

**Universidad de las Ciencias Informáticas  
Facultad # 3**



**Módulo Control Interno para el Sistema  
de Administración y Economía de la  
Facultad 3**

**Trabajo de Diploma para optar por el título de  
Ingeniero en Ciencias Informáticas**

**Autores:**

Yanirosa Almenares Ramirez  
Yeisel Pérez Rivera

**Tutor:**

Ing. Idel Jorge Sánchez González

**La Habana, Junio 2015**



*"No creo que seamos parientes muy cercanos, pero si usted es capaz de temblar de indignación cada vez que se comete una injusticia en el mundo, somos compañeros, que es más importante."*

*Ernesto "Che" Guevara*

# *Declaración de Autoría*

## DECLARACIÓN DE AUTORÍA

Declaramos ser autoras de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

---

Firma del Autor  
Yanirosa Almenares Ramirez

---

Firma del Autor  
Yeisel Pérez Rivera.

---

Firma del Tutor  
Ing. Idel Jorge Sánchez González

# *Datos de contacto*

## DATOS DE CONTACTO

**Autor:** Yanirosa Almenares Ramirez

**Correo electrónico:** [yalmenares@estudiantes.uci.cu](mailto:yalmenares@estudiantes.uci.cu)

**Autor:** Yeisel Pérez Rivera.

**Correo electrónico:** [yprivera@estudiantes.uci.cu](mailto:yprivera@estudiantes.uci.cu)

**Tutor:**

Ing. Idel Jorge Sánchez González

**Correo electrónico:** [isgonzalez@uci.cu](mailto:isgonzalez@uci.cu)

## AGRADECIMIENTOS

*En nuestra vida tenemos muchas personas que sin ellas nuestro existir no tendría sentido, a esas personas mis respetos y agradecimientos por ayudarme a alcanzar esta meta.*

*Primero agradecer a mi abuela Mima por ser mi ejemplo a seguir y por traerme de su mano hasta donde estoy hoy, por llegar a este mundo y permitirme ser su nieta preferida, por brindarme todo el amor y la educación del mundo, por ser la mujer ejemplar que ha sido para mí, no tengo palabras para describir a este gran amor de mi vida, por todo gracias Mima.*

*A mi mamá por aguantar mis malcriadeces y estar en los momentos más difíciles de mi vida y ayudarme a salir de ellos, por darme todo su apoyo incondicional, para que hoy pueda convertirme en lo que soy hoy Ingeniera en Ciencias Informáticas, a ella gracias por su amor, dedicación y saberme aconsejar cuando más lo necesitaba, a ella gracias, sin ti esto no hubiera sido posible, gracias mamá.*

*A mi padrastro Pedro, por ser mi papá de corazón y no un simple padrastro, por toda la ayuda que me ha dado en cada momento que lo necesitaba, por defenderme de las peleas de mi mamá, y por tener tanta paciencia con ella, por malcriarme tanto y darme todos mis gustos, por permitirme ser la hija hembra que nunca tuvo, a él gracias por existir y aparecer en nuestras vidas, por ser la mejor persona que he conocido en el mundo, gracias Pedri.*

*A mi tía María por ser mi otra abuela, por apoyarme tanto, por dejarme estar con ella todos los fines de semana desde que estaba en la primaria, porque me encantaba estar en su casa, y por permitirme dormir con ella desde que vine al mundo, a esta tía maravillosa mil gracias por cuidar de mí y por ser el alma gemela de mi abuela, estar con ella cuando yo no estoy.*

*A mi familia, a mi tía Luli, a su esposo Ráidel, a mi prima Yakima, a mis primos y tíos que de una forma u otra me apoyaron para que pudiera cumplir mis sueños.*

*A mi primito David de 8 años por ser mi motor impulsor para llegar a esta meta y mañana ser su guía y ejemplo y se convierta en un excelente profesional, por ser el segundo amor de vida, por sacar tanto amor de su cuerpo pequeñito y dármelo cada vez que estamos juntos, te quiero David, gracias por llegar a este mundo.*

*A Daniel por llegar en el momento más importante de mi vida, para brindarme su más sincero amor y apoyo incondicional, en tan poco tiempo se convirtió en unas de las personas más importantes de mi vida y del logro que hoy he alcanzado, a él muchas gracias por existir y llamar a mi casa equivocadamente, llegaste justo cuando lo necesitaba, Te quiero.*

# *Agradecimientos*

---

*A su familia por apoyarme desde el primer momento en que me conocieron, a su mamá Baby, a su papá Vladimr y a su abuelita Nancy, gracias por quererme tanto y darme tantas fuerzas de que todo iba a salir bien, gracias por dejarme entrar a su hogar y hacerme sentir otro miembro más de este, gracias.*

*Y a mis mejores amigos, a Rey, eres y serás siempre una persona importante en mi vida, por apoyarme, ayudarme, aconsejarme, y darme todas las fuerzas del mundo durante toda la carrera, por ser un excelente amigo, a él gracias, Te quiero.*

*A Iri, mi chiquitica, por ser tan dulce y delicada, en tan poco tiempo se coló en mi corazón convirtiéndose en mi mejor amiga, la quiero y le agradezco todo su amor y cariño y por aguantarme algunas de mis pesadeces, gracias amiga.*

*A Michel Sariol Fernández, que casi nunca está conmigo, porque es un chico que atiende a muchas chicas, pero no sé cómo se las arregla para estar justo a mi lado, en el momento que más lo necesito, ¿muchacho de dónde tu saliste?, gracias Miche.*

*A mi compañera de tesis por decirme Yanita todo va a salir bien, calma y respira profundo, gracias por relajarme en esos momentos.*

*Y al último que no es el menos importante, todo lo contrario, lo dejo de último porque considero que es el eslabón fundamental de este logro, a mi queridísimo tutor, para el mis más sinceros agradecimientos, por apoyarnos durante todo este tiempo, por dejar robarle horas de su sueño, por estar ahí siempre que lo necesitábamos y por darme fuerzas siempre que me estresaba, con sus palabras mágicas siempre me ponía nuevamente mi mundo en su lugar "Apulula y Apululita ustedes saldrán bien".*

*A todos los que de una forma u otra aportaron su granito de arena, para que hoy esto fuera realidad, gracias a todos.*

*Yani.*

*Agradezco a mis padres por todo el amor, apoyo y confianza que me brindan siempre, por estar siempre que los necesite, por creer en mí más que yo misma, los quiero mucho.*

*A mi hermano por ser mi ejemplo a seguir, por ayudarme en todo momento durante todo este tiempo, por él me esforcé para poder estar a su altura.*

*A mis tíos Noraida y Osmany por ser mis segundos padres, por estar ahí en ausencia de los míos, por guiarme en esta travesía, a ellos mil gracias.*

*A mi familia por enseñarme sobre el respeto, amor y la unidad familiar, doy gracias por la familia tan linda que tengo que siempre nos mantengamos unidos apoyándonos unos a los otros.*

# *Agradecimientos*

---

*A Alexei por su amor incondicional, paciencia, respeto, cariño, preocupación, atención, por dejarme entrar en su vida y hacer de la mía un cuento de hadas.*

*A Mireya y Basiliano, por recibirme y tratarme como la hija que nunca tuvieron.*

*A mi grupo 3507 por todas las aventuras vividas, por ser amigos incondicionales, por ser mis compañeros de viaje en esta travesía que fue la universidad, los quiero mucho, no va a hacer fácil olvidarme de algunos de ustedes.*

*A mi compañera de tesis Yanirosa por mantenerse de pie junto a mí en todo momento y nunca rendirse, por planificar bien nuestras actividades productivas y por ser la guía en el desarrollo de nuestra tesis.*

*A nuestro queridísimo tutor por defendernos en todo momento y contar con él siempre que lo necesitábamos.*

*A Yoslenys, Taimé y Raydel Zumeta por haberme ayudado directamente en la realización de este trabajo.*

*A mis compañeras de cuarto Laura, Delisle, Carmen, Nicole y Oneysi por ser mi familia aquí en la UCI, las quiero mucho, sé que las voy a extrañar cuando ya no estén.*

*A mi gente del baile y de la gimnasia por todos los momentos gloriosos vivido y haber compartido con ellos los más espectaculares escenarios aquí en La Habana, las noches de fiestas, las largas y dolorosas horas de ensayos, los quiero a todos, gracias chicos por esto.*

*Por último agradezco a un grupo de personas que formaron parte de mi vida al inicio de esta travesía y que en su momento significaban todo para mí, me enseñaron que las personas tenemos el derecho a equivocarnos y de reparar nuestros errores, eso me hizo más fuerte y a ser una persona decidida, a ellos los recuerdo con mucho cariño.*

*Yei.*

## **DEDICATORIA**

*Dedico con todo mi corazón este trabajo de diploma a mi abuela Mima, que ha sido y será por siempre mi guía, mi luz al final del camino, por darme la educación más perfecta del mundo, por estar a mi lado en todos los momentos de mi vida, por formarme en la mujer que soy hoy, por sus preciados consejos, por darme todo el amor del mundo y por convertirse en lo que más amo en este mundo. Aquí le dedico la meta que alcanzamos las dos juntas y nuestros sueños realidad. Te quiero Mucho.*

*Yani.*

*Dedico la realización de este trabajo a mi mamá y a mi papá por ser mi razón de ser y que por ellos me levanto todos los días a dar lo mejor de mí.*

*Yei.*



# Resumen

## RESUMEN

En la actualidad, a las organizaciones les resulta de vital importancia contar con información actualizada en cuanto al estado en el que se encuentra la misma. Esta constante necesidad ha provocado que muchas de ellas se encuentren inmersas en la informatización de los procesos de control de la entidad con la intención de agilizar los mismos. Contar con información actualizada, controlada y disponible sobre el desarrollo de la entidad que refleje el cumplimiento de los objetivos estratégicos planificados, es fundamental para una empresa. El presente trabajo de diploma describe el desarrollo del Módulo Control Interno para el Sistema de Administración y Economía de la Facultad 3 de la Universidad de las Ciencias Informáticas, tomando como base los principios, componentes y normas emitidas en la Resolución 60 de la Contraloría General de la República de Cuba. La propuesta de solución desarrollada permite realizar el proceso de control interno en la Facultad 3. Con la utilización de este módulo se obtendrá información fiable, refiriéndose a cualquier tipo de información como su posición y desempeño de cada una de las áreas de la facultad, elevándose la disponibilidad y el control de dicha información.

**PALABRAS CLAVE:** control interno, información, resolución 60 de la contraloría general de la república de Cuba.

## ÍNDICE

<b>INTRODUCCIÓN .....</b>	<b>1</b>
<b>CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA.....</b>	<b>5</b>
1.1 Marco Teórico. Conceptos y definiciones .....	5
1.2 Análisis de las soluciones existentes.....	7
1.3 Metodología de desarrollo de software AUP .....	12
1.4 Estudio de herramientas y tecnologías .....	13
1.5 Conclusiones parciales .....	17
<b>CAPÍTULO 2. MODELADO DEL NEGOCIO, ANÁLISIS Y DISEÑO DE LA SOLUCIÓN.....</b>	<b>18</b>
2.1 Modelado del negocio .....	18
2.1.1 Procesos del CI .....	18
2.1.3 Modelo conceptual .....	21
2.2 Requisitos .....	23
2.2.1 Técnicas para la captura de requisitos funcionales.....	23
2.2.2 Requisitos funcionales del sistema .....	23
2.2.3 Descripción de requisitos funcionales del sistema .....	25
2.2.4 Requisitos no funcionales del sistema .....	27
2.2.5 Técnicas para la validación de requisitos.....	28
2.3 Análisis y diseño .....	29
2.3.1 Arquitectura del sistema .....	29
2.3.2 Diagrama de clases del diseño.....	31
2.3.3 Patrones de software.....	32
2.3.5 Modelo de datos.....	38
2.3.6 Validación del proceso de diseño .....	39
2.4 Conclusiones parciales .....	44
<b>CAPÍTULO 3. IMPLEMENTACIÓN Y VALIDACIÓN DE LA SOLUCIÓN .....</b>	<b>46</b>
3.1 Implementación .....	46
3.1.1 Diagrama de componentes.....	46

3.1.2 Diagrama de despliegue.....	47
3.1.3 Estándares de codificación.....	48
3.2 Pruebas de software .....	50
3.2.1 Pruebas internas .....	50
3.2.2 Pruebas de Aceptación.....	57
3.4 Conclusiones parciales .....	62
<b>CONCLUSIONES .....</b>	<b>63</b>
<b>RECOMENDACIONES .....</b>	<b>64</b>
<b>REFERENCIAS.....</b>	<b>65</b>
<b>GLOSARIO .....</b>	<b>68</b>
<b>ANEXOS.....</b>	<b>69</b>

## ÍNDICE DE TABLAS

Tabla 1. Comparación entre los Sistemas de CI. Fuente: Elaboración propia .....	11
Tabla 2. Descripción de proceso de negocio Adecuar Guía de Autocontrol. Fuente: Elaboración propia .....	20
Tabla 3. Requisitos Funcionales del Módulo CI. Fuente: Elaboración propia .....	25
Tabla 4. Historia de Usuario Adecuar Guía de Autocontrol. Fuente: Elaboración propia .....	26
Tabla 5. Requisitos No Funcionales del Módulo CI. Fuente: Elaboración propia .....	28
Tabla 6. Descripción de las tablas de la BD. Fuente: Elaboración propia .....	39
Tabla 7. Atributos de calidad de la métrica TOC. Fuente: Elaboración propia .....	40
Tabla 8. Rango de valores para medir la afectación de los atributos de calidad de la métrica TOC. Fuente: Elaboración propia .....	40
Tabla 9. Atributos de calidad de la métrica RC. Fuente: Elaboración propia .....	42
Tabla 10. Rango de valores para medir la afectación de los atributos de calidad de la métrica RC. Fuente: Elaboración propia .....	43
Tabla 11. Caso de Prueba Gestionar adecuación de la Guía de Autocontrol. Fuente: Elaboración propia. ....	52
Tabla 12. Caminos básicos del flujo del método Crear adecuación de la Guía de Autocontrol. Fuente: Elaboración propia .....	56
Tabla 13. Caso de Prueba del camino básico 1. Fuente: Elaboración propia .....	57
Tabla 14. Resultado de la validación de las variables disponibilidad y control de la información. Fuente: Elaboración propia .....	59
Tabla 15. Cuadro lógico de ladov. Fuente: Elaboración propia .....	60
Tabla 16. Resultado de aplicación de la técnica ladov. Fuente: Elaboración propia .....	60
Tabla 17. Resultado de aplicación de la técnica ladov. Fuente: Elaboración propia .....	61
Tabla 18. Descripción del proceso de negocio Aplicar Guía de Autocontrol. Fuente: Elaboración propia .....	70
Tabla 19. Descripción del proceso de negocio Realizar seguimiento del Plan de Medidas. Fuente: Elaboración propia .....	72
Tabla 20. Concepto Área. Fuente: Elaboración propia .....	74
Tabla 21. Concepto Usuario. Fuente: Elaboración propia .....	74
Tabla 22. Concepto Supervisión. Fuente: Elaboración propia .....	74
Tabla 23. Concepto Informe. Fuente: Elaboración propia .....	76
Tabla 24. Concepto Plan de Medidas. Fuente: Elaboración propia .....	77
Tabla 25. Concepto Seguimiento. Fuente: Elaboración propia .....	77
Tabla 26. Concepto Mejoras. Fuente: Elaboración propia .....	78
Tabla 27. Concepto Medidas. Fuente: Elaboración propia .....	78
Tabla 28. Concepto Deficiencias. Fuente: Elaboración propia .....	79
Tabla 29. Concepto Guía de Autocontrol. Fuente: Elaboración propia .....	79
Tabla 30. Concepto Componente. Fuente: Elaboración propia .....	80
Tabla 31. Concepto Norma. Fuente: Elaboración propia .....	80
Tabla 32. Concepto Ítem. Fuente: Elaboración propia .....	81
Tabla 33. Concepto Aplicación. Fuente: Elaboración propia .....	81
Tabla 34. Concepto Adecuación. Fuente: Elaboración propia .....	82
Tabla 35. Historia de Usuario Modificar Adecuación de la Guía de Autocontrol. Fuente: Elaboración propia .....	83
Tabla 36. Historia de Usuario Eliminar Adecuación de la Guía de Autocontrol. Fuente: Elaboración propia .....	84
Tabla 37. Historia de Usuario Adicionar Guía de Autocontrol. Fuente: Elaboración propia .....	85
Tabla 38. Historia de Usuario Modificar Guía de Autocontrol. Fuente: Elaboración propia .....	86
Tabla 39. Historia de Usuario Eliminar Guía de Autocontrol. Fuente: Elaboración propia .....	87
Tabla 40. Historia de Usuario Adicionar Componente. Fuente: Elaboración propia .....	88
Tabla 41. Historia de Usuario Modificar Componente. Fuente: Elaboración propia .....	89
Tabla 42. Historia de Usuario Eliminar Componente. Fuente: Elaboración propia .....	90
Tabla 43. Historia de Usuario Listar Componente. Fuente: Elaboración propia .....	91
Tabla 44. Historia de Usuario Adicionar Norma. Fuente: Elaboración propia .....	92
Tabla 45. Historia de Usuario Modificar Norma. Fuente: Elaboración propia .....	93
Tabla 46. Historia de Usuario Eliminar Norma. Fuente: Elaboración propia .....	94
Tabla 47. Historia de Usuario Listar Norma. Fuente: Elaboración propia .....	95
Tabla 48. Historia de Usuario Adicionar Ítem. Fuente: Elaboración propia .....	96
Tabla 49. Historia de Usuario Modificar Ítem. Fuente: Elaboración propia .....	97
Tabla 50. Historia de Usuario Eliminar Ítem. Fuente: Elaboración propia .....	98
Tabla 51. Historia de Usuario Listar Ítem. Fuente: Elaboración propia .....	99

