los datos coinciden mostrará la vista principal, de lo contrario notificará inconsistencia.		

Ilustración 52 Inicio de Sesión y Acceso

Tabla 30 Caso de Prueba CP02

Caso de prueba CP02		
Objetivo del caso de prueba	Crear un nuevo registro a un aspirante que requiera información de los programas académicos de la Corporación Universitaria del Huila – Corhuila.	
Identificador	CP-02	
Nombre del caso	Nuevo Registro	
Precondiciones	Para crear un Usuario es requerido que se realice desde el rol de administrador, ya que es el único con permisos para realizar esta acción.	
Pasos	Resultado Esperado	Resultado Real
1) Desde el menú principal se selecciona el módulo “CRM” y a continuación damos clic en el submenú “Registro Aspirante”	Se mostrará la vista de registro con el respectivo formulario.	

2) Ingresar datos generales del registro. N/A

(*) Contacto:

(*) Nombre Aspirante: C.C.:

Teléfono: Dirección:

Celular: Ciudad:

Email: (*) Período Académico:

3) Seleccionar los Programas Académicos de interés del Aspirante N/A

Programas Académicos

☐ Administración Bancaria	☐ Ingeniería Ambiental
☐ Administración Comercial	☒ Ingeniería Sistemas
☐ Administración Empresas Agropecuarias	☐ Ingeniería Industrial
☐ Administración Empresas Turísticas	☐ Ingeniería Mecatrónica
☐ Negocios Internacionales	☒ Mercadeo, Publicidad y Ventas
☐ Medicina Veterinaria y Zootecnia	-OTRO-

4) Ingresar una descripción como información adicional del registro. N/A

Descripción:

5) Dar clic en el botón “Guardar” para proceder a guardar los datos del registro. Se procede a validar los datos del registro, si no genera inconsistencia en los formatos de los datos o registro repetido, mostrará la notificación de registro exitoso.

Registro Aspirante

Se realizó el registro satisfactoriamente

(*) Contacto:

(*) Nombre Aspirante:

Teléfono:

Celular:

Email:

C.C.:

Dirección:

Ciudad:

(*) Período Académico:

Programas Académicos

☐ Administración Bancaria	☐ Ingeniería Ambiental
☐ Administración Comercial	☐ Ingeniería Sistemas
☐ Administración Empresas Agropecuarias	☐ Ingeniería Industrial
☐ Administración Empresas Turísticas	☐ Ingeniería Mecatrónica
☐ Negocios Internacionales	☐ Mercadeo, Publicidad y Ventas
☐ Medicina Veterinaria y Zootecnia	☐ -OTRO-

Ilustración 53 Formulario Registro

Tabla 31 Caso de Prueba CP03

Caso de Prueba CP03		
Objetivo del caso de prueba	Actualizar los datos de los registros, para generar las alertas y los saltos de etapas.	
Identificador	CP03	
Nombre del caso	Actualizar Registros.	
Precondiciones	La actualización deberá ser realizada exclusivamente por el administrador.	
Pasos	Resultado Esperado	Resultado Real
1) Desde el menú principal se selecciona el módulo “Usuario” y a continuación damos clic en el submenú “Actualizar”	Actualizará los registros, incluyendo la generación de alertas, los saltos de etapas.	
2) Ver las actualizaciones realizadas a los registros, entrando al módulo “CRM” y ver los registros desde la vista “Registros Aspirantes”.	Los datos estarán actualizados y con alertas respectivas.	

Ilustración 54 Vista Registro

Tabla 32 Caso de Prueba CP04

Caso de prueba CP04		
Objetivo del caso de prueba	Creación de un nuevo usuario y asignación de roles.	
Identificador	CP01 – CP02	
Nombre del caso	Usuarios Nuevos - Roles y Permisos	
Precondiciones	Es necesario realizar esta acción desde el administrador	
Pasos	Resultado Esperado	Resultado Real
1) Desde el menú principal se selecciona el módulo “Seguridad” y a continuación damos clic en el submenú “Control”	Se mostrará la vista de administración.	
2) Se procede a ingresar los datos del usuario que se desea crear.	N/A	
3) Una vez hayamos ingresado los datos necesarios, presionamos el botón “Nuevo Usuario”	Los datos serán validados y posteriormente guardados con el rol suministrado por la dependencia.	