Tabla 52. Historia de Usuario Aplicar GA. Fuente: Elaboración propia.....	100
Tabla 53. Historia de Usuario Generar Informe de la aplicación de la GA. Fuente: Elaboración propia .....	101
Tabla 54. Historia de Usuario Generar Plan de Medidas. Fuente: Elaboración propia .....	102
Tabla 55. Historia de Usuario Generar Seguimiento del Plan de Medidas. Fuente: Elaboración propia .....	103
Tabla 56. Historia de Usuario Adicionar área. Fuente: Elaboración propia .....	104
Tabla 57. Historia de Usuario Modificar área. Fuente: Elaboración propia .....	105
Tabla 58. Historia de Usuario Eliminar área. Fuente: Elaboración propia .....	106
Tabla 59. Historia de Usuario Adicionar deficiencia. Fuente: Elaboración propia.....	107
Tabla 60. Historia de Usuario Modificar deficiencia. Fuente: Elaboración propia .....	108
Tabla 61. Historia de Usuario Eliminar deficiencia. Fuente: Elaboración propia .....	109
Tabla 62. Historia de Usuario Adicionar medida. Fuente: Elaboración propia .....	110
Tabla 63. Historia de Usuario Modificar medida. Fuente: Elaboración propia .....	111
Tabla 64. Historia de Usuario Eliminar medida. Fuente: Elaboración propia .....	112
Tabla 65. Historia de Usuario Adicionar mejora. Fuente: Elaboración propia.....	113
Tabla 66. Historia de Usuario Modificar mejora. Fuente: Elaboración propia .....	114
Tabla 67. Historia de Usuario Eliminar mejora. Fuente: Elaboración propia.....	115
Tabla 68. Historia de Usuario Autenticar usuario. Fuente: Elaboración propia.....	116
Tabla 69. Historia de Usuario Adicionar usuario. Fuente: Elaboración propia.....	117
Tabla 70. Historia de Usuario Modificar usuario. Fuente: Elaboración propia .....	118
Tabla 71. Historia de Usuario Eliminar usuario. Fuente: Elaboración propia.....	119
Tabla 72. Caso de Prueba Gestionar Guía de Autocontrol. Fuente: Elaboración propia.....	125
Tabla 73. Caso de Prueba Gestionar Componente. Fuente: Elaboración propia.....	126
Tabla 74. Caso de Prueba Gestionar Norma. Fuente: Elaboración propia.....	127
Tabla 75. Caso de Prueba Gestionar Ítem. Fuente: Elaboración propia.....	128
Tabla 76. Caso de Prueba Gestionar Área. Fuente: Elaboración propia.....	129
Tabla 77. Caso de Prueba Gestionar Usuario. Fuente: Elaboración propia.....	130
Tabla 78. Caso de Prueba Gestionar Deficiencia. Fuente: Elaboración propia.....	131
Tabla 79. Caso de Prueba Gestionar Medida. Fuente: Elaboración propia.....	132
Tabla 80. Caso de Prueba Gestionar Mejora. Fuente: Elaboración propia.....	133
Tabla 81. Caso de Prueba del camino básico 2. Fuente: Elaboración propia .....	134
Tabla 82. Caso de Prueba del camino básico 3. Fuente: Elaboración propia .....	134
Tabla 83. Caso de Prueba del camino básico 4. Fuente: Elaboración propia .....	135
Tabla 84. Caso de Prueba del camino básico 5. Fuente: Elaboración propia .....	135

## ÍNDICE DE FIGURAS

Figura 1. Fases e iteraciones de AUP (Rodríguez, 2014).....	12
Figura 2. Diagrama de proceso de negocio Adecuar Guía de Autocontrol. Fuente: Elaboración propia .....	21
Figura 3. Modelo conceptual del Módulo CI. Fuente: Elaboración propia .....	22
Figura 4. Arquitectura interna en el marco de trabajo Symfony 2 (Aguiluz, 2013). .....	30
Figura 5. Arquitectura MVC en el escenario Gestionar adecuación de la Guía de Autocontrol. Fuente: Elaboración propia.....	31
Figura 6. Diagrama de clases de diseño. Escenario Gestionar Adecuación de la Guía de Autocontrol. Fuente: Elaboración propia. ....	32
Figura 7. Organización propuesta por Symfony2. Fuente: Elaboración propia. ....	35
Figura 8. Organización propuesta por Symfony2. Clases Repository. Fuente: Elaboración propia. ....	35
Figura 9. Organización propuesta por Symfony2. Entidades. Fuente: Elaboración propia. ....	36
Figura 10. Organización propuesta por Symfony2. Vistas. Fuente: Elaboración propia. ....	37
Figura 11. Organización propuesta por Symfony2. Controlador en Datos Principales. Fuente: Elaboración propia.....	37
Figura 12. Modelo de Datos. Fuente: Elaboración propia.....	38
Figura 13. Representación de la evaluación de la métrica TOC para el atributo responsabilidad. Fuente: Elaboración propia.....	41
Figura 14. Representación de la evaluación de la métrica TOC para el atributo complejidad. Fuente: Elaboración propia.....	41
Figura 15. Representación de la evaluación de la métrica TOC para el atributo reutilización. Fuente: Elaboración propia.....	41
Figura 16. Representación de la evaluación de la métrica RC en el atributo Acoplamiento. Fuente: Elaboración propia.....	43
Figura 17. Representación de la evaluación de la métrica RC en el atributo complejidad de mantenimiento. Fuente: Elaboración propia. ....	43
Figura 18. Representación de la evaluación de la métrica RC en el atributo cantidad de pruebas. Fuente: Elaboración propia.....	44
Figura 19. Representación de la evaluación de la métrica RC en el atributo reutilización. Fuente: Elaboración propia.....	44
Figura 20. Diagrama de componentes Fuente: Elaboración propia .....	47
Figura 21. Diagrama de despliegue. Fuente: Elaboración propia.....	47
Figura 22. Resultado de no conformidades de la prueba de caja negra. ....	53
Figura 23. Código fuente del método Crear adecuación de la Guía de Autocontrol. Fuente: Elaboración propia.....	55
Figura 24. Grafo de flujo del método Crear adecuación de la Guía de Autocontrol. Fuente: Elaboración propia.....	55
Figura 25. Diagrama de proceso de negocio Aplicar Guía de Autocontrol. Fuente: Elaboración propia.....	71
Figura 26. Diagrama del proceso de negocio Realizar seguimiento del Plan de Medidas. Fuente: Elaboración propia.....	73
Figura 27. Diagrama de clases de diseño. Escenario Gestionar Componente. Fuente: Elaboración propia. ....	119
Figura 28. Diagrama de clases de diseño. Escenario Gestionar Norma. Fuente: Elaboración propia.....	120
Figura 29. Diagrama de clases de diseño. Escenario Gestionar Ítem. Fuente: Elaboración propia. ....	120
Figura 30. Diagrama de clases de diseño. Escenario Gestionar Usuario. Fuente: Elaboración propia. ....	121
Figura 31. Diagrama de clases de diseño. Escenario Aplicar Guía de Autocontrol. Fuente: Elaboración propia.....	122
Figura 32. Diagrama de clases de diseño. Escenario Generar Plan de Medidas. Fuente: Elaboración propia. ....	122
Figura 33. Diagrama de clases de diseño. Escenario Gestionar Medida. Fuente: Elaboración propia.....	123
Figura 34. Diagrama de clases de diseño. Escenario Gestionar Mejora. Fuente: Elaboración propia. ....	123
Figura 35. Diagrama de clases de diseño. Escenario Gestionar Deficiencia. Fuente: Elaboración propia...	124
Figura 36. Diagrama de clases de diseño. Escenario Gestionar Área. Fuente: Elaboración propia. ....	124
Figura 39. Diagrama de clases de diseño. Escenario Generar Informe. Fuente: Elaboración propia.....	125

## **INTRODUCCIÓN**

La contemporaneidad vislumbra una acelerada aparición de información y transmisión de la misma. Este amplio flujo incluye adecuadas estrategias organizativas, dado el alto grado de complejidad de la realidad y los riesgos que origina. En este sentido, las nuevas tecnologías solventan algunas de las dificultades resultantes de dicha realidad. Las industrias y empresas, en aras de optimizar sus rendimientos, laboran en ordenar sus procesos controlando internamente la entidad.

El Control Interno (CI), a nivel mundial, es un mecanismo cuyo uso es muy asistido en la mayoría de las empresas para ayudar a que la entidad logre sus metas de rentabilidad, rendimiento y prevención de la pérdida de recursos. Este proceso constituye un punto clave en la obtención de información financiera fiable, refiriéndose a cualquier tipo de información como la posición y desempeño financiero de dicha entidad. Proporciona, además, elementos de juicio para valorar el comportamiento futuro de los flujos de efectivo de la organización y de esta forma lograr el cumplimiento de las leyes y normas que son aplicables a cada uno de los componentes que son auditables. Determina el cumplimiento de los objetivos estratégicos planificados en las entidades, a partir de identificar y elucidar los riesgos asociados con cada actividad y proceso, sustentado dicho criterio en el cuidado de los activos, los intereses que se persiguen y previsión de fraudes y riesgos innecesarios(Cra. Daniela Biasco, 2008).

Según César Gaviria Trujillo, en la Ley 87 de 1993 de Santa Fé, Bogotá, se entiende por CI el sistema integrado por el esquema de organización y el conjunto de los planes, métodos, principios, normas, procedimientos y mecanismos de verificación y evaluación adoptados por una entidad. Posee como fin procurar que todas las actividades, operaciones y actuaciones, la administración de la información y los recursos, se realicen de acuerdo con las normas constitucionales y legales vigentes dentro de las políticas trazadas por la dirección, atendiendo a las metas u objetivos previstos(Trujillo, 1993.).

En Cuba el CI ha sufrido varias transformaciones desde los inicios del triunfo de la Revolución debido a los diversos sistemas económicos aplicados en el país. La Resolución 60 de fecha 1ro de marzo de 2011 de la Contraloría General de la República, establece las normas y principios básicos de obligada observancia para los sujetos de las acciones de auditorías, supervisión y control de ese Órgano, constituyendo un modelo estándar del Sistema de Control Interno (SCI). El SCI es de preocupación para los directivos de las entidades, en mayor o menor grado, con diferentes enfoques y terminologías. Esto ha permitido que con el tiempo se hayan planteado diferentes concepciones acerca del mismo, sus principios así como elementos que se deben conocer e instrumentar en la entidad cubana actual. Siendo necesario brindarle a los cuadros,

dirigentes, funcionarios y demás trabajadores, un instrumento de trabajo que le permita implementar en sus entidades el SCI(Fuentes, y otros, 2012).

La solidificación del CI para la reestructuración cultural, económica y social que tiene lugar en Cuba reviste una trascendental significación en dicho escenario. Su dinamización alcanza las herramientas más sofisticadas, en aras de lograr un ordenamiento consecuente con la política del país. La Universidad de las Ciencias Informáticas (UCI) tiene entre sus líneas investigativas-productivas la creación de software y/o aplicaciones que agilicen diversos procesos y apoyen el ordenamiento de la información en la sociedad cubana.

El Vicedecanato de Economía y Administración de la Facultad 3 de la UCI dirige el proceso del CI de la facultad, donde se controlan cinco componentes definidos en la Resolución 60 de la Contraloría General de la República: Ambiente de Control, Gestión y Prevención de Riesgos, Actividades de Control, Información y Comunicación y Supervisión y Monitoreo. Los componentes generan un gran cúmulo de información en las diferentes áreas, relacionadas con el control por parte de los directivos sobre sus subordinados, la accesibilidad y supervisión de las acciones laborales que realizan. En la actualidad el CI se realiza de forma manual, utilizando como soporte el papel lo que provoca lentitud en el proceso, imprecisiones e incertidumbre. Se muestra elevada preocupación en los directivos porque la información presenta problemas de difícil acceso, desactualización, deterioro, pérdida de datos y dificultad en la obtención de reportes necesarios sobre el funcionamiento de la organización por lo que provoca dificultad en la toma de decisiones.

En aras de delimitar el conjunto de soluciones que propicia la problemática antes mencionada se plantea como **problema a resolver**: ¿cómo gestionar la información del proceso de CI de la Facultad 3 de manera que se eleve la disponibilidad y control de la información?

A partir del problema planteado se define como **objeto de estudio** de la investigación: el proceso de CI, teniendo como **objetivo general**: desarrollar el módulo CI para el Sistema de Administración y Economía de la Facultad 3, de manera que se eleve la disponibilidad y el control de la información.

Para dar cumplimiento a lo anteriormente planteado se trazaron los siguientes **objetivos específicos**:



1. Analizar los elementos, mecanismos y sistemas de CI para las empresas.
2. Modelar las funcionalidades del proceso de CI de la Facultad 3.
3. Diseñar una propuesta de solución de los requisitos para el módulo CI.
4. Implementar una herramienta con los componentes de software del proceso de CI.
5. Validar la solución propuesta mediante la realización de pruebas de software.
6. Validar las variables de la investigación.

El **campo de acción** abarca el proceso de CI de la Facultad 3, por tanto se define la siguiente **idea a defender**: con la implementación del módulo CI para el Sistema de Administración y Economía de la Facultad 3 se elevará la disponibilidad y el control de la información de dicho proceso.

Como **tareas de la investigación** se definen las siguientes:

1. Análisis de los componentes asociados al CI de la Facultad 3: Ambiente de Control, Gestión y Prevención de Riesgos, Actividades de Control, Información y Comunicación y Supervisión y Monitoreo.
2. Levantamiento de requisitos de los componentes: Ambiente de Control, Gestión y Prevención de Riesgos, Actividades de Control, Información y Comunicación y Supervisión y Monitoreo.
3. Caracterización de la metodología, las tecnologías y las herramientas a utilizar para el desarrollo de la solución.
4. Diseño de la estructura y comportamiento de los componentes: Ambiente de Control, Gestión y Prevención de Riesgos, Actividades de Control, Información y Comunicación y Supervisión y Monitoreo.
5. Diseño del modelo de datos.
6. Diseño del diagrama de despliegue del sistema.
7. Validación del diseño propuesto.
8. Implementación de los componentes diseñados.
9. Validación de la implementación realizada.
10. Validación de las variables de la investigación

Para la realización de las tareas antes mencionadas se emplearon métodos científicos. Los cuales permiten abordar la realidad, estudiar la sociedad y el pensamiento con el propósito de descubrir su esencia. Se clasifican en teóricos y empíricos.

Los **métodos teóricos** posibilitan las condiciones para buscar las características triviales de la realidad, permiten explicar los hechos y profundizar en las principales relaciones y cualidades de los fenómenos, hechos y procesos.