Ilustración 55 Vista Administrador - Rol y permisos

Tabla 33 Caso de Prueba CP05

Caso de Pruebas CP05		
Objetivo del caso de prueba	Eliminar Usuario	
Identificador	CP03	
Nombre del caso	Modificar y Eliminar Usuario.	
Precondiciones	Es necesario realizar esta acción desde el administrador	
Pasos	Resultado Esperado	Resultado Real
1) Desde el menú principal se selecciona el módulo “Seguridad” y a continuación damos clic en el submenú “Control”	Se mostrará la vista de administración.	
2) Se procede a seleccionar un usuario a eliminar y se da clic sobre el botón con el icono del bote de basura.	Mostrará un mensaje donde solicitará al usuario una confirmación para continuar.	
3) El administrador deberá proceder a escoger una de las dos opciones para eliminar o no hacerlo.	Si la respuesta es sí, se eliminará de la base de datos.	

Ilustración 56 Vista Administrador- Eliminar Usuario

Se concluye que después de realizadas las pruebas en sus dos fases por parte de los usuarios el aplicativo web es aceptado de manera satisfactoria por parte del área de admisiones cumpliendo con todas las necesidades planteadas anteriormente.

8.6. Fase de Implementación

En esta fase se hace un recuento de las tecnologías, motores y demás empleados en el funcionamiento de la aplicación web.

8.6.1. Servidor Web

Es un programa que responde a las peticiones de los usuarios, proporcionándole los diferentes recursos mediante el protocolo HTTP o HTTPS, la versión segura, cifrada y autentica de HTTP, aunque su funcionamiento es muy sencillo por su ejecución en bucle infinito. El cual espera peticiones por el puerto TCP asignado, con un puerto listo recibe la petición, con una petición en lista busca el recurso en la cadena y esta es enviada por la misma conexión volviendo a su estado de recepción de peticiones (Barzanallana, 2012).

8.6.2. Servidor de Aplicaciones GlassFish

Implementa las tecnologías definidas en la plataforma de Java EE, que soporta tecnologías como JSF, JSP, Servlets, API para servicios web, enlaces XML (JAXB) etc. (Manchado & Puentes, 2010). Proporciona funcionalidades de forma transparente al usuario, lo cual no es necesario la escritura de código fuente esto es posible debido a su funcionamiento dentro de un espacio virtual dentro del contenedor llamado dominio de ejecución que se interpone entre las llamadas hechas a los métodos, verificando si el usuario que llama al método tiene las credenciales permitidas antes de hacer la solicitud (Manchado & Puentes, 2010).

GlassFish se compone de una arquitectura modular, permitiendo solo la instalación de los módulos necesarios para la aplicación haciendo de esta más flexible evitando el consumo de memoria, espacio en disco y el tiempo de inicio de las aplicaciones, pueden ser instalados remotamente y sin reinicio del servidor (Manchado & Puentes, 2010).