- ✓ **Análisis Histórico – Lógico:** este método permitió realizar un estudio acerca de cómo se lleva a cabo el proceso de CI en las empresas desde su surgimiento hasta la actualidad, tanto en el ámbito nacional como internacional, además de los sistemas informáticos existentes que realizan este proceso.
- ✓ **Modelación:** se utilizó para la elaboración de los artefactos generados en los flujos de trabajo correspondientes al proceso de CI que se realiza en la facultad 3.
- ✓ **Análisis documental:** permitió el estudio de la bibliografía concretando la de interés para la investigación del proceso CI con el propósito de determinar el sistema de conceptos y categorías del mismo. Permitted extraer las relaciones esenciales del objeto investigado para descubrir y estudiar nuevas cualidades, relaciones, principios o leyes del CI.

Los **métodos empíricos** revelan, describen y explican las características y relaciones esenciales del objeto basando su contenido en la experiencia.

- ✓ **Entrevista:** fueron realizadas a la Vicedecana de Administración y Economía de la Facultad 3 con el objetivo de reunir información relevante para el desarrollo del módulo CI. Las entrevistas se realizaron en el horario de la tarde para fomentar la comunicación con el cliente.

Para lograr la claridad y comprensión de los contenidos de la investigación realizada se ha estructurado el documento en tres capítulos que cubren la fundamentación teórica, características del sistema y la implementación y validación de la solución, además de las conclusiones, recomendaciones, referencias bibliográficas, glosario de términos y anexos.

**Capítulo 1. Fundamentación teórica:** en este capítulo se explica la metodología de desarrollo, los lenguajes usados, las herramientas utilizadas para el desarrollo de la aplicación e incluye además un análisis comparativo de algunos Sistemas de CI existentes en el mundo para identificar posibles fortalezas que puedan ser reutilizadas en la solución del problema planteado.

**Capítulo 2. Modelado del negocio, Análisis y diseño de la solución:** en este capítulo se incluye un análisis del funcionamiento actual del negocio y características del módulo CI. Se identifican, describen y validan los principales requisitos del sistema. Se exponen los detalles relacionados con el diseño así como la utilización de los patrones de diseño, arquitectónico y se aplican métricas para la validación del diseño propuesto.

**Capítulo 3. Implementación y validación de la solución:** en este capítulo se detalla el resultado final de la solución, características esenciales así como el aporte práctico del módulo CI. Se muestran los elementos por los que se validó la solución implementada.

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

### Introducción

En el presente capítulo se abordarán aspectos fundamentales que permiten describir teóricamente el módulo a implementar. Se realizará la revisión de la literatura acerca de sistemas informáticos del ámbito nacional e internacional que realizan el CI identificando características valiosas para la definición del módulo. Se describirán los componentes definidos en la Resolución 60 para el CI. Se describirán la metodología de desarrollo, las tecnologías, lenguajes y herramientas a utilizar en la presente investigación.

### 1.1 Marco Teórico. Conceptos y definiciones

Los conceptos que se definen en este capítulo están relacionados con el dominio del problema, aportando un breve entendimiento de todos los aspectos que se tratan posteriormente.

#### Control Interno

El Instituto Mexicano de Contadores Públicos plantea: *el CI comprende el plan de organización con los métodos y procedimientos que en forma coordinada se adoptan en un negocio para la protección de sus activos, la obtención de información financiera correcta y segura, la promoción de eficiencia de operación y la adhesión a políticas prescritas por la dirección*(ICMP, 1982).

En Colombia, la propuesta del tratadista contable Yanel Blanco Luna plantea: *el CI es el plan de organización y todos los métodos y procedimientos que adopta la administración de una entidad para ayudar al logro del objetivo administrativo. Asegura, en cuanto sea posible, la conducción ordenada y eficiente de su negocio. Incluye la adherencia a las políticas administrativas, la salvaguarda de los activos. Previene y detecta fraudes y errores, la corrección de los registros contables y preparación oportuna de la información financiera contable* (Blanco, 1988).

En el mes de octubre de 1992 en el seno del XIV Congreso Mundial de Contadores celebrado en Washington se plantea: *el CI es un proceso, ejecutado por la junta directiva o consejo de administración de una entidad, por su grupo directivo (gerencia) y por el resto del personal, diseñado específicamente para proporcionarles seguridad de conseguir en la empresa las tres siguientes categorías de objetivos:*

- ✓ *Efectividad y eficiencia en las operaciones.*
- ✓ *Suficiencia y confiabilidad de la información financiera.*
- ✓ *Cumplimiento de las leyes y regulaciones aplicables*(CTCP, 1999).

A partir del estudio de las definiciones del CI se considera que el mismo está dirigido a la organización, obtención de información financiera correcta y segura y la conducción ordenada y eficiente de su empresa, pero están enfocados a la realidad donde se desenvuelven, que es diferente al sistema político-económico de las empresas cubanas. Teniendo en cuenta las

particularidades de la Facultad 3 de la UCI se toma como base la definición de CI, por la Resolución 60/11 de la Contraloría General de la República:

*El proceso integrado a las operaciones con un enfoque de mejoramiento continuo, extendido a todas las actividades inherentes a la gestión, efectuado por la dirección y el resto del personal; se implementa mediante un sistema integrado de normas y procedimientos, que contribuyen a prever y limitar los riesgos internos y externos, proporciona una seguridad razonable al logro de los objetivos institucionales y una adecuada rendición de cuentas (Contraloría General De La República, 2011).*

## **Componentes del CI**

El CI analiza cinco componentes interrelacionados, que se derivan de la forma en que la administración maneja el ente, y están integrados a los procesos administrativos, los cuales se clasifican como:

- ✓ Ambiente de Control
- ✓ Evaluación de Riesgos
- ✓ Actividades de Control
- ✓ Información y Comunicación
- ✓ Supervisión y Monitoreo

## **Ambiente de control**

El componente Ambiente de control se define como las pautas para el funcionamiento legal y armónico de los órganos, organismos, organizaciones y demás entidades, para el desarrollo de las acciones. Refleja la actitud asumida por la máxima dirección y el resto del colectivo en relación con la importancia del SCI. Se considera la base de los demás componentes. Conforman el conjunto de buenas prácticas y documentos referidos a la constitución de la organización, al marco legal de las operaciones aprobadas, a la creación de sus órganos de dirección y consultivos, a los procesos, sistemas, políticas, disposiciones legales y procedimientos; lo que tiene que ser del dominio de todos los implicados y estar a su alcance (Contraloría General De La República, 2011).

## **Gestión y prevención de riesgos**

El componente Gestión y prevención de riesgos define las bases para la identificación y análisis de los riesgos que enfrentan las organizaciones para alcanzar sus objetivos. Una vez clasificados los riesgos en internos y externos, por procesos, actividades y operaciones, y evaluadas las principales vulnerabilidades, se determinan los objetivos de control y se conforma el Plan de Prevención de Riesgos para definir el modo en que deberán gestionarse. Existen riesgos que están regulados por disposiciones legales de los organismos rectores, los que se gestionan según los modelos de administración previstos (Contraloría General De La República, 2011).

## **Actividades de control**

El componente Actividades de control establece las políticas, disposiciones legales y procedimientos de control para gestionar y verificar la calidad de la gestión para el cumplimiento de los objetivos y misión de las organizaciones. Las actividades de control son aplicables a las operaciones de todo tipo, las que tributan a la fiabilidad de la información financiera y al cumplimiento de las disposiciones legales correspondientes al marco de desarrollo de la actividad, así como a la comprobación de las transacciones u operaciones económicas que le dan cobertura a los objetivos y metas en cuanto a su exactitud, autorización y registro contable conforme a las normas cubanas establecidas al efecto, con un enfoque de mejoramiento continuo(Contraloria General De La República, 2011).

## **Información y comunicación**

El componente Información y comunicación precisa que las entidades deben disponer de información oportuna, fiable y definir su sistema de información adecuado a sus características. Genera datos, documentos y reportes que sustentan los resultados de las actividades operativas, financieras y relacionadas con el cumplimiento de los objetivos, metas y estrategias, con mecanismos de retroalimentación y la rendición transparente de cuentas. La información debe protegerse y conservarse según las disposiciones legales vigentes(Contraloria General De La República, 2011).

## **Supervisión y monitoreo**

El componente Supervisión y monitoreo está dirigido a la detección de errores e irregularidades que no fueron detectados con las actividades de control, permitiendo realizar las correcciones y modificaciones necesarias. Estas se realizan mediante dos modalidades de supervisión: actividades continuas, que son aquellas que, incorporadas a las actividades normales, generan respuestas dinámicas, entre ellas el seguimiento sistemático realizado por las diferentes estructuras de dirección y evaluaciones puntuales que son ejecutadas por los responsables de las áreas, por auditorías internas y externas(Contraloria General De La República, 2011).

Las definiciones anteriormente analizadas sentaron las bases para un entendimiento de los aspectos claves que maneja el proceso de CI y tenerlos en cuenta durante el desarrollo de la investigación. A partir del estudio de las características del CI se realizará una búsqueda de sistemas informáticos nacionales e internacionales que cumplan con los aspectos antes mencionados de alguna forma.

## **1.2 Análisis de las soluciones existentes**

En este epígrafe se identificaron las soluciones homólogas existentes que evalúan de alguna forma el CI en sistemas de software tanto en el ámbito internacional como nacional.

## **Sistemas de CI**

## **MindManager**

MindManager<sup>1</sup> es un programa comercial de mapas mentales (mind map por su descripción en inglés) desarrollado por Mindjet Corporation (Compañía Mindjet). Es un gestor de proyectos profesional basado en mapas de ideas, esquemas gráficos que permiten simplificar y entender datos complejos. Los mapas mentales creados en MindManager se basan en el método de mapeo de la mente definido por Tony Buzan. La última versión, MindManager 8, está disponible solamente para Microsoft Windows, aunque las versiones anteriores están disponibles para Mac OS X y los archivos creados en las versiones más recientes son compatibles con ambas plataformas. Mindjet Corporation en 2008 presentó Mindjet Connect, un servicio de colaboración en línea con el almacenamiento centralizado. Connect es accesible con MindManager 8 en Windows y Web a través de MindManager, basado en un cliente de Flash8.

Los mapas pueden extraer datos de Microsoft Excel y Outlook, y pueden ser exportados a Microsoft Word, PowerPoint, Visio y Project, así como a páginas web HTML y a documentos en formato PDF (López, 2012).

### Ventajas

- ✓ Fácil de usar.
- ✓ Buena selección de plantillas y temas.
- ✓ Admite toda clase de contenido.
- ✓ Gestión de tareas y tiempo invertido.
- ✓ Múltiples opciones para exportar.

### Desventajas

- ✓ Pocas opciones para importar.
- ✓ Exceso de paneles complementarios.

## **DATEV SINFOPAC Audit**

DATEV SINFOPAC Audit<sup>2</sup> es una aplicación para el trabajo de auditoría desarrollado por profesionales de varios países europeos cuyo objetivo es, además de satisfacer las exigencias de la armonización legislativa de inspiración comunitaria y de la reforma contable basada en las NIIF/NIC<sup>3</sup> y en el nuevo PGC<sup>4</sup>, garantizar la homogeneidad y calidad en el trabajo de los auditores en los diferentes países comunitarios, respetando su idioma y particularidades (Sinfopac, 2014).

El software proporciona:

- ✓ Apoyo en todo el proceso de auditoría con una excelente gestión de la documentación.

---

<sup>1</sup><http://www.mindmanager.softonic.com>

<sup>2</sup><http://www.datevsinfopac.com>

<sup>3</sup> Normas internacionales de contabilidad e información financiera. <http://www.nicniif.org>

<sup>4</sup> Plan general contable. <http://www.plangeneralcontable.com/>

- ✓ Un sistema de referencia y archivo flexible para que el trabajo se haga con calidad, coherencia y rentabilidad, cumpliendo los requisitos legales y comunitarios para ofrecer a los auditores europeos, sean individuales o de un grupo, el intercambio de información.
- ✓ Aporta soluciones para capturar datos, brindar mantenimiento de diversas bases contables(interinos, finales...), planificar el trabajo, facilitar el conocimiento de la entidad, evaluar su CI, la interrelación de riesgos y materialidad, crear, evidenciar y documentar los papeles de trabajo, facilitando las conclusiones en base a fórmulas objetivas que apoyen el juicio profesional del auditor(**Sinfopac, 2014**).

## Funcionalidades

- ✓ Exportación de diferentes formatos (Excel, txt, csv).
- ✓ Asignación de diferentes esquemas estructurales (Planes de cuentas).
  - ✓ Predefinidos según sector, país.
  - ✓ Totalmente personalizables por despacho o por cliente.
- ✓ Ejecución (interinos, finales).
  - ✓ Documentación.
  - ✓ Diferentes bases contables de la información.
  - ✓ Seguimiento del estado de la auditoría en todo momento.
- ✓ Elaboración automatizada de informes.
- ✓ Folleto / Manuales.

## SE Audit

SE Audit<sup>5</sup> es un software que realiza la administración de todas las etapas del proceso de auditoría, desde la planificación y aprobación, hasta la monitorización, sea ella interna (primera parte), de proveedores y clientes (segunda parte), y/o de organismos certificadores (tercera parte). Es un sistema WEB, multiusuario y multidepartamental, que incorpora herramientas de organización, clasificación y búsqueda. (SoftExpert Software for Performance Excellence, 2011).

El software dispone de diversas funcionalidades, entre ellas:

- ✓ Registro de requisitos de auditorías con informaciones que pueden ser impresas y utilizadas por los auditores durante el proceso de auditoría.
- ✓ Registro de criterios de auditoría.
- ✓ Registro de proveedores y personas auditables en auditorías de segunda parte.
- ✓ Registro de clientes y organismos certificadores que realizarán auditorías de segunda y tercera partes.
- ✓ Registro de evidencias identificadas durante la realización de la auditoría.
- ✓ Anexo de cualquier tipo de documento (textos, plantillas, presentaciones, gráficos e imágenes).

---

<sup>5</sup><http://www.softexpert.es>

- ✓ Registro de ocurrencias como buenas prácticas, oportunidades de mejora, observaciones y no conformidades evidenciadas durante la realización de la auditoría.

El software proporciona el Trabajo en equipo, a través de un práctico mecanismo de control de pendientes, denominado Team Workflow (Flujo de trabajo de Equipo), que notifica vía correo el momento exacto, a los responsables por actividades pendientes, exhibe estas pendientes y autoriza el registro de las firmas electrónicas y demás informaciones aplicables a cada etapa del proceso. Este mecanismo asegura la agilidad y el compromiso con el cumplimiento de los plazos en todas las etapas del proceso de auditoría (SoftExpert Software for Performance Excellence, 2011).

## **Meycor COSO AG**

El software Meycor COSO<sup>6</sup> ha sido desarrollado por DATASEC. El producto permite realizar una evaluación del CI según el informe COSO<sup>7</sup>, evaluación de los riesgos y realizar auditorías de las evaluaciones. Está compuesto por tres módulos: Meycor COSO AG - Eval, Meycor COSO AG - Audit y Meycor COSO AG - Web.

Meycor COSO AG - Eval: constituye la parte de recopilación de información de la organización, así como la parte de evaluación de CI y los riesgos (Ibañez, 2008).

Meycor COSO AG - Audit: este módulo permite organizar una auditoría de las evaluaciones de riesgos realizadas en el módulo anterior. El producto permite la creación de proyectos de auditoría por parte de los supervisores que incluyen la definición de los auditores del proyecto, la asignación de riesgos a auditar y la asignación de riesgos a auditores.

Meycor COSO AG - Web: este módulo permite el acceso de usuarios por la Web para contestar cuestionarios generales del informe COSO, ver documentos que tienen disponibles y marcar su grado de acuerdo o desacuerdo con los documentos (Ibañez, 2008).

## **Farola**

Es un sistema basado en plataforma distribuida que permite almacenar datos importantes referentes a la Gestión y Prevención de los Riesgos en todos los Grupos organizacionales, Organizaciones y/o Unidades organizativas, procesos, subprocesos y actividades, asociándolo con los objetivos. Inicia en la detección de estos por cada proceso, teniendo en cuenta las actividades que se vinculan, así como los objetivos que estos pueden afectar.

### **Características**

Se tienen los reportes de las variables relacionadas por diferentes criterios de filtrado. Se obtiene una estadística muy detallada de dichos reportes. Se puede llevar un seguimiento de las acciones planificadas tanto preventivas, correctivas y de mejora con el objetivo de minimizar o erradicar los riesgos que pueden convertirse, con determinada probabilidad, en no conformidades futuras.

---

<sup>6</sup>Informe Meycor COSO. 2008 <http://www.datesec-soft.com>

<sup>7</sup>Comité de Organizaciones Patrocinadoras



Es un programa que no sólo emplea uno de los componentes de la Resolución 60 de Cuba, referida al CI, sino que se vincula también de forma sistémica con la administración por objetivos con enfoque a procesos. Este programa puede ser empleado, además, para la implantación del Sistema de Gestión de la Calidad y del Perfeccionamiento Empresarial en cualesquiera de las organizaciones (Desoft Guantánamo, 2011.).

- ✓ Principales funcionalidades:
- ✓ Registra usuarios con diferentes roles.
- ✓ Registra el Plan de Prevención de indisciplina, ilegalidades y manifestaciones de corrupción.
- ✓ Registra los Grupos organizacionales y/o Unidades organizativas.
- ✓ Registra los Procesos.
- ✓ Registra los Subprocesos.
- ✓ Registra Objetivos.
- ✓ Registra Actividades.
- ✓ Registra Riesgos

A continuación se muestra una tabla comparativa de los sistemas existentes respondiendo a los siguientes indicadores: multiplataforma, ambiente, licencia y gestión de los componentes del CI.

Sistemas	Multiplataforma	Ambiente		Licencia (Gratuito)	Gestión de los componentes del CI (mínimo uno)
		Web	Local		
MindManager	No	-	Si	Si	No
DATEV SINFOPAC Audit	Si	Si	-	No	No
SE Audit	Si	Si	-	Si	No
Meycor COSO AG	No	Si	-	No	Si
Farola	Si	Si	-	Si	Si

Tabla 1. Comparación entre los Sistemas de CI. Fuente: Elaboración propia

Después de analizar los indicadores, la tabla anterior arrojó como resultado que tres de los sistemas no tienen en cuenta los componentes que se miden para el CI en las empresas cubanas. El ambiente en la mayoría es Web, por lo que necesitan de Internet para su uso, cosa que en Cuba resulta difícil contar con este servicio. Además de que no todos son gratuitos y no corren en varias plataformas.

### **Análisis crítico de los sistemas informáticos existentes**

Luego de analizadas las características de los sistemas informáticos existentes se afirma que ninguno cumple con el objetivo general de la presente investigación. Estos sistemas tratan de manera muy general el proceso de CI, por lo que no se adapta a las exigencias del Vicedecanato de Economía y Administración de la Facultad 3, principalmente porque todos ellos son aplicaciones desarrolladas bajo leyes de CI que son muy distintas a las de Cuba. El desarrollo del módulo para el Sistema de Administración y Economía de la Facultad 3, a diferencia de los antes

analizados, incluirá la gestión de los 5 componentes que se miden para el CI según la Resolución 60 por las que se rigen las empresas cubanas.

Con el estudio de los conceptos y sistemas informáticos que realizan el CI se adquirieron las ventajas de los mismos para desarrollar el Módulo CI incluyendo las características que exige el cliente. Esto da inicio al ambiente en el que se desarrollará la presente investigación, definiendola metodología de desarrollo de software, las herramientas y tecnologías a utilizar.

### 1.3 Metodología de desarrollo de software AUP

La Universidad de las Ciencias Informáticas (UCI), opta por utilizar la variación de la metodología AUP para la UCI (Metodología de desarrollo para la actividad productiva) en el desarrollo de sistemas informáticos. La variación de AUP será la utilizada para realizar la presente investigación, debido al programa de mejora en que se encuentra inmersa la universidad, además de que el módulo a desarrollar se integra a un sistema, que tenía definido esta metodología para su completo desarrollo.

La variación de AUP describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles como:

- ✓ Desarrollo Dirigido por Pruebas
- ✓ Modelado ágil

Se basa en tres fases como se muestra en la Figura 1:

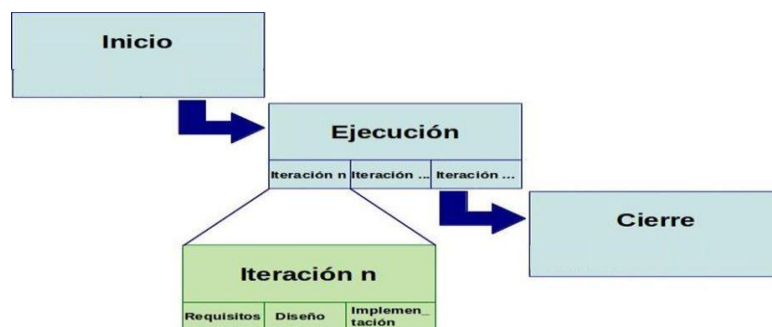


Figura 1. Fases e iteraciones de AUP(Rodriguez, 2014).

Dentro de la fase de Ejecución se encuentran las disciplinas: Modelado de Negocio, Requisitos, Análisis y Diseño, Implementación, Pruebas y Despliegue. Los artefactos a generar por cada una de las disciplinas en el desarrollo de la presente investigación son:

**Modelado de Negocio:** Mapa de procesos, Descripción de procesos de negocio, Reglas de negocio.

**Requisitos:** Modelo conceptual, Diccionario de datos, Especificación de requisitos, Evaluación de requisitos, Historias de usuario, Criterios para validar requisitos del cliente, Criterios para validar requisitos del producto.

**Análisis y diseño:** Modelo de datos, Modelo de diseño, Arquitectura.

**Implementación y Pruebas:** Diseño de casos de pruebas, Diagrama de despliegue, Diagrama de componentes.

## 1.4 Estudio de herramientas y tecnologías

En este epígrafe se describirán las herramientas y tecnologías a utilizar para el desarrollo del módulo, teniendo en cuenta las características que este debe cumplir y las necesidades del cliente. Proporcionará un ambiente cómodo para el desarrollador de acuerdo a las características que presentan las mismas.

### Lenguaje Unificado de Modelado (UML)

Este lenguaje prescribe un conjunto de notaciones y diagramas estándares para modelar sistemas orientados a objetos y describe la semántica esencial de lo que estos diagramas y símbolos significan; posibilitando así visualizar, especificar y documentar los artefactos o toda información que se obtiene o modifica durante un proceso de desarrollo de software, además de poder utilizarse para modelar distintos tipos de sistemas de software, hardware y organizaciones del mundo real(Oracle., 2013).

Principales características:

- ✓ Permite modelar sistemas haciendo uso de técnicas orientadas a objetos (OO).
- ✓ Permite especificar todas las decisiones de análisis y diseño, construyéndose así modelos precisos, no ambiguos y completos.
- ✓ Puede conectarse con lenguajes de programación (Ingeniería directa e inversa).
- ✓ Permite documentar todos los artefactos de un proceso de desarrollo (requisitos, arquitectura, pruebas, versiones, etc.).
- ✓ Es un lenguaje muy expresivo que cubre todas las vistas necesarias para desarrollar y luego desplegar los sistemas.

### Visual Paradigm para UML 8.0

Es una herramienta profesional que soporta el ciclo de vida completo del desarrollo de software: Análisis y Diseño orientados a objetos, Construcción, Pruebas y Despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, haciéndolas mejores y a un menor costo. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. La herramienta UML CASE proporciona abundantes tutoriales de UML, demostraciones interactivas de UML y proyectos UML(2014).

### PHP5.4.4-14

PHP es el acrónimo recursivo del inglés Hypertext Pre-processor (Pre-procesador de hipertextos).Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación.

Un lenguaje del lado del servidor es aquel que se ejecuta en el servidor web, justo antes de que se envíe la página a través de Internet al cliente. Las páginas que se ejecutan en el servidor pueden realizar accesos a bases de datos, conexiones en red y otras tareas para crear la página final que verá el cliente. El cliente solamente recibe una página con el código HTML<sup>8</sup> resultante de la ejecución del PHP. Como la página resultante contiene únicamente código HTML, por lo que es compatible con todos los navegadores(Álvarez, 2001).

Es necesario señalar que respecto a la seguridad, es importante el hecho de que en muchas ocasiones PHP se encuentra instalado sobre servidores Unix o Linux, que brindan un elevado nivel de seguridad intrínseco, presentan una gran estabilidad, acceso al código fuente y permiten auditar la seguridad y privacidad de los datos tratados. PHP permite configurar el servidor de modo que se apruebe o rechacen diferentes usos, lo que puede hacer que el lenguaje tenga un nivel de seguridad elevado dependiendo este de las necesidades de cada cual. Este lenguaje de programación está preparado para realizar muchos tipos de aplicaciones web gracias a la extensa librería de funciones con la que está dotado (Álvarez, 2001).

### **Symfony 2.6.3**

Es un marco de trabajo que ayuda a simplificar el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes. Además, proporciona estructura al código fuente, forzando al desarrollador a crear código legible y fácil de mantener. Por último, facilita la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas.

Symfony es un completo marco de trabajo diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones web. Separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja como el ORM Doctrine, el componente formulario que soporta la validación automática de los datos y el manejo de cache reduce la carga del servidor y disminuye el tiempo de respuesta de las peticiones del usuario. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación(Aguiluz, 2013).

Entre las características destacadas que ofrece a los desarrolladores de productos de software se encuentran las siguientes:

- ✓ Fácil de instalar y configurar en la mayoría de plataformas.
- ✓ Independiente del Sistema Gestor de Bases de Datos.
- ✓ Sencillo de usar en la mayoría de casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos.

---

<sup>8</sup>Lenguaje de marcado de hipertexto. Disponible en <http://www.ecured.cu/index.php/Html>.

- ✓ Sigue la mayoría de las mejores prácticas y patrones de diseño para la web(Aguiluz, 2013).

## **JQuery1.9.0**

JQuery es una biblioteca gratuita de JavaScript, cuyo objetivo principal es simplificar las tareas de creación de páginas web responsivas, acordes a lo estipulado en la Web 2.0, la cual funciona en todos los navegadores modernos. JQuery ayuda de gran manera en el diseño del sitio, al abstraer por completo todas las características específicas de cada uno de los navegadores. Otra de las grandes ventajas de JQuery es que se enfoca en simplificar los scripts y en acceder/modificar el contenido de una página web. Finalmente, JQuery agrega una cantidad impresionante de efectos nuevos a JavaScript<sup>9</sup>, los cuales podrán ser utilizados en tus sitios Web(The jQuery Foundation, 2012).

Beneficios del uso de JQuery:

- ✓ Utiliza sintaxis muy parecida a CSS<sup>10</sup>.
- ✓ Permite manipular series de elementos y modificarlas con una simple línea de código.
- ✓ Es muy fácil de expandir, ya que cuenta con gran cantidad de complementos que se pueden utilizar o hasta crear uno propio.
- ✓ Compatible con todos los navegadores modernos.

## **Bootstrap3.1**

Bootstrap es una colección de varios elementos web personalizables y funciones, todo esto empaquetado en una sola herramienta. Cuando se diseña una web con Bootstrap, los desarrolladores pueden elegir qué elementos utilizar. Aún más importante, tienen la certeza de saber que los elementos que elijan no generarán conflictos entre ellos.

Los elementos personalizables de Bootstrap son una combinación de HTML, CSS y JavaScript y se le han añadido una variedad de funcionalidades tales como una selección amplia de complementos JQuery(Sanchez, 2013).

## **PostgreSQL9.1.14**

PostgreSQL es un sistema de gestión de base de datos relacional orientada a objetos y libre, publicado bajo la licencia de Distribución de Software Berkeley<sup>11</sup> (BSD, Berkeley Software Distribution, según sus siglas en inglés). PostgreSQL da la posibilidad de que mientras un proceso es escrito en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases. Implementa el uso de retrocesos, subconsultas y transacciones, haciendo su funcionamiento mucho más eficaz. Posee la capacidad de comprobar la integridad referencial, así como también la de almacenar procedimientos en la propia base de datos(Martinez, 2010).

---

<sup>9</sup>Lenguaje de programación interpretado .<http://librosweb.es/javascript/>

<sup>10</sup>Hoja de estilo en cascada. <http://www.w3c.es/Divulgacion/GuiasBreves/HojasEstilo>

<sup>11</sup> Sistema operativo derivado del sistema Unix realizado por la Universidad de California en Berkeley.

## **PgAdmin 1.14.2-2**

Es una herramienta de código abierto para la administración de bases de datos PostgreSQL incluye:

- ✓ Interfaz administrativa gráfica.
- ✓ Herramienta de consulta SQL.
- ✓ Editor de código procedural.

PgAdmin se diseña para responder a las necesidades de la mayoría de los usuarios, desde escribir simples consultas SQL hasta desarrollar bases de datos complejas. La interfaz gráfica soporta todas las características de PostgreSQL y hace simple la administración. Está disponible en más de una docena de lenguajes y para varios sistemas operativos, incluyendo Microsoft Windows, Linux, Mac OSX y Solaris (Martinez, 2010).

## **PhpStorm8.0**

PhpStorm es un Entorno de Desarrollo Integrado (IDE) inteligente para desarrollar aplicaciones en Pre-procesador de Hipertextos (PHP), proporcionando herramientas esenciales como análisis de código y comprobación de errores (Company, 2014).

Las principales novedades en PhpStorm 2.0 incluyen:

- ✓ Soporte para PHP 5.3, incluyendo espacios de nombre.
- ✓ Depuración con cero configuraciones con todos los navegadores.
- ✓ Compatibilidad con el marco de trabajo Symfony.
- ✓ Editores de Lenguaje de Consulta Estructurado (SQL) con resultados editables.
- ✓ Soporte para Lenguaje Marcado de Hipertextos (HTML5).

## **Apache2.2.22-13**

Apache es el servidor web por excelencia, su facilidad de configuración, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. Se ejecuta en gran cantidad de sistemas operativos, lo que lo hace prácticamente universal. Es una tecnología gratuita, de código abierto, altamente configurable y de diseño modular por lo que resulta muy sencillo ampliar sus capacidades. Permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor y es posible configurarlo para que ejecute un determinado script cuando esto suceda (Red Hat Enterprise, 2003).

## **Subversion 1.6.17**

Es un sistema de control de versiones libre y de código fuente abierto. Gestiona un árbol de ficheros en un repositorio central. El repositorio se refiere a un servidor de ficheros ordinario, excepto porque recuerda todos los cambios hechos a sus ficheros y directorios. Esto le permite recuperar versiones antiguas de sus datos o examinar el historial de cambios de los mismos. Subversion es un sistema general que puede ser usado para administrar cualquier conjunto de ficheros, ya sean código fuente, archivos multimedia o cualquier otro tipo de documento (2004).

Ventajas:

- ✓ Sigue la historia de los archivos y directorios a través de copias y renombrados.
- ✓ La creación de ramas y etiquetas es una operación más eficiente. Tiene costo de complejidad constante ( $O(1)$ )<sup>12</sup>.
- ✓ Permite selectivamente el bloqueo de archivos.
- ✓ Cuando se usa integrado a Apache permite utilizar todas las opciones que este servidor provee a la hora de autenticar archivos (SQL, LDAP<sup>13</sup>, entre otros).

Las herramientas y tecnologías antes mencionadas fueron seleccionadas debido a las potencialidades que presentan, además de que el módulo a desarrollar debe integrarse a un sistema, el cual ya tenía definidas estas herramientas y tecnologías para su implementación y que el equipo de desarrollo tiene experiencia en el uso de las mismas.

## 1.5 Conclusiones parciales

- ✓ La revisión de la literatura mostró las características principales de los procesos que se realizan en el CI, las cuales están asociadas a los 5 componentes que se manejan en dicho proceso.
- ✓ Se realizó un estudio de los sistemas existentes que realizan el CI a nivel internacional y nacional, obteniendo como resultado que ninguno de los sistemas consultados incluye la gestión de los 5 componentes con cada una de las normas especificadas en la Resolución 60.
- ✓ Para el desarrollo del SCI, dada las condiciones del equipo de trabajo y de las características del mismo, se seleccionó la metodología de desarrollo del software la variación de AUP para la UCI por las conveniencias que este brinda con respecto al equipo de trabajo.
- ✓ Para la descripción y el modelado del sistema se optó por la herramienta CASE Visual Paradigm para UML 8.0 y para la implementación del sistema se empleó el IDE PhpStorm en su versión 8.0. Se seleccionaron como marcos de trabajo Symfony 2 versión 2.5.6, Bootstrap 3.1 y JQuery 1.9.0 y como sistema gestor de base de datos se utilizará PostgreSQL 9.1.14.