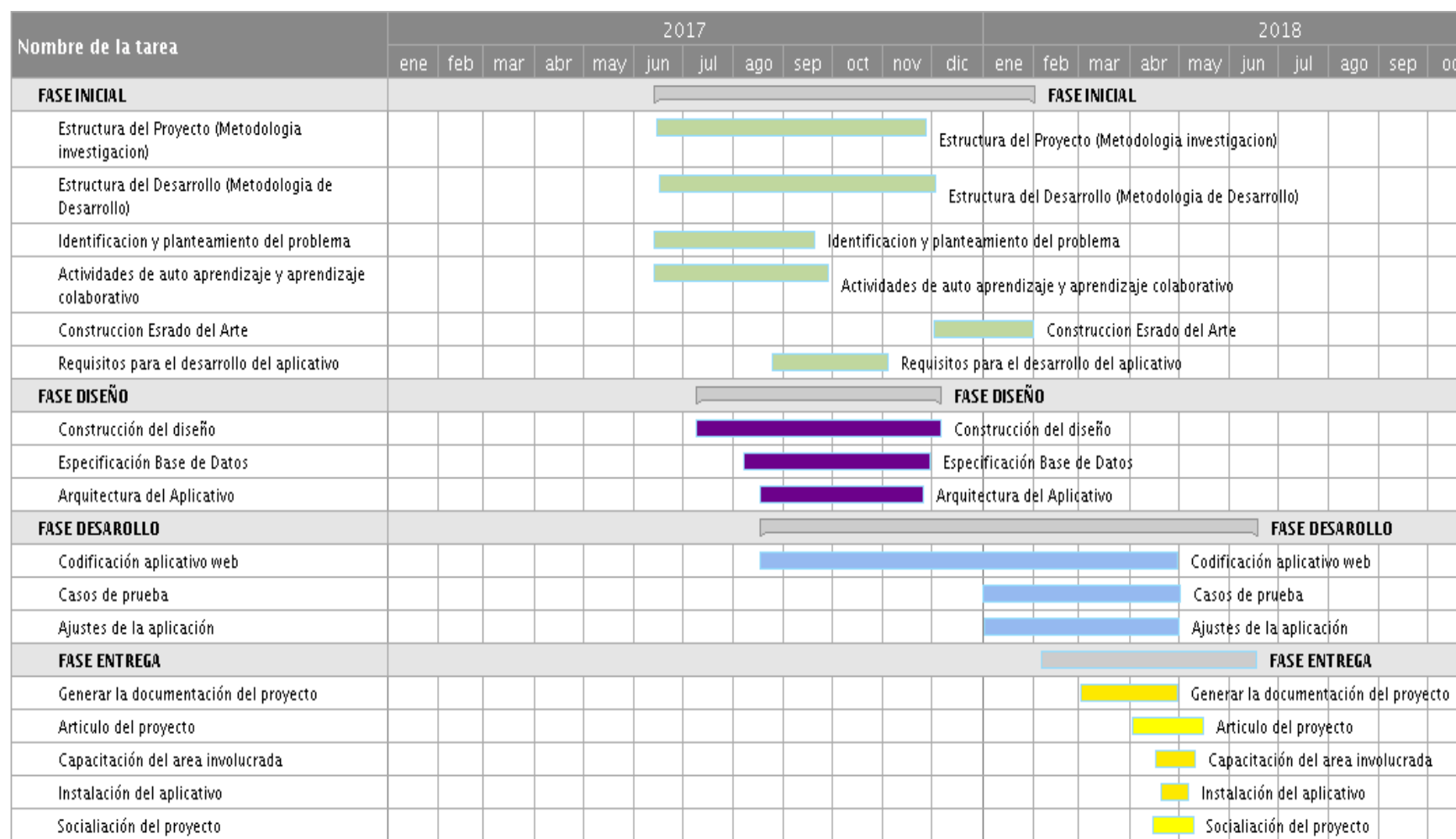
8.6.3. Motor de Base de Datos PostgreSQL

Las características de principales de PostgreSQL fueron lo que determinaron la decisión de implementarla en el aplicativo para el resguardo de la información a continuación mencionaremos algunas:

- Multiplataforma lo que permite ejecutarse en diferentes sistemas operativos entre estos GNU/Linux y Windows.
- Incluye vistas
- Permite el almacenamiento de video, imágenes, sonido.
- Permite realizar copias de respaldo en línea o en caliente (Online/hot backups).
- Integra llaves foráneas, permite las uniones.
- dispone de disparadores (triggers).
- Incluye la mayoría de los tipos de datos de SQL, como los son integer, numeric, boolean, char, varchar, date etc.

Referencia General: (Pavas Correa & Neira Mosquera, 2015)

9. Cronograma de Actividades



Nota Fuente: Los Autores

Ilustración 57 Cronograma de Actividades

10. Presupuesto

Tabla 34 Presupuesto Global por Fuentes de Financiación

RUBROS	FUENTES		TOTAL
	CORHUILA	CONTRAPARTIDA GRUPO DE INV.	
PERSONAL DE PLANTA	No financiable	\$14.650.000	
COMPRA DE EQUIPOS	0	\$2.500.000	
SOFTWARE	0	0	
MATERIALES E INSUMOS	0	\$186.000	
SALIDAS DE CAMPO	0	0	
MATERIAL BIBLIOGRÁFICO	No financiable	No financiable	
SERVICIOS TÉCNICOS	0	0	
DIVULGACIÓN DE LA INVESTIGACIÓN	\$1.200.000	0	
CONSTRUCCIONES	No financiable	No financiable	
OTROS	No financiable	No financiable	
IMPREVISTOS	No financiable	No financiable	
MANTENIMIENTO	No financiable	No financiable	
TOTAL	\$1.200.000	\$17.336.000	\$18.536.000

Tabla 35 Descripción del Personal Investigador

NOMBRE DEL INVESTIGADOR / EXPERTO/ AUXILIAR	FORMACIÓN ACADÉMICA	FUNCIÓN EN EL PROYECTO	DEDICACIÓN HORAS/ SEMANA	RECURSOS	
				CORHUILA (no financiable)	CONTRAPARTIDA GRUPO DE INVESTIGACIÓN
José Miguel Llanos Mosquera	Maestría	Inv. Principal	4 horas	0	\$13.450.000
Henry Andrés Perdomo Roa	Pregrado	Coinvestigador	8 horas	0	\$600.000
Martha Milena Vanegas Capera	Pregrado	Coinvestigador	8 horas	0	\$600.000
TOTAL				0	\$14.650.000

Tabla 36 Materiales e Insumos

Materiales*	Justificación	Cantidad	Valor unitario	Valor total
Resmas de papel carta	Documentación del proyecto	2	\$10.000	\$20.000
Fotocopias	Aplicación del instrumento de recolección de información	200	\$100	\$20.000
Impresiones	Impresión de documentos, proyecto y artículo investigativo	400	\$200	\$80.000
Memoria USB de 16GB	Almacenamiento de información.	2	\$33.000	\$66.000
TOTAL				186.000

11. Conclusiones

Empleando la metodología Scrum para el desarrollo del aplicativo web, se economizó en tiempo para levantar los requisitos de la aplicación, lo cual fue de gran ayuda al momento de hacer las modificaciones pertinentes, teniendo el acompañamiento del área en cuestión se crea un intercambio de información más eficiente y se cuenta con la aprobación del cliente que permite el desarrollo en tiempos más cortos.

Se Construyo una base de datos, al conocer el proceso interno del área de admisiones que permite acceder a la información de una manera más precisa y actualizada, ayudando a economizar tiempo en los procesos de monitoreo por parte de los usuarios.

La implementación de las pruebas durante cada fase del sprint permitió el monitoreo a los errores y fallas presentadas, a los módulos recientemente codificados durante cada entrega, consiguiendo un mayor nivel y asegurando un producto en calidad.