---

<sup>12</sup> Cuando las instrucciones se ejecutan una vez (Complejidad constante).  
<http://www.lab.dit.upm.es/~lprg/material/apuntes/o/>

<sup>13</sup> Protocolo Ligero Simplificado de Acceso a Directorios. <http://es.kioskea.net/contents/269-protocolo-ldap>

# Capítulo 2

## **CAPÍTULO 2. MODELADO DEL NEGOCIO, ANÁLISIS Y DISEÑO DE LA SOLUCIÓN**

### **Introducción**

En este capítulo se realizará el modelado del negocio. Antes de desarrollar un software se hace necesario comprender el negocio mediante el estudio de los procesos que en él se desarrollan, se realizará una descripción del mismo y se detallarán los procesos implicados. Posteriormente se identificarán los requisitos funcionales con el cliente mediante las técnicas tormenta de ideas y entrevista, validando estos requisitos mediante las técnicas prototipado y casos de pruebas. Se reflejarán los requisitos no funcionales definidos para el sistema. Se mostrará cómo está concebida la arquitectura y el diseño del software mediante los artefactos generados, haciendo uso de patrones y aplicando métricas para la validación del diseño propuesto. Lo antes mencionado se desglosa en las disciplinas Modelado de negocio, Requisitos, Análisis y Diseño. Según la metodología de desarrollo estas se realizan en la fase de Ejecución.

### **2.1 Modelado del negocio**

El Modelado del Negocio es la disciplina destinada a comprender los procesos de negocio de una organización. Se comprende cómo funciona el negocio que se desea informatizar para tener garantías de que el software desarrollado va a cumplir su propósito (Rodríguez, 2014). Se identificarán los procesos de negocio a través de una entrevista realizada al cliente describiendo como se ejecutan cada uno de estos procesos de forma manual en el Vicedecanato de Economía y Administración de la Facultad 3. A continuación se realiza la modelación del negocio asociado al Módulo CI.

#### **2.1.1 Procesos del CI**

Un proceso de negocio es un conjunto de tareas relacionadas lógicamente llevadas a cabo para lograr un resultado de negocio definido (Ould, 2011). Los procesos de negocio que fueron identificados son:

- ✓ Adecuar Guía de Autocontrol
- ✓ Aplicar Guía de Autocontrol
- ✓ Realizar seguimiento al Plan de Medidas

Se muestra la descripción de uno de los procesos de negocio "Adecuar Guía de Autocontrol". Este escenario es el escogido para desarrollar todos los artefactos que se muestran en la presente investigación porque es el primer proceso que se ejecuta a la hora de realizar el CI en la facultad. Las restantes descripciones pueden ser consultadas en Anexo 1.



## Capítulo 2

### Descripción del proceso de negocio Adecuar Guía de Autocontrol

<b>Objetivo</b>	Realizar la adecuación de la Guía de Autocontrol General para cada una de las áreas
<b>Evento(s) que lo genera(n)</b>	Adecuar la Guía de Autocontrol Genérica a cada una de las áreas Aplicar Guía de Autocontrol a cada una de las áreas
<b>Pre condiciones</b>	Se cuenta con los objetivos de trabajo de cada una de las áreas y con la Guía de autocontrol Genérica
<b>Marco legal</b>	Resolución 60 de fecha 1ro de marzo del 2011
<b>Reglas de negocio</b>	La adecuación de la Guía de Autocontrol solamente la pueden realizar los jefes de áreas.
<b>Responsable</b>	Vicedecana de Economía y Administración
<b>Clientes internos</b>	Vicedecana de Economía y Administración Jefes de áreas Trabajadores de las áreas
<b>Clientes externos</b>	N/A
<b>Entradas</b>	Objetivos de trabajo de las áreas Guía de Autocontrol Genérica
<b>Flujo de eventos</b>	
<b>Flujo básico</b>	
1.	Solicitar objetivos de trabajo de cada una de las áreas. La Vicedecana de Economía y Administración(VEA) solicita a cada uno de los jefes de áreas los objetivos de trabajo correspondientes de la misma.
1.	Enviar objetivos de trabajo del área. El jefe de área le envía los objetivos de trabajo a la VAE.
2.	Recibir los objetivos de trabajo del área. La VEA recibe los objetivos de cada una de las áreas.
3.	Revisar la Guía de Autocontrol Genérica (GAG) para cada una de las áreas. La VEA revisa la GAG de acuerdo a los objetivos de trabajo que presenta cada área.
4.	Seleccionar los ítems de la GAG que se adecuan con los objetivos de trabajo de cada una de las áreas. Una vez revisada la GAG por la VEA de acuerdo a los objetivos que tiene cada área, la misma selecciona cuáles son los ítems que

## Capítulo 2

	se van a corresponder con dicha área.
5.	Revisar la adecuación. Una vez seleccionado los ítems quedando conformada la adecuación la VEA revisa dicha adecuación. En caso de la adecuación tenga algún error ver flujo alterno <u>5.a Adecuación Incorrecta</u>
6.	Crear una nueva Guía de Autocontrol (GA) para cada área. Una vez seleccionados los ítems que se corresponden con cada área según sus objetivos la VEA conforma una nueva GA para cada una de las áreas para su posterior supervisión.
7.	Enviar GA. La VEA envía la adecuación a los jefes de áreas.
8.	Recibir GA. Los jefes de áreas reciben la GA adecuada correspondiente.
<b>Pos-condiciones</b>	
1.	Se crea una nueva Guía de Autocontrol
<b>Salidas</b>	
7.	Guía de Autocontrol
<b>Flujos paralelos</b>	
N/A	
<b>Pos-condiciones</b>	
N/A	
<b>Salidas</b>	
N/A	
<b>Flujos alternos</b> <u>5.a Adecuación Incorrecta</u>	
5.a La VEA informa que los ítems seleccionados no son los correctos esa adecuación. Vuelve al paso 4 del flujo básico.	
<b>Pos-condiciones</b>	
Se informa que los ítems seleccionados no son los correctos para esa adecuación.	
<b>Salidas</b>	
N/A	
<b>Asuntos pendientes</b>	
N/A	

Tabla 2. Descripción de proceso de negocio Adecuar Guía de Autocontrol. Fuente:Elaboración propia

### Diagrama de proceso de negocio

Un diagrama de procesos de negocio es un conjunto de objetos gráficos, correspondientes a las actividades y controles de flujo que definen el orden de ejecución de estas. Seguidamente se muestra un ejemplo del diagrama de proceso de negocio correspondiente al proceso Adecuar Guía de Autocontrol:

# Capítulo 2

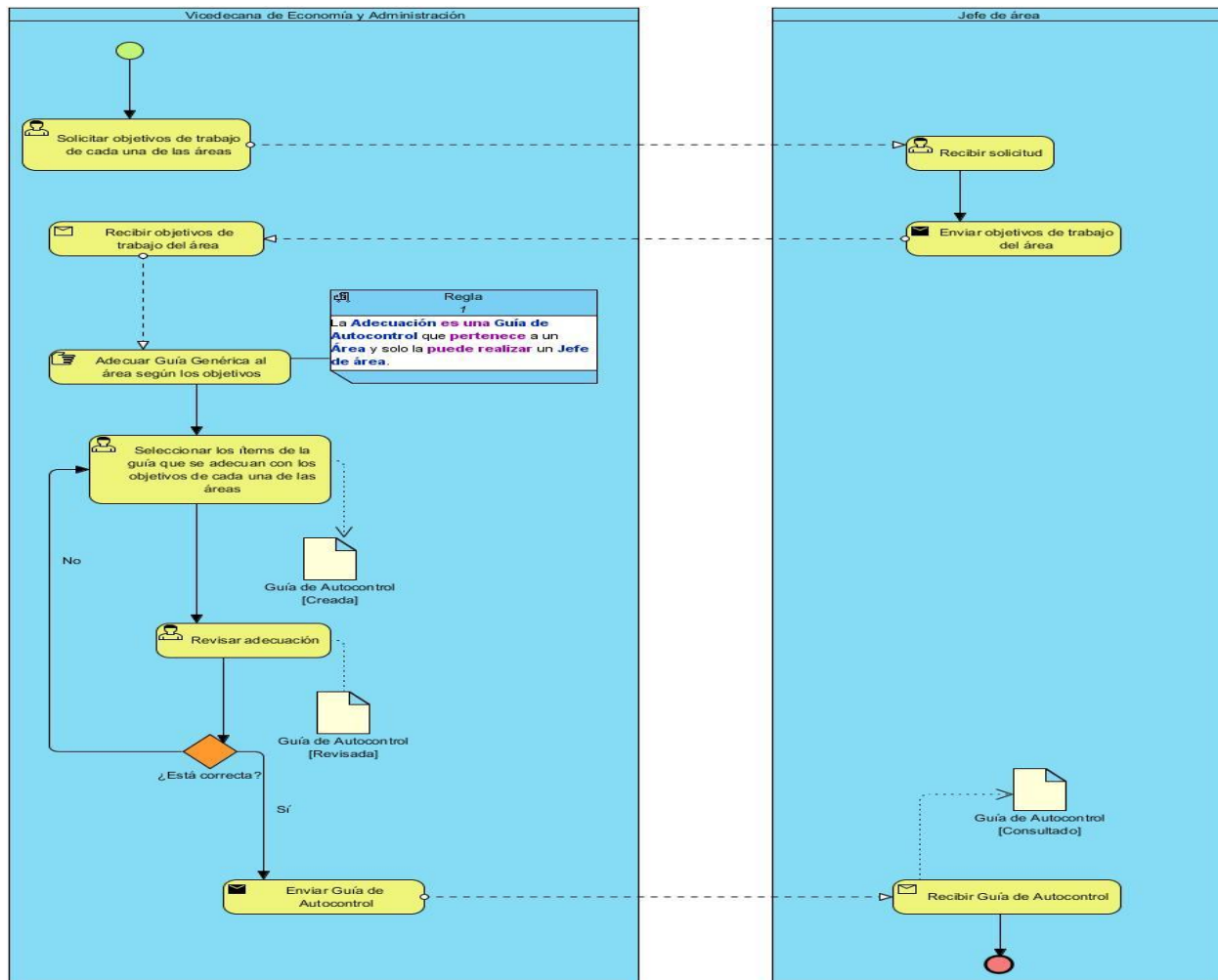


Figura 2. Diagrama de proceso de negocio Adecuar Guía de Autocontrol. Fuente:Elaboración propia

La descripción de los procesos de negociopermitió tener una visión detallada de los procesos que se llevan a cabo en el Vicedecanato de Economía y Administración en cuanto a actividades que se realizan, artefactos que se generan y el flujo de ejecución de las mismas. Estas descripciones ayudan a identificar los principales conceptos que se manejan en el negocio como se muestra en el epígrafe siguiente.

## 2.1.3 Modelo conceptual

Los mapas conceptuales proporcionan un resumen esquemático, claro y ordenado de lo que se ha estudiado, en cuanto a conceptos y sus relaciones.

Importancia:

- ✓ Facilitan una rápida visualización de los contenidos de aprendizaje.
- ✓ Favorecen el recuerdo y el aprendizaje de manera organizada y jerarquizada.

# Capítulo 2

- ✓ Permiten una rápida detección de los conceptos clave de un tema, así como de las relaciones entre los mismos.

Se realizó un Modelo conceptual que ilustra los conceptos involucrados en el CI que se realiza en la Facultad 3 y la relación que existe entre estos:

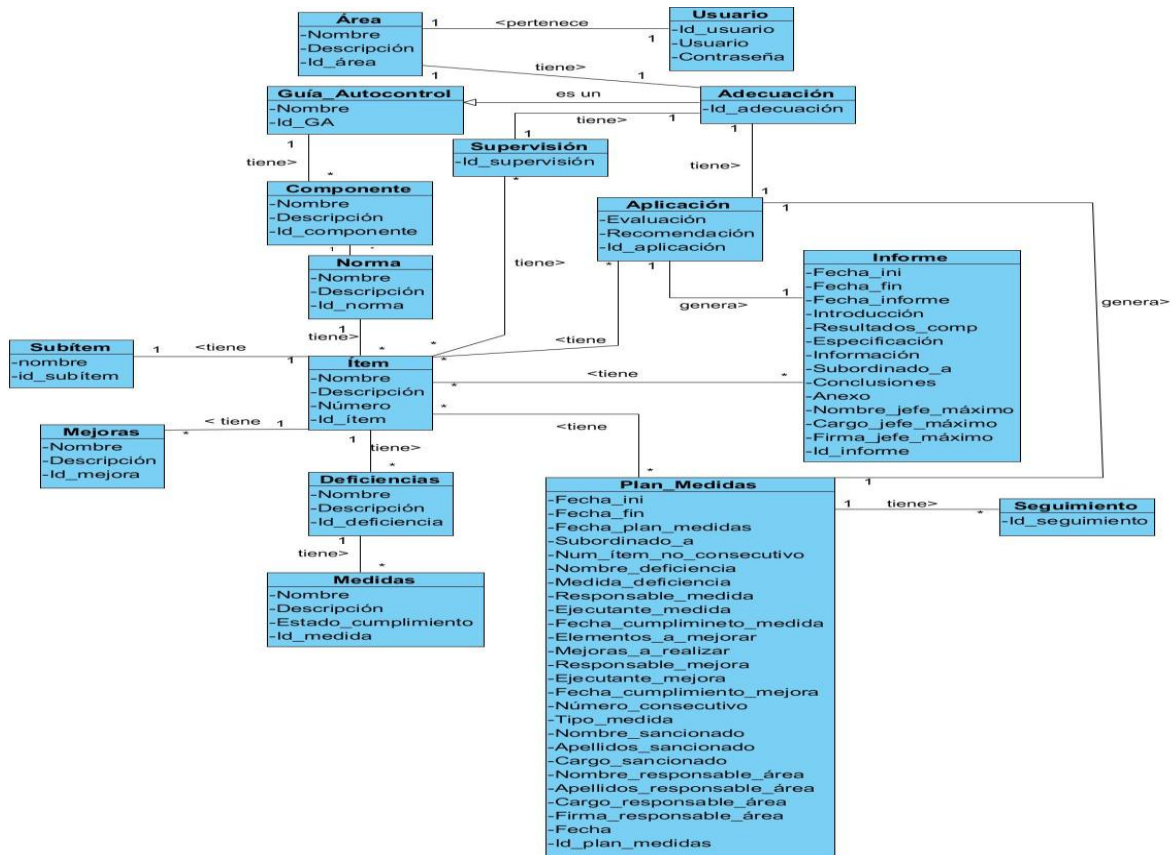


Figura 3. Modelo conceptual del Módulo CI. Fuente: Elaboración propia

A continuación se muestran las descripciones de los principales conceptos del módulo CI:

**Guía de Autocontrol (GA):** la GA es un documento que se adecua según el área a la que se le vaya a realizar el CI. Contiene los 5 componentes registrados en la Resolución 60 de la Contraloría General de la República, con cada una de las normas correspondientes a dicho componente, además de los ítems a evaluar en dicho control.

**Componente:** se refiere a los componentes que establece la GA definidos en la Resolución 60 de la Contraloría General de la República para realizar el CI.

**Norma:** son las normas correspondientes a cada componente de la GA definidas en la Resolución 60 de la Contraloría General de la República.

**Ítem:** se refiere a los ítems (parámetros a evaluar) correspondientes a cada norma de los componentes de la GA.

## Capítulo 2

**Adecuación:** la adecuación se le realiza a la GA a un área determinada. Contiene los componentes de la Resolución 60 de la Contraloría General de la República, con cada una de sus normas y los ítems que se evalúan en dicha área.

**Aplicación:** es la acción de aplicar la GA al área que será controlada. Esta aplicación ya debe tener la guía adecuada para dicha área.

Las restantes descripciones pueden ser consultadas en el diccionario de datos que se encuentra en Anexo 2.

Luego de describir los procesos de negocio detalladamente se identificaron las posibles actividades a informatizar convirtiéndose en requerimientos que debe cumplir el módulo a desarrollar.