Referencias

- Almazán, F., & Camus, J. (2016). Modelo de Test de Usuario, 1–15. Retrieved from <http://www.guiadigital.gob.cl/guia-v2/capitulos/05/anexos/pauta-test-usuario.pdf>
- Andajani, E. (2015). Understanding Customer Experience Management in Retailing. *Procedia - Social and Behavioral Sciences*, 211(September), 629–633. <https://doi.org/10.1016/j.sbspro.2015.11.082>
- Arroyo López, M. del P. E. (2010). LA IMAGEN INSTITUCIONAL COMO DETERMINANTE DE LAS EXPECTATIVAS DE SERVICIO Y LA LEALTAD EN EL SECTOR DE EDUCACIÓN SUPERIOR PRIVADA. In *Congreso Internacional de Contaduría, Administración e Informática* (pp. 1–22). San Antonio Buenavista, Toluca, Méx. C. P. Retrieved from <http://congreso.investiga.fca.unam.mx/docs/xv/docs/14.pdf>
- Assunção, M., Silva, R., & D'Amore, T. (2015). O Papel Do Crm Social Em Uma Instituição De Ensino: O Caso Do Instituto Federal De Educação, Ciência E Tecnologia Do Rio Grande. *Holos*, 3, 1–11. <https://doi.org/10.15628/holos.2015.2888>
- Barzanallana, R. (2012). *Desarrollo de Aplicaciones Web*. Retrieved from <http://www.um.es/docencia/barzana/DAWEB/Desarrollo-de-aplicaciones-web-Xampp.html>
- BRIONES, D. S. S. (2017). *DISEÑO E IMPLANTACIÓN DEL PROCESO DE CRM EN LA UNIVERSIDAD DEL AZUAY , PARA LA GESTIÓN DE LOS INSTITUTOS DE BACHILLERATO*. Universidad Politecnica de Valencia.
- Brito, J. G., Laaser, W., & Toloza, E. A. (2012). El uso de redes sociales por parte de las universidades a nivel institucional. Un estudio comparativo. *RED, Revista de Educación a Distancia*, 32, 1–38. Retrieved from <http://search.ebscohost.com/login.aspx?direct=true&db=fua&AN=90229759&lang=es&site=ehost-live>
- Calsina Ponce, A. J., & Cazorla Choque, R. (2012). “propuesta de integración de google web toolkit con glassfish y enterprise java bean para desarrollar un sistema web y la administración de informacion.” *Revista Española de Documentación Científica*, 3, 9–14.
- Canchala Fernández, L. A. (2018). Fundamentos de la POO. Retrieved April 13, 2018, from <https://msdn.microsoft.com/es-co/library/bb972232.aspx>
- Castillo Lorente, K., Martínez, R., & Marín Sánchez, J. C. (2013). *DISEÑO DE CRM PARA LA OFICINA DE MERCADEO DE LA UNAC*. Corporación Universitaria Adventista.
- Cataldo, S., Massaro, R., & Biglia, A. (2014). Marketing educativo. Cómo captar, retener y fidelizar alumnos. In *Reflexión Académica en Diseño y Comunicación* (Vol. 22, pp. 47–50).
- Cobo Yera, A. (2007). *Diseño y programación de bases de datos*. Visión Libros. Retrieved from <https://books.google.com.co/books?id=anCDr9N-kGsC&pg=PA7&lpg=PA7&dq=Diseño+y+Programacion+de+Bases+de+Datos&source=bl&ots=UYDBn4uwuU&sig=HMLQ7GesQ7paJKGeXJ6LfnQ-wxg&hl=es-419&sa=X&ved=0ahUKEwj3roSFyojaAhXlguAKHWINATkQ6AEIcjAI#v=onepage&q&f=false>
- Córdoba Filio, C. C., & Julca Segovia, C. A. (2005). Implementacion del Modelo CRM para una Institucion educativa: caso de aplicacion FISI-UNMSM, 106.
- Dimes, T., & Jimenez, M. (2015). *Conceptos Basicos De Scrum : Desarrollo De Software Agile Y*

- Manejo De Proyectos Agiles*. Babelcube Incorporated. Retrieved from <https://books.google.com.co/books?id=ETuXBgAAQBAJ>
- Duque, P. G. – C. & B. S. R. (2017, January 17). CRM en el sector educativo: Aplicaciones prácticas. Retrieved April 14, 2018, from <https://www.intelligencepartner.com/crm-sector-educativo-aplicaciones-practicas/>
- Ellucian. Case Study: Managing growth through technology (2016).
- Ellucian. (2017). *Ellucian CRM: platform overview Enterprise-class, higher education-specific Built*.
- Fernández Romero, Y., & Díaz Gonzáles, Y. (2012). Patrón Modelo-Vista-Controlador. *Revista Telem@tica*, 11(1), 1–11. Retrieved from <http://revistatelematica.cujae.edu.cu/index.php/tele/article/view/15/10>
- Franco Restrepo, J. G., & Arrubla Zapata, J. P. (2014). MARKETING EN UNIVERSIDADES : Descripción ,Análisis y Propuesta. In *XVI Congreso Internacional De Contaduría, Administración e Informática. Ponencia* (p. 24). <https://doi.org/10.13140/2.1.2464.9921>
- García Villalobos, A. (2006). *Propuesta de un modelo de marketing interno en el CBtis 39. Universidad Autónoma de aguascalientes*. Universidad Autonoma de Aguascalientes. Retrieved from <http://bdigital.dgse.uaa.mx:8080/xmlui/bitstream/handle/123456789/138/274486.pdf?sequence=1>
- Garrido-Moreno, A., & Padilla-Meléndez, A. (2011). Analyzing the impact of knowledge management on CRM success: The mediating effects of organizational factors. *International Journal of Information Management*, 31(5), 437–444. <https://doi.org/10.1016/j.ijinfomgt.2011.01.002>
- Gozales, J. (2015). Qué es el Código Limpio? | MondongoSoft. Retrieved April 4, 2018, from <https://mondongorongo.wordpress.com/2015/01/21/que-es-el-codigo-limpio/>
- Grupo Educacion al Futuro. (2011). A través de la redes sociales: Oportunidad para captar alumnos. Retrieved March 26, 2018, from <https://educacionalfuturo.dev.braulioaquino.net/wp/mundo-educativo/a-traves-de-la-redes-sociales-oportunidad-para-captar-alumnos/>
- Gutiérrez, J. (2006). ¿ Qué es un framework web?, 1–4. Retrieved from http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf
- Kruntchen, P. (1995). Architectural blueprints—the” 4+ 1” view model of software architecture. *IEEE Software*, 12(November), 42–50. <https://doi.org/10.1145/216591.216611>
- Llorente Alonso, C. (2017). *Marketing educativo : captación y fidelización de alumnos*. ESIC.
- López Torres, M. del R. (2017). Estrategias de marketing interno para incrementar la motivación del cliente interno - Grupo Milenio. Retrieved January 11, 2018, from http://www.milenio.com/firmas/universidad_politecnica_de_tulancingo/Estrategias-marketing-interno-incrementar-motivacion_18_900090056.html
- Mancha, C.-L. A. (2012). RUIDERAe: Revista de Unidades de Información. Número 2 (2012) ISSN 2254-7177, 2, 1–26.
- Manchado, D. S., & Puntos, D. F. (2010). Estudio del servidor de aplicaciones Glassfish y de las aplicaciones J2EE, 1–182.
- Marzo Navarro, M., Pedraja Iglesias, M., & Rivera Torres, P. (2004). Tipología de estudiantes en