### **2.2 Requisitos**

El esfuerzo principal en la disciplina Requisitos es desarrollar un modelo del sistema que se va a construir. Esta disciplina comprende la administración y gestión de los requisitos funcionales y no funcionales del producto (Rodríguez, 2014). Una adecuada comprensión de los requisitos favorece al desarrollo de nuevos sistemas informáticos que cumplan con las necesidades y expectativas del cliente. Para realizar este procedimiento existen diversas técnicas que guían al analista en el proceso de comunicación con el cliente y el equipo de desarrollo.

#### **2.2.1 Técnicas para la captura de requisitos funcionales**

En la presente investigación se utilizaron las técnicas: entrevista y tormenta de ideas.

**Entrevista:** es de gran utilidad para obtener información cualitativa como opiniones o descripciones de actividades. Requiere seleccionar bien a los entrevistados para obtener la mayor cantidad de información en el menor tiempo posible. Es muy aceptada y permite acercarse al problema de una manera natural (Raghavan, y otros, 1994).

**Tormenta de ideas:** esta técnica se puede utilizar para identificar un primer conjunto de requisitos en aquellos casos donde no están muy claras las necesidades que hay que cubrir (Raghavan, y otros, 1994). Para la aplicación de esta técnica se realizaron varias reuniones entre el cliente y el equipo de desarrollo donde ambas partes brindaban sus ideas en cuanto a la propuesta de solución.

#### **2.2.2 Requisitos funcionales del sistema**

Los requisitos funcionales reflejan lo que debe hacer y de qué manera debe reaccionar el sistema ante las entradas y cómo debe comportarse en situaciones específicas. Estos detallan la función del producto de software, entrada, salidas, excepciones y usuarios.

## Capítulo 2

Como resultado del uso de estas técnicas de obtención de requisitos se identificaron 35 requisitos funcionales, 28 de complejidad baja, 2 de complejidad media y 5 de complejidad alta, los cuales se representan a continuación:

Prioridad	Complejidad	Nro. del Requisito	Nombre del Requisito
Muy Alta	Alta	RF1	Adecuar Guía de Autocontrol
	Alta	RF2	Modificar Adecuación de la Guía de Autocontrol
	Media	RF3	Eliminar Adecuación de la Guía de Autocontrol
	Alta	RF4	Aplicar Guía de Autocontrol
	Alta	RF5	Generar Informe de la aplicación de la Guía de Autocontrol
	Alta	RF6	Generar Plan de Medidas de la aplicación de la Guía de Autocontrol
	Media	RF7	Generar seguimiento del Plan de Medidas
Alta	Baja	RF8	Adicionar Guía de Autocontrol
	Baja	RF9	Modificar Guía de Autocontrol
	Baja	RF10	Eliminar Guía de Autocontrol
	Baja	RF11	Adicionar componente
	Baja	RF12	Modificar componente
	Baja	RF13	Eliminar componente
	Baja	RF14	Adicionar norma
	Baja	RF15	Modificar norma
	Baja	RF16	Eliminar norma
	Baja	RF17	Adicionar ítem
	Baja	RF18	Modificar ítem
	Baja	RF19	Eliminar ítem
Media	Baja	RF20	Adicionar área
	Baja	RF21	Modificar área
	Baja	RF22	Eliminar área
	Baja	RF23	Adicionar deficiencias
	Baja	RF24	Modificar deficiencias
	Baja	RF25	Eliminar deficiencias
	Baja	RF26	Adicionar medidas

## Capítulo 2

	Baja	RF27	Modificar medidas
	Baja	RF28	Eliminar medidas
	Baja	RF29	Adicionar mejoras
	Baja	RF30	Modificar mejoras
	Baja	RF31	Eliminar mejoras
	Baja	RF32	Autenticar Usuario
	Baja	RF33	Adicionar usuario
	Baja	RF34	Modificar usuario
	Baja	RF35	Eliminar usuario

*Tabla 3. Requisitos Funcionales del Módulo CI. Fuente: Elaboración propia*

La tabla ilustra los requisitos funcionales definidos para el Módulo CI con la complejidad y prioridad de cada uno de ellos. La complejidad se calculó teniendo en cuenta los criterios del cliente y mediante el artefacto Evaluación de requisitos que evalúa los parámetros interfaces, diferentes comportamientos, formas de inicialización, consultas fuentes de datos, restricciones de validación, grado de reutilización y lógica de negocio definido en el expediente de la metodología de desarrollo. La prioridad se calculó mediante la técnica de Pressman y Wieggers donde definen dos parámetros: importancia y urgencia. Los requisitos son urgentes o importantes según el criterio del cliente y del desarrollador. Las diferentes combinaciones de ambos parámetros dan el resultado de la prioridad, con la ayuda del cálculo de la Trazabilidad de los requisitos, artefacto definido en el expediente de la metodología. A continuación se describe con claridad la técnica de Wieggers Referencia (E. Wieggers, 2003):

1. Si el requisito es importante y urgente la prioridad es alta.
2. Si el requisito es importante y no urgente la prioridad es media.
3. Si el requisito no es importante pero es urgente hay que revisar dicho requisito.
4. Si el requisito no es importante y no es urgente la prioridad es baja.

Una vez identificados los requisitos se realiza una descripción de los mismos para tener un entendimiento claro y ordenado para su posterior implementación en el Módulo CI.

### **2.2.3 Descripción de requisitos funcionales del sistema**

Las descripciones de los requisitos se realizan mediante las Historias de Usuario (HU), ya que pueden escribirse sobre una nota adhesiva pequeña y al ser muy cortas, éstas representan requisitos del modelo de negocio que pueden implementarse rápidamente.

#### **Historias de Usuario**

## Capítulo 2

Las historias de usuario son una forma rápida de administrar los requisitos de los usuarios sin tener que elaborar gran cantidad de documentos formales y sin requerir de mucho tiempo para administrarlos. Las historias de usuario permiten responder rápidamente a los requisitos cambiantes (Wesley, 2004).

Seguidamente se muestra la descripción de uno de los requisitos funcionales, mediante la historia de usuario "Adecuar Guía de Autocontrol perteneciente al requisito Gestionar Adecuación de la Guía de Autocontrol. Las restantes descripciones pueden ser consultadas en Anexo 3.

### HU Gestionar adecuación de la Guía de Autocontrol

#### HU-01-Adecuar Guía de Autocontrol


<b>Número:</b> HU-01		<b>Nombre del requisito:</b> Adecuar Guía de Autocontrol	
<b>Programador:</b> Yanirosa Almenares		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 7 días	
<b>Riesgo en Desarrollo:</b> Alta		<b>Tiempo Real:</b> 2 semanas	
<b>Descripción:</b> El usuario podrá seleccionar dentro de los componentes y sus normas, los ítems adecuados al área, quedando conformado la Guía de Autocontrol para dicha área.			
<b>Prototipo de interfaz:</b>			
			

Tabla 4. Historia de Usuario Adecuar Guía de Autocontrol. Fuente: Elaboración propia



## Capítulo 2

Después de haber definidos los comportamientos específicos para el Módulo CI, el equipo de desarrollo define un conjunto de requisitos no funcionales, propiedades o cualidades que el Módulo CI debe cumplir.

### **2.2.4 Requisitos no funcionales del sistema**

Los requisitos no funcionales (RNF) son restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo y estándares. A menudo son aplicados en su totalidad al sistema y normalmente apenas se aplican a características o servicios individuales del sistema (Sommerville, 2007).

Se identificaron 22 requisitos no funcionales los cuales se muestran en la siguiente tabla:

<b>Requisitos no funcionales</b>	
<b>Usabilidad</b>	
<u>Capacidad para ser atendido</u>	<ul style="list-style-type: none"><li>✓ El tiempo de entrenamiento requerido para que usuarios normales y avanzados sean productivos operando el módulo es de 2 días.</li><li>✓ El módulo notifica a los usuarios los errores y sugiere cómo corregirlos.</li><li>✓ El módulo valida automáticamente la información contenida en los formularios de ingreso. En el proceso de validación de la información, se tienen en cuenta aspectos tales como obligatoriedad de campos, longitud de caracteres permitida por campo, manejo de tipos de datos, el sistema no permitirá la entrada de datos incorrectos.</li></ul>
<u>Capacidad para ser aprendido</u>	<ul style="list-style-type: none"><li>✓ El módulo estandariza los botones Adicionar de color verde, los de Modificar azul y los de Regresar y Cancelar anaranjado.</li><li>✓ Los botones siempre aparecen del lado izquierdo de la interfaz.</li><li>✓ La forma de llenar los formularios es de arriba hacia abajo.</li><li>✓ Los menús aparecen en el orden en que se lleva a cabo el negocio.</li></ul>
<u>Capacidad de atracción</u>	<ul style="list-style-type: none"><li>✓ El módulo ofrece una interfaz fácil de operar para el cliente.</li><li>✓ Diseño sencillo, con pocas entradas, permitiendo que no sea necesario mucho entrenamiento para que los usuarios puedan utilizar el módulo.</li></ul>
<b>Funcionalidad</b>	

## Capítulo 2

<u>Seguridad de acceso</u> <ul style="list-style-type: none"> <li>✓ El control de acceso se establecerá por roles que se le asignarán a los usuarios que interactúen con el módulo.</li> <li>✓ Todo uso de las funcionalidades del módulo requiere la autenticación de los usuarios.</li> <li>✓ El módulo concederá acceso a cada usuario autenticado solo a las funciones que le estén permitidas, de acuerdo a la configuración del sistema.</li> <li>✓ Los documentos que exporta el módulo deben estar dotados de las configuraciones necesarias para que sean de solo lectura.</li> </ul>
<b>Eficiencia</b>
<u>Utilización de recursos</u> <ul style="list-style-type: none"> <li>✓ El módulo interactuará con impresoras para imprimir los diferentes documentos que genere la aplicación como respuesta a las funcionalidades del sistema.</li> </ul>
<b>Portabilidad</b>
<u>Adaptabilidad</u> <ul style="list-style-type: none"> <li>✓ El módulo permitirá la manipulación de dispositivos externos como las impresoras.</li> </ul>
<u>Instalabilidad</u> <ul style="list-style-type: none"> <li>✓ El sistema podrá ser instalado en el ambiente especificado en los requisitos tecnológicos para servidores.</li> </ul>
<b>Hardware</b>
El hardware mínimo para oficinas que necesitará el módulo es de 1 GB de memoria RAM, procesador Pentium IV y 80 GB de disco duro.
<ul style="list-style-type: none"> <li>✓ El hardware mínimo para oficinas que necesitará el módulo es de 2 GB de memoria RAM, procesador Dual Core o superior y 120 GB de disco duro.</li> </ul>
<b>Software</b>
<ul style="list-style-type: none"> <li>✓ Servidores de base de datos locales PostgreSQL versión 9.1</li> <li>✓ Servidor web Apache versión 2.2.22-13</li> <li>✓ Multiplataforma</li> <li>✓ En distintos navegadores</li> </ul>

*Tabla 5. Requisitos No Funcionales del Módulo CI. Fuente: Elaboración propia.*

Los requisitos una vez definidos necesitan ser válidos, estas validaciones tienen como misión demostrar que la definición de ellos concreta el sistema que el usuario necesita o que el cliente desea.

### **2.2.5 Técnicas para la validación de requisitos**

La validación de requisitos concreta el sistema que el usuario necesita o el cliente desea. Esta es una actividad fundamental, un levantamiento de requisitos con errores que no se detecten a

## Capítulo 2

tiempo y que conduzcan a resultados inesperados, provoca costos excesivos y gran pérdida de tiempo.

Para la validación de los requisitos de la presente investigación se aplicaron dos técnicas:

**Prototipado:**son la versión inicial de un sistema de software que se utiliza para demostrar los conceptos, probar las opciones de diseño y entender mejor el problema y su solución. Permiten revelar errores u omisiones en los requisitos propuestos.

**Casos de prueba:** permiten definir las entradas, las salidas del software y acciones del usuario para que se pueda completar lo que expresa el requisito. Además, como los requisitos deben ser posibles de probar, se le debe de asociar un caso de prueba por cada uno.

El análisis de los requisitos y la captura de los mismos aseguraron el desarrollo del módulo dando una base sólida sobre la que trabajar. Para el desarrollo de Módulo CI, no solo es importante tener en cuenta las características a desarrollar, sino también la arquitectura que debe regir la solución técnica, para esto se inicia la disciplina Análisis y diseño.

### **2.3 Análisis y diseño**

En esta disciplina, si se considera necesario, los requisitos pueden ser refinados y estructurados para conseguir una comprensión precisa de estos, una descripción que sea fácil de mantener y ayude a la estructuración del sistema (incluyendo su arquitectura). Además, en esta disciplina se modela el sistema y su forma (incluida su arquitectura) para que soporte todos los requisitos, incluyendo los requisitos no funcionales (Rodríguez, 2014).

El diseño que se realiza debe buscar ante todo satisfacer los requerimientos que influyen a la arquitectura que se definirá para el Módulo CI.

#### **2.3.1 Arquitectura del sistema**

Para el desarrollo del módulo se utiliza el marco de trabajo Symfony. Está basado en la arquitectura MVC (Modelo-Vista-Controlador). Esta arquitectura permite dividir la aplicación en tres grandes capas: *Vista*, *Modelo* y *Controlador*. La siguiente imagen muestra el funcionamiento interno de la arquitectura de Symfony:

## Capítulo 2

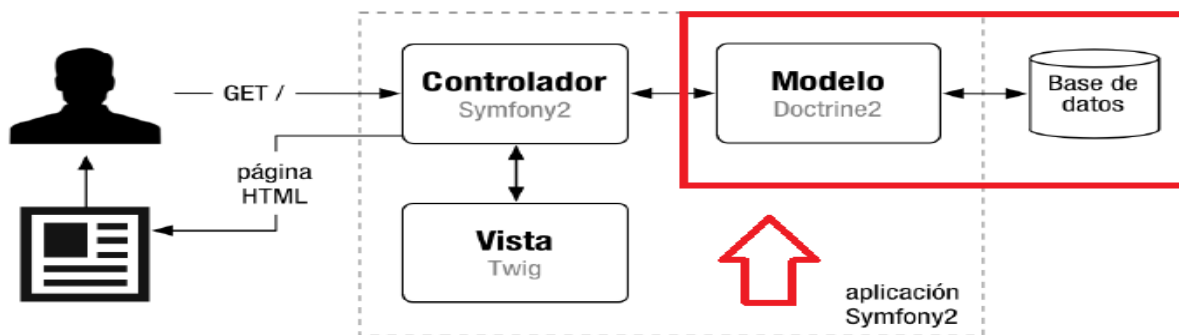


Figura 4. Arquitectura interna en el marco de trabajo Symfony 2(Aguiluz, 2013).

Cuando el usuario solicita ver la portada del sitio, internamente sucede lo siguiente(Aguiluz, 2013):

1. El sistema de enrutamiento determina qué *Controlador* está asociado con la página de la portada.
2. Symfony2 ejecuta el *Controlador* asociado a la portada. Un *controlador* no es más que una clase PHP en la que puedes ejecutar cualquier código que quieras.
3. El *Controlador* solicita al *Modelo* los datos. El *modelo* no es más que una clase PHP especializada en obtener información, normalmente de una base de datos.
4. Con los datos devueltos por el *Modelo*, el *Controlador* solicita a la *Vista* que cree una página mediante una plantilla y que inserte los datos del *Modelo*. La *Vista* es la página HTML que utilizan los usuarios para interactuar con la aplicación.
9. El *Controlador* entrega al servidor la página creada por la *Vista*.