- función de su satisfacción con los cursos de verano. *Revista ELección de Investigación Y Evaluación Educativa*, 81–96. Retrieved from http://www.uv.es/RELIEVE/v10n1/RELIEVEv10n1_5.htm
- Mayo, J. (2017). *Programming the Microsoft Bot Framework: a multiplatform approach to building chatbots*. Pearson Education.
- Mendoza, G. (2015). Herramienta de Desarrollo Netbeans, 1–5. Retrieved from http://www.consultorjava.com/wp/wp-content/uploads/2015/09/herramienta_desarrollo_netbeans.pdf
- Mesén Figueroa, V. (2011). Fidelización de clientes: concepto y perspectiva contable. *Tec Empresarial*, ISSN-E 1659-3359, Vol. 5, N°. 3, 2011, Págs. 29-35, 5(3), 29–35. <https://doi.org/10.16805/1659-2395>
- Meyer, C., & Schwager, A. (2007). Understanding Customer Experience. *Harvard Business Review*, (Harvard Business Review). Retrieved from <https://hbr.org/2007/02/understanding-customer-experience>
- Naranjo Hernández, C. (2011). Marketing Educativo; desarrollo de una estrategia C.E.M. aplicado a a Universidad Nacional sede Manizales como base para la fidelización de clientes y complemento a la estrategia de C.R.M., 1–172.
- Neghina, C. (2016). CRM systems in higher education. *StudyPortals*, 1. Retrieved from <http://www.studyportals.com/intelligence/crm-in-higher-education/>
- Niño Rojas, V. M. (2011). *Metodología de la investigación: diseño y ejecución*. Investigacion Diseño y Ejecución. Retrieved from [http://roa.ult.edu.cu/bitstream/123456789/3243/1/METODOLOGIA DE LA INVESTIGACION DISENO Y EJECUCION.pdf](http://roa.ult.edu.cu/bitstream/123456789/3243/1/METODOLOGIA_DE_LA_INVESTIGACION_DISENO_Y_EJECUCION.pdf)
- Oracle. (2017). JavaServer Faces Overview. Retrieved March 25, 2018, from <http://www.oracle.com/technetwork/java/javaee/overview-140548.html>
- Oracle Corporation. (2017). ¿Qué es Java y para qué es necesario? Retrieved March 25, 2018, from https://www.java.com/es/download/faq/whatis_java.xml
- Pavas Correa, D. S., & Neira Mosquera, Y. L. (2015). *DISEÑO E IMPLEMENTACION DE UNA PLATAFORMA WEB PARA LA SOLICITUD, ADMINISTRACION Y GESTION DE ORDENES DE COMPRA E INVENTARIOS EN LA EMPRESA TECNOPRECISION DAVID*. UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS.
- Pech-May, F., Gomez-Rodriguez, M. A., Cruz-Diaz, L. A. de la, & Lara-Jeronimo, S. U. (2010). Desarrollo de Aplicaciones web con JPA, EJB, JSF y PrimeFaces, 9. Retrieved from <http://www.tamps.cinvestav.mx/~fpech/sd/files/paper001.pdf>
- Petrella, C. (2008). Gestión de la relación de las universidades con docentes, estudiantes y egresados. *Revista Iberoamericana de Educacion*, 47(5), 1–14.
- Pineda Rodríguez, L. M., & Jaime Bohada, A. J. (2017). METODOLOGIA CRM ORIENTADA A LAS UNIVERSIDADES PÚBLICAS EN COLOMBIA Resumen: CRM METHODOLOGY ORIENTED TO PUBLIC UNIVERSITIES IN COLOMBIA Abstract METODOLOGIA DE CRM ORIENTADA A UNIVERSIDADES PÚBLICAS NA COLÔMBIA Resumo. *Face-Facultad de Ciencias Economica Y Empresariales*, 65–72.
- Probar, J. R., Unitarias, P., & Test, U. (2006). Pruebas unitarias.
- Programming guidelines. (2017). Programming guidelines - HaskellWiki. Retrieved April 1, 2018,