Se evidencia el funcionamiento de esta arquitectura en el escenario Gestionar Adecuación de la Guía de Autocontrol como muestra la Figura:

## Capítulo 2

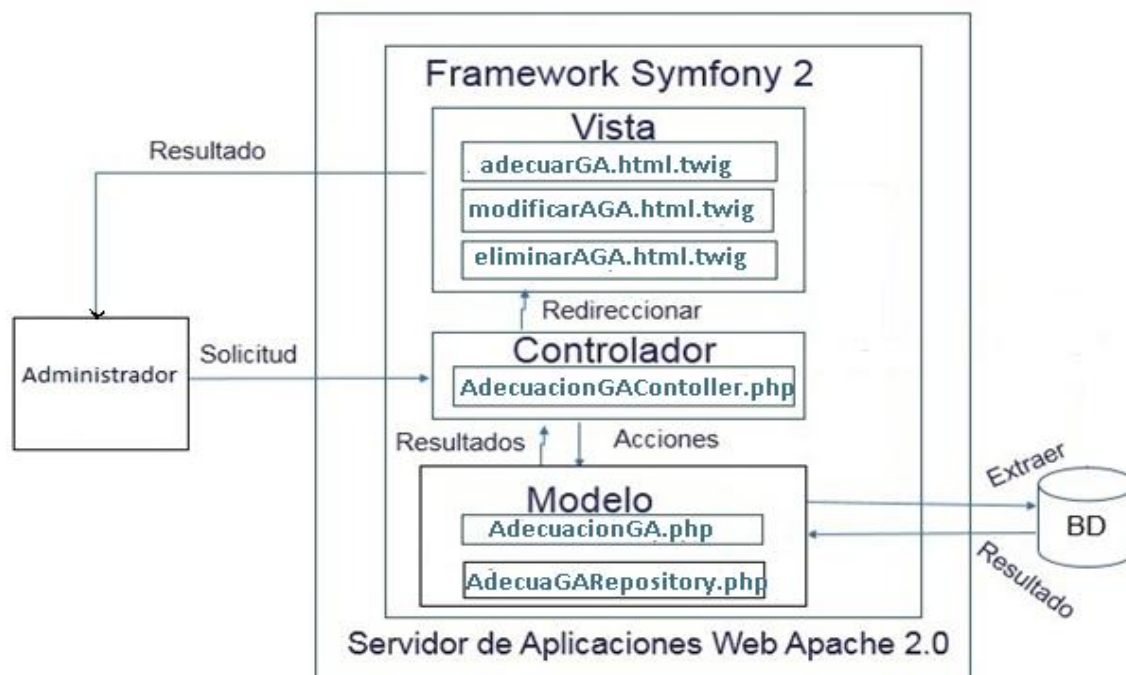


Figura 5. Arquitectura MVC en el escenario Gestionar adecuación de la Guía de Autocontrol. Fuente: Elaboración propia.

Se definieron las estructuras que componen la arquitectura, donde la creación de estas se hace en base a patrones de software para realizar los diagramas de clases del diseño detallando concretamente la implementación del Módulo CI.

### 2.3.2 Diagrama de clases del diseño

Los diagramas de clases son utilizados durante el proceso de Diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, los componentes que se encargarán del funcionamiento y la relación entre uno y otro. Un diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos(Larman, 1999).

En el siguiente diagrama de clases se muestra el escenario “Gestionar Adecuación de la Guía de Autocontrol”. Entre las principales clases que se representan se encuentran:

- ✓ La clase AdecuacionGA.html.twig que representa la capa de presentación o vista del patrón MVC. Es la interfaz con la que el usuario interactúa, mostrándole los datos necesarios y las acciones q puede llevar a cabo.
- ✓ La clase controladora “AdecuaciónController.php”, encargada de las funcionalidades que debe realizar el escenario anterior.

# Capítulo 2

- ✓ La clase entidad “CE\_Adecuacion” encargada de almacenar la información referente a una *Adecuación* la cual se utiliza para dar cumplimiento a las funcionalidades del escenario anterior.

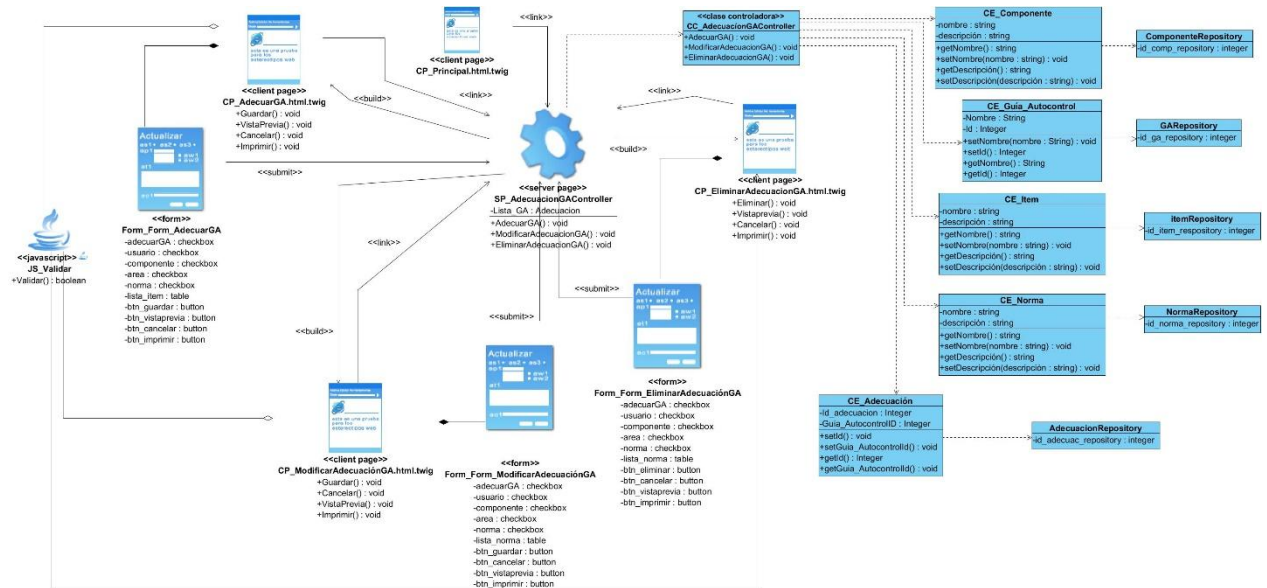


Figura 6. Diagrama de clases de diseño. Escenario Gestionar Adecuación de la Guía de Autocontrol. Fuente: Elaboración propia.

Los restantes diagramas de clases de diseño pueden ser consultados en Anexo 4.

Los diagramas de clases del diseño brindaron la estructura del Módulo CI mostrando sus clases, atributos y las relaciones entre ellos, así como el correcto diseño de las mismas basado en el estudio de los patrones de software: patrones de diseño y arquitectónicos.

## 2.3.3 Patrones de software

### Patrones de diseño

Los patrones son soluciones simples y elegantes a problemas específicos y comunes del diseño Orientado a Objetos basadas en la experiencia, estos permiten reutilizar la experiencia de los desarrolladores, clasificar y describir formas de solucionar problemas que ocurren de forma frecuente en el desarrollo y está basado en la recopilación del conocimiento de los expertos en desarrollo de software (S. Pressman, 2010).

### GRASP

GRASP significa Patrones Generales de Software para Asignar Responsabilidades, describen los principios fundamentales del diseño de objetos y la asignación de responsabilidades,

## Capítulo 2

expresados como patrones. Constituyen la base de cómo se diseñará el sistema (Larman, 2003)

Entre los patrones utilizados para el desarrollo del sistema se encuentran los siguientes:

Experto: el patrón indica que la responsabilidad de la creación de un objeto o la implementación de un método, debe recaer sobre la clase que conoce toda la información necesaria para crearlo. Fue usado con el objetivo de darle a las clases la responsabilidad necesaria siempre que contarán con la información para cumplirla (Larman, 1999). Logrando así un mejor comportamiento entre las clases, fáciles de comprender y mantener. Este patrón es ampliamente usado en las clases entidades del modelo, por ejemplo, en la clase `AdecuacionGA.php`, es la responsable de manejar la información correspondiente a las adecuaciones y la encargada de toda la lógica del acceso a los datos.

Creador: el patrón creador guía la asignación de responsabilidades relacionadas con la creación de objetos. Su propósito es encontrar un creador que se conecte con el objeto producido en cualquier evento (Larman, 1999). En la solución propuesta este patrón se evidencia en las clases controladoras ya que tienen toda la información necesaria para realizar la acción y que usa directamente las instancias creadas del objeto. Un ejemplo es la clase `AdecuacionGAController.php`, permitiendo crear una nueva adecuación para el usuario autenticado.

Controlador: es un objeto que no pertenece a la interfaz de usuario, responsable de recibir o manejar un evento del sistema. Define además el método de su operación (Larman, 1999). El patrón controlador es el encargado de asignar la responsabilidad de controlar el flujo de eventos del sistema, a una clase específica. Este patrón se ve reflejado en las clases controladoras, ejemplo la clase `ComponenteController.php` ya que controla la información a almacenar de los Componentes.

Bajo acoplamiento: el acoplamiento es una medida de la fuerza con que una clase está conectada a otras clases, con que las conoce y con que recurre a ellas. Una clase con bajo acoplamiento no depende de muchas otras clases (Larman, 1999). Este patrón tiene como idea, tener las clases lo menos ligadas entre sí que se pueda. De tal forma que en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de las clases, potenciando la reutilización y disminuyendo la dependencia entre ellas. Por ejemplo, se tiene la necesidad de crear una instancia de *Componente*, el patrón Creador sugiere la creación de esta instancia en la clase `ComponenteController.php`, por los elementos anteriormente mencionado cuando se analizó este patrón, aunque esta clase tiene una relación con `GuíaAutocontrolController.php` y `NormaController.php`. La asignación de responsabilidad que se

## Capítulo 2

sugiere no aumenta acoplamiento entre las clases. Desde el punto de vista puramente del acoplamiento, es preferible realizar la instanciación en la clase `ComponenteController.php` y se mantiene el acoplamiento global más bajo entre estas tres clases.

Alta cohesión: en la perspectiva del diseño orientado a objetos, es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase. Una alta cohesión caracteriza a las clases con responsabilidades estrechamente relacionadas que no realicen un trabajo enorme (Larman, 1999). Cada clase del modelo implementa sus propios métodos y ayuda a resolver alguna tarea en otra clase que dependa de esta. El ejemplo que se analizó anteriormente soporta tanto bajo acoplamiento como alta cohesión y el diseño delega la responsabilidad de “*AdicionarComponente*” a la clase `ComponenteController.php` que favorece una cohesión muy alta entre las tres clases. Igual que el Bajo Acoplamiento, el patrón de Alta Cohesión es un principio a tener en cuenta durante todas las decisiones de diseño; es un objetivo subyacente a tener en cuenta continuamente. Es un principio evaluativo que aplica un diseñador mientras evalúa todas las decisiones de diseño (Larman, 1999).

### **Patrones arquitectónicos**

Un patrón arquitectónico expresa un esquema de organización estructural esencial para un sistema de software, que consta de subsistemas, sus responsabilidades e interrelaciones y una serie de recomendaciones para organizar los distintos componentes (Larman, 2003).

Resuelven problemas arquitectónicos, adaptabilidad a requerimientos cambiantes, modularidad y acoplamiento. La solución que plantea es la creación de patrones de llamadas entre objetos, decisiones y criterios arquitectónicos, empaquetado de funcionalidad.

### Patrón MVC

Este patrón separa los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos. Frecuentemente se evidencia en aplicaciones web, donde la vista es la página HTML con la que interactúa el usuario, el modelo es la capa del dominio que representa la lógica de negocio y el controlador es el código que provee de datos dinámicos a la página (Larman, 2003). A continuación se muestra en la Figura la organización propuesta por Symfony 2.



## Capítulo 2

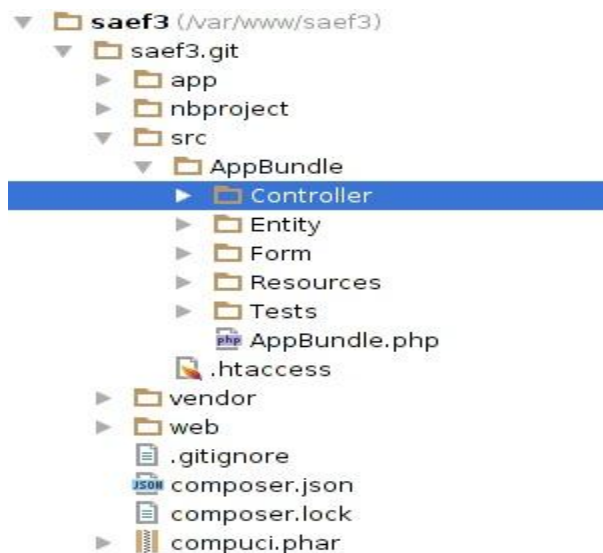


Figura 7. Organización propuesta por Symfony2. Fuente: Elaboración propia.

Modelo: está dividida en capa de acceso a datos y capa de abstracción de base de datos. La lógica de datos asegura la integridad de estos y permite derivar nuevos datos (Larman, 2003). A la capa de acceso a datos pertenece la clase Repository que se encuentran en el paquete "src/AppBundle/Repository". La clase Repository es donde se encuentran las consultas a la base de datos, aislándolas del modelo. En esta capa también se encuentra el ORM Doctrine que, mediante su lenguaje propio de consultas DQL, permite la independencia de la aplicación respecto al gestor de base de datos. Se muestra en la Figura la estructura anteriormente mencionada.

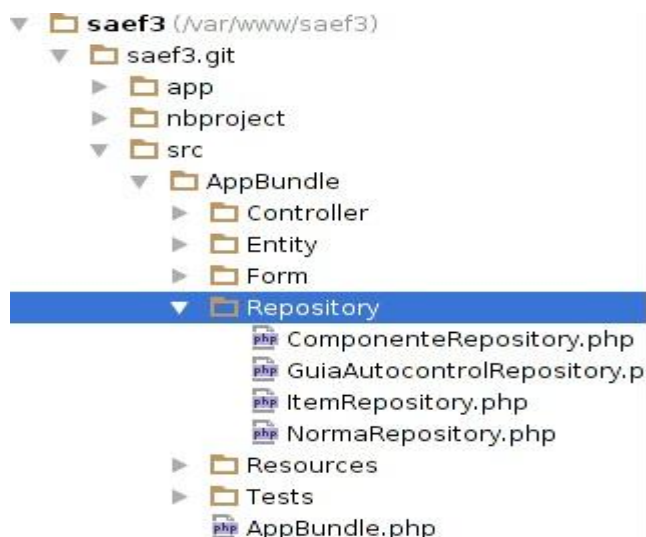


Figura 8. Organización propuesta por Symfony2. Clases Repository. Fuente: Elaboración propia.

A la capa de abstracción de base de datos pertenecen las entidades que representan las tablas de la base de datos previamente mapeadas por Doctrine. Estas se encuentran en el paquete

## Capítulo 2

“Entity” donde se encuentran las entidades mapeadas por Doctrine para el posterior manejo de los datos, en “src/AppBundle/Entity”.



Figura 9. Organización propuesta por Symfony2. Entidades. Fuente: Elaboración propia.

Vista: presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario (Larman, 2003). La plantilla `base.html.twig` que se encuentra dentro del paquete `/app/Resources/views`, en conjunto con las plantillas que se encuentran en `/src/AppBundle/Resources/views` son las que responden a esta capa, como se aprecia en la figura 9 respectivamente.

Estas transforman los datos obtenidos del modelo en vistas que son las que permiten al usuario interactuar con la aplicación. Para hacer esto se apoyan en la utilización del motor de plantillas Twig y el uso de los marcos de trabajos CSS, Bootstrap y JQuery respectivamente, que facilitan la construcción de interfaces bien acabadas y agradables a la vista del usuario.

## Capítulo 2

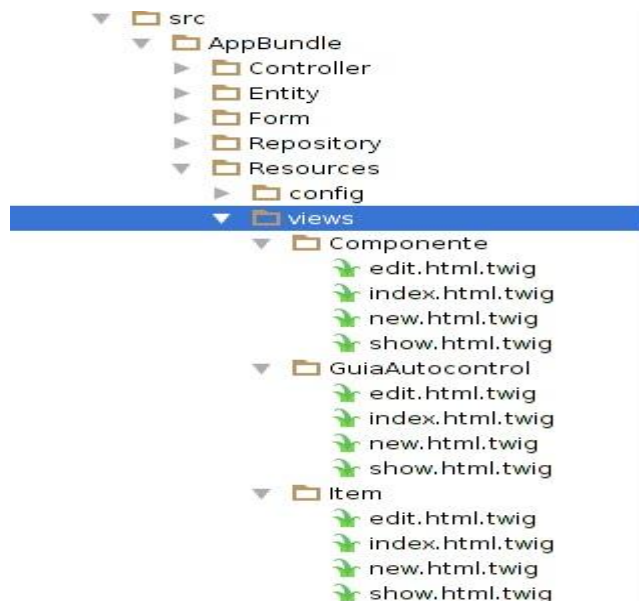


Figura 10. Organización propuesta por Symfony2. Vistas. Fuente: Elaboración propia.