- from https://wiki.haskell.org/Programming_guidelines#File_Format
- Publicaci, B., & Cardiovasculares, I. (2013). Tipos de diseños de los estudios clínicos y epidemiológicos (Types of clinical and epidemiologic study designs) Resumen Introducción. *Portal de Revistas Universidad de Los Andes*, 2(2), 76–99. Retrieved from <http://150.185.138.216/index.php/biomedicina/article/download/4410/4200>
- Quichimbo Sangurima, L. A., & Vallejo Bojorque, A. P. (2012). “Desarrollar una propuesta de implantación de un sistema CR M (CUSTOMER RELATIONSHIP MANAGEMENT/ ADMINISTRACION DE RELACIONES CON EL CLIENTE) para la Universidad Politécnica Salesiana Sede Cuenca. UNIVERSIDAD POLITÉCNICA SALESIANA SEDE CUENCA. Retrieved from <http://dspace.ups.edu.ec/bitstream/123456789/2107/12/UPS-CT002384.pdf>
- Rajola, F. (2013). Customer Relationship Management: Organizational and Technological Perspectives. Springer Berlin Heidelberg. Retrieved from <https://books.google.com.co/books?id=U2L2BwAAQBAJ>
- Rigo, G.-E., Pedron, C. D., Caldeira, M., & Araújo, C. C. S. de. (2016). CRM Adoption in a Higher Education Institution. *Journal of Information Systems and Technology Management*, 13(1), 45–60. <https://doi.org/10.4301/S1807-17752016000100003>
- Rivera, F. L. O. (2008). *Base de datos relacionales*. ITM. Retrieved from https://books.google.com.co/books?id=buM5rlZME-cC&dq=modelo+entidad+relacion+de+una+base+de+datos&source=gbs_navlinks_s
- Roca, G. (2006). Sumario : La presencia de las universidades en la red. In *Revista de Universidad y Sociedad del Conocimiento* (Vol. 3, p. 80). Retrieved from <http://revista-iberoamericana.pitt.edu/ojs/index.php/Iberoamericana/article/view/5396>
- Rocha, R. (2017). El modelo entidad-relación, 1–55.
- Ruiz Correa, E. C., & Villegas Quiroga, L. V. (2013). EL SERVICIO AL CLIENTE UNA HERRAMIENTA QUE FORTALECE LAS MARCAS. Santiago de Cali. Retrieved from <https://red.uao.edu.co/bitstream/10614/5627/1/TCP02068.pdf>
- Satpathy, T. (2016). *Una guía para el cuerpo de conocimiento de scrum (Guía SBOK)*. SCRUMstudy™, una marca de VMedu, Inc.
- Schwaber, K., & Sutherland, J. (2013). La Guía de Scrum La Guía Definitiva de Scrum: Las Reglas del Juego.
- SofDoit So easyto compare. (2018). Crm universidad. Ranking neutral. Retrieved January 15, 2018, from <https://www.softwaredoit.es/crm-universidad/crm-universidad.html>
- Solis Hurtado, J. M. (2004). *El Manejo del Marketing en el servicio educativo; en la organización educativa continental*. Universidad Nacional Mayor de San Marcos.
- Study Case:Provide students with better services and Support. The American University of Kuwait implements technology to help provide a better student experience and help students succeed (2016).
- Taboada González, J. A., & Cotos Yáñez, J. M. (2005). *Sistemas de información medioambiental*. Netbiblo.
- Taide, M. (2016). Algunas herramientas para un buen Marketing Educativo de su escuela o Universidad | Consultoria y Marketing para Colegios. Retrieved January 8, 2018, from <http://consultoriaparaColegios.com/algunas-herramientas-de-marketing-educativo-para-su->

- universidad/
- Tecnósfera. (2017, August 8). Ya hay más de 3.000 millones de usuarios de redes sociales en el mundo. *El Tiempo*, p. 1. Retrieved from <http://www.eltiempo.com/tecnosfera/novedades-tecnologia/numero-de-usuarios-de-redes-sociales-en-el-mundo-117556>
- The PostgreSQL Global Development Group. (2017). PostgreSQL 9.6.2 Documentation The PostgreSQL Global Development Group, 3533. Retrieved from <https://www.postgresql.org/files/documentation/pdf/9.6/postgresql-9.6-US.pdf>
- Trigas Gallego, M. (2012). Gestión de Proyectos Informáticos. Metodología Scrum. *Openaccess.Uoc.Edu*, 56. Retrieved from <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/17885/1/mtrigasTFC0612memoria.pdf>
- Universidad Catolica San Antonio. (2012). *Captación de estudiantes. información, orientación y apoyo a los matriculados*.
- Universidad de Valencia. (2017). Métodos. Retrieved April 3, 2018, from https://www.uv.es/webgid/Descriptiva/331_mtodos.html
- Universidad del Bosque. (2009). *Manual de Identidad Visual*. Bogota DC.
- Vaquero, B., Portabales, Y., & de la Fuente, J. (2018). Master Executive en Marketing Relacional, CRM y Comercio Electrónico (Marketing Relacional y CRM). Retrieved from <http://www.icemd.com/>
- Wali, A. F., & Wright, L. T. (2016). Customer relationship management and service quality: influences in higher education. *Journal of Customer Behaviour*, 15 (1)(1475–3928), 67–79. Retrieved from <http://www.ingentaconnect.com/contentone/westburn/jcb/2016/00000015/00000001/art00006>
- WikiBooks. (2018). Programación Orientada a Objetos/Características de la POO. Retrieved April 13, 2018, from https://es.wikibooks.org/wiki/Programación_Orientada_a_Objeto/Características_de_la_POO
- Yanez, D. (2018). Método descriptivo: características, etapas y ejemplos - Lifeder. Retrieved April 3, 2018, from <https://www.lifeder.com/metodo-descriptivo/>