Controlador: responde a eventos, usualmente acciones del usuario, e invoca cambios en el modelo y probablemente en la vista. Este recibe las solicitudes o peticiones realizadas por el usuario desde la vista, ejecuta las acciones pertinentes y devuelve una respuesta a la interfaz con los resultados de las operaciones realizadas (Larman, 2003).

Estas clases se encuentran dentro del paquete “/src/AppBundle/Controller” como se aprecia en la Figura y constituyen la parte de la aplicación que se encarga de realizar una funcionalidad completa y específica, además en ellos se establecen las reglas que deben cumplirse.

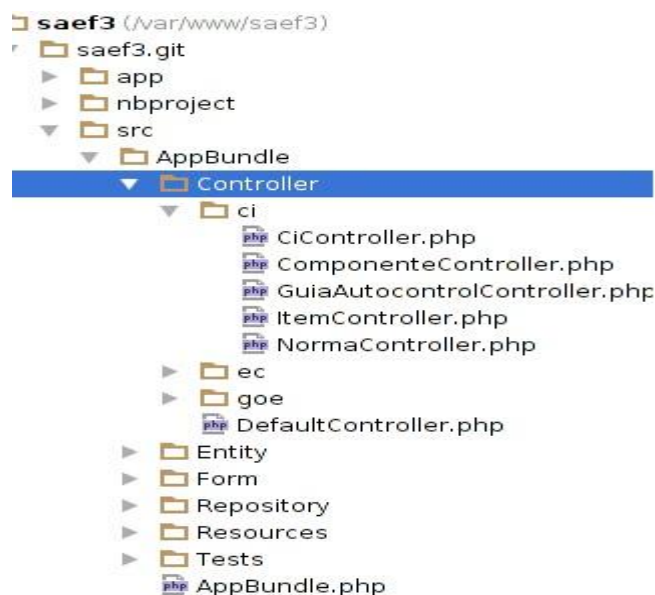


Figura 11. Organización propuesta por Symfony2. Controlador en Datos Principales. Fuente: Elaboración propia.

# Capítulo 2

## 2.3.5 Modelo de datos

Para la construcción del diseño del Modelo de datos se utilizó el patrón de llave subrogada, creando una llave primaria única para cada entidad en vez de usar un atributo identificador en el contexto dado. El citado patrón es necesario para un diseño cuidadoso. Un error pudiera conllevar a inconsistencias en la base de datos, consecuentemente, a producir información errónea en consultas.

El modelo de datos se encuentra en tercera Forma Normal (FN). Se describe el modelo de datos, cada una de las tablas y sus atributos. A continuación se presentan las descripciones de las tablas de mayor importancia para la investigación por los datos que persisten en ellas.

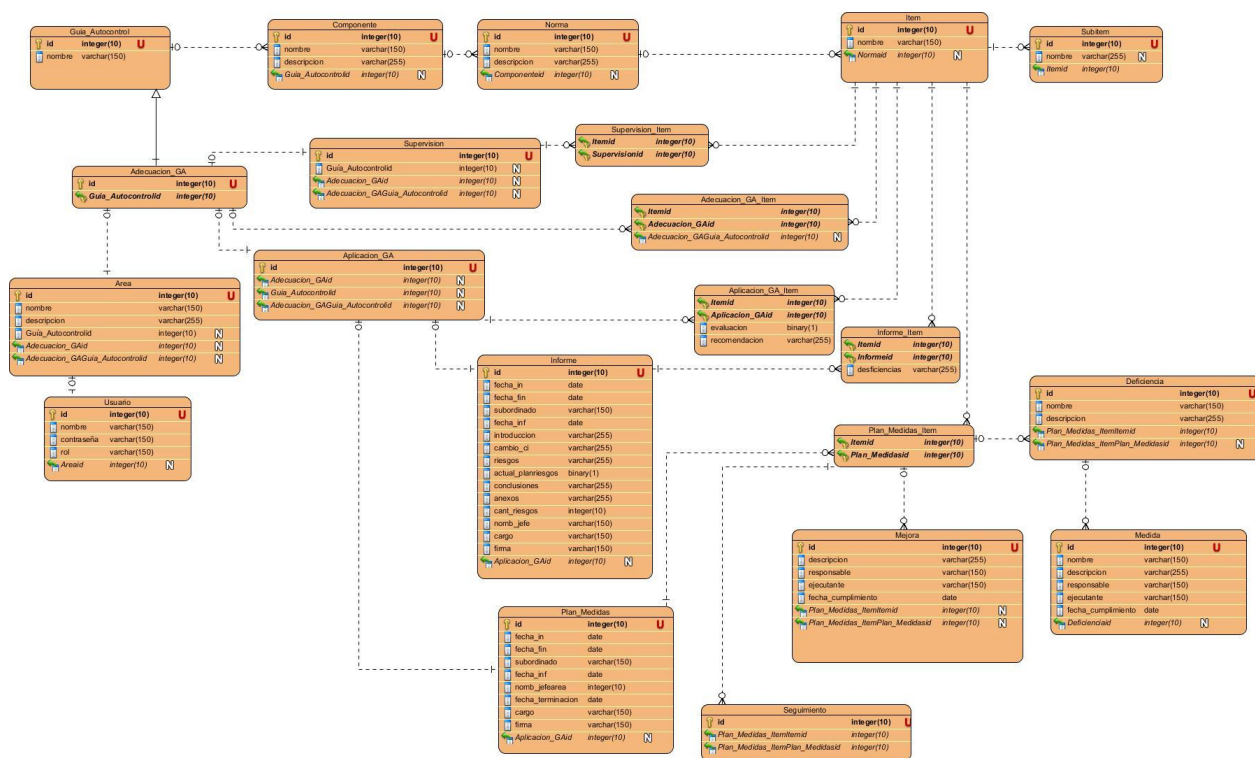


Figura 12. Modelo de Datos. Fuente: Elaboración propia.

### Descripción de las tablas de la BD

Nombre	Descripción
GuiaAutocontrol	Entidad que almacena la información de la Guía de Autocontrol.
Componente	Entidad que almacena la información de los componentes pertenecientes a la Guía de Autocontrol
Norma	Entidad que almacena la información de las normas pertenecientes a cada uno de los componentes de la Guía de Autocontrol.

## Capítulo 2












 Item	Entidad que almacena la información de los ítems pertenecientes a las normas de los componentes de la Guía de Autocontrol.
 Area	Entidad que almacena la información de las áreas a las que se les realiza el CI.
 Supervision	Entidad que almacena la información de la supervisión que se le realiza a un área determinada.
 AplicacionGA	Entidad que almacena la información de la aplicación de la Guía de Autocontrol a cada una de las áreas.
 AdecuacionGA	Entidad que almacena la información de la adecuación que se le realiza a la Guía de Autocontrol por cada una de las áreas
 Usuario	Entidad que almacena la información de los usuarios que estarán presentes en la aplicación.
 Informe	Entidad que almacena la información del informe que se generará luego de aplicar la Guía de Autocontrol.
 PlanMedidas	Entidad que almacena la información de las medidas que se aplicarán a los ítems no cumplidos identificados luego de aplicar la Guía de
 Deficiencia	Entidad que almacena la información de las deficiencias en cada uno de los ítems no cumplidos.
 Medida	Entidad que almacena la información de las medidas que se le aplicarán a las deficiencias encontradas dentro de los ítems no cumplidos.
 Mejora	Entidad que almacena la información de las mejoras que se le aplicarán a las deficiencias encontradas dentro de los ítems no cumplidos.

Tabla 6. Descripción de las tablas de la BD. Fuente: Elaboración propia

Luego de describir las entidades y las relaciones entre estas quedó conformada la base de datos, guardando la información referente al Módulo CI. Se procede a la validación del diseño a través de la aplicación de métricas para medir la calidad del mismo.

### **2.3.6 Validación del proceso de diseño**

#### **Métricas de diseño**

Para medir el diseño se decidió utilizar métricas básicas inspiradas en el estudio de la calidad del diseño orientado a objeto, principalmente la métrica RC (Relaciones entre clases) y la métrica TOC (Tamaño Operacional de Clase) que proponen Lorenz y Kidd (S. Pressman, 2010). Fueron seleccionadas en la presente investigación para la validación del diseño dado el nivel de aceptación que tienen en el ambiente donde son aplicadas y permiten visualizar si se distribuyen correctamente las asignaciones de responsabilidades entre las clases, verificándose así la cohesión y armonía entre las clases:

#### **Métrica Tamaño Operacional de Clase (TOC)**

Para aplicar la métrica TOC se tuvo en cuenta un conjunto de atributos de calidad como los que se muestran a continuación:

## Capítulo 2

Atributos de calidad	Definición
Responsabilidad	Consisten en asignarle una responsabilidad a una clase en un marco de modelado de dominio
Complejidad de implementación	Consiste en hacer compleja las pruebas del sistema
Reutilización	Consiste en el grado de reutilización presente en una clase

Tabla 7. Atributos de calidad de la métrica TOC. Fuente: Elaboración propia.

El tamaño operacional de una clase se puede determinar a partir del número de operaciones que posee, el resultado es tomado como umbral.

Si existen valores grandes de TOC, estos estarán demostrando que una clase puede tener demasiada responsabilidad, lo cual reduce la reutilización de la clase y complejizará la implementación. De forma contraria sucede si los valores TOC son pequeños. El próximo paso es calcular el promedio correspondiente a los umbrales, una vez obtenidos se procede a calcular la afectación de los parámetros establecidos.

Atributos de calidad	Clasificación	Criterio
Responsabilidad	Baja	Umbral $\leq$ Promedio
	Media	Promedio $<$ Umbral $\leq 2^*$ promedio
	Alta	Umbral $>$ $2^*$ promedio
Complejidad de implementación	Baja	Umbral $\leq$ Promedio
	Media	Promedio $<$ Umbral $\leq 2^*$ promedio
	Alta	Umbral $>$ $2^*$ promedio
Reutilización	Baja	Umbral $>$ $2^*$ promedio
	Media	Promedio $<$ Umbral $\leq 2^*$ promedio
	Alta	Umbral $\leq$ Promedio

Tabla 8. Rango de valores para medir la afectación de los atributos de calidad de la métrica TOC. Fuente: Elaboración propia.

A continuación se muestra el resultado de la aplicación de esta métrica al diseño.

## Capítulo 2

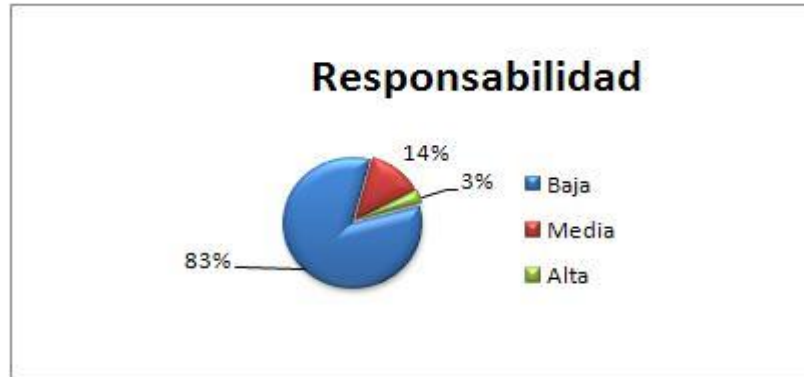


Figura 13. Representación de la evaluación de la métrica TOC para el atributo responsabilidad. Fuente: Elaboración propia.



Figura 14. Representación de la evaluación de la métrica TOC para el atributo complejidad. Fuente: Elaboración propia.



Figura 15. Representación de la evaluación de la métrica TOC para el atributo reutilización. Fuente: Elaboración propia.

La aplicación de la métrica TOC arrojó como resultado una baja responsabilidad y complejidad y una alta reutilización, permitiendo la eliminación de clases repetidas, teniendo en cuenta que

## Capítulo 2

más de la mitad de las clases poseen baja dependencia respecto a otras, y el sistema tendrá una compleja implementación.

### **Métrica Relaciones entre Clases (RC)**

Está dado por el número de relaciones de uso de una clase con otra. A continuación se muestra una serie de tablas encaminadas a un mejor entendimiento de la utilización de esta métrica.

Atributos de calidad	Definición
Acoplamiento	Consiste en el grado de dependencia o interconexión de una clase con otras.
Complejidad de mantenimiento	Consiste en el grado de esfuerzo necesario a realizar para desarrollar un arreglo, una mejora o una rectificación de algún error de un diseño de software.
Reutilización	Consiste en el grado de reutilización presente en una clase, dentro de un diseño de software.
Cantidad de pruebas	Consiste en el número necesario de pruebas a realizar.

*Tabla 9. Atributos de calidad de la métrica RC. Fuente: Elaboración propia.*

Para determinar el grado de afectación de los atributos de calidad que mide la métrica RC es necesario determinar la cantidad de relaciones de uso (CRU) que posee cada una de las clases a medir. Una vez determinada la CRU, se procede a calcular el promedio de las mismas y teniendo ambos valores según los criterios expuestos a continuación.

Atributos de calidad	Clasificación	Criterio
Acoplamiento	Ninguna	$CRU = 0$
	Baja	$CRU = 1$
	Media	$CRU = 2$
	Alta	$CRU > 2$
	Baja	$CRU \leq \text{Promedio}$
Complejidad de mantenimiento	Baja	$CRU \leq \text{Promedio}$
	Media	$\text{Promedio} < CRU \leq 2 * \text{promedio}$
	Alta	$CRU > 2 * \text{promedio}$
Reutilización	Baja	$CRU > 2 * \text{promedio}$
	Media	$\text{Promedio} < CRU \leq 2 * \text{promedio}$



## Capítulo 2

	Alta	CRU $\leq$ Promedio
Cantidad de pruebas	Baja	CRC $\leq$ Promedio
	Media	Promedio $\leq$ CRC $<$ 2*Promedio
	Alta	CRC $\geq$ 2*Promedio

Tabla 10. Rango de valores para medir la afectación de los atributos de calidad de la métrica RC. Fuente: Elaboración propia.

A continuación se muestra el resultado de la aplicación de esta métrica al diseño.



Figura 16. Representación de la evaluación de la métrica RC en el atributo Acoplamiento. Fuente: Elaboración propia.



Figura 17. Representación de la evaluación de la métrica RC en el atributo complejidad de mantenimiento. Fuente: Elaboración propia.

## Capítulo 2



Figura 18. Representación de la evaluación de la métrica RC en el atributo cantidad de pruebas. Fuente: Elaboración propia.



Figura 19. Representación de la evaluación de la métrica RC en el atributo reutilización. Fuente: Elaboración propia.

Los resultados obtenidos durante la evaluación del instrumento de medición de la métrica RC demuestran que los atributos de calidad fueron evaluados satisfactoriamente confirmando que las clases están diseñadas correctamente. Presentan una elevada reutilización, un bajo acoplamiento, poca complejidad de mantenimiento y poca cantidad de pruebas que implica que se necesitará menos esfuerzo a la hora de realizar pruebas unitarias a estas clases.

### 2.4 Conclusiones parciales

- ✓ A partir de la arquitectura del sistema y la realización de los diagramas de clases se obtuvo una propuesta para realizar la implementación del módulo CI, de manera que se obtenga un diseño con calidad, mediante el uso de los patrones de diseño.
- ✓ Con el Análisis y Diseño de la solución se obtuvieron 35 requisitos funcionales para el sistema a partir de la aplicación de las técnicas de validación de requisitos y se generaron los artefactos definidos por la metodología utilizada, obteniéndose entre ellos la Descripción de los procesos de negocio, Especificación de requisitos de software,

## Capítulo 2

Historias de usuario, los Diagramas de clases del diseño, Prototipos de interfaz gráfica de usuario, entre otros artefactos.

- ✓ Los resultados obtenidos durante la aplicación de las métricas para la validación del diseño propuesto para el módulo CI demuestran que se encuentra dentro de los niveles de calidad requeridos. Además, se sentaron las bases para poder realizar la implementación y validación de la solución.

# Capítulo 3

## CAPÍTULO 3. IMPLEMENTACIÓN Y VALIDACIÓN DE LA SOLUCIÓN

### Introducción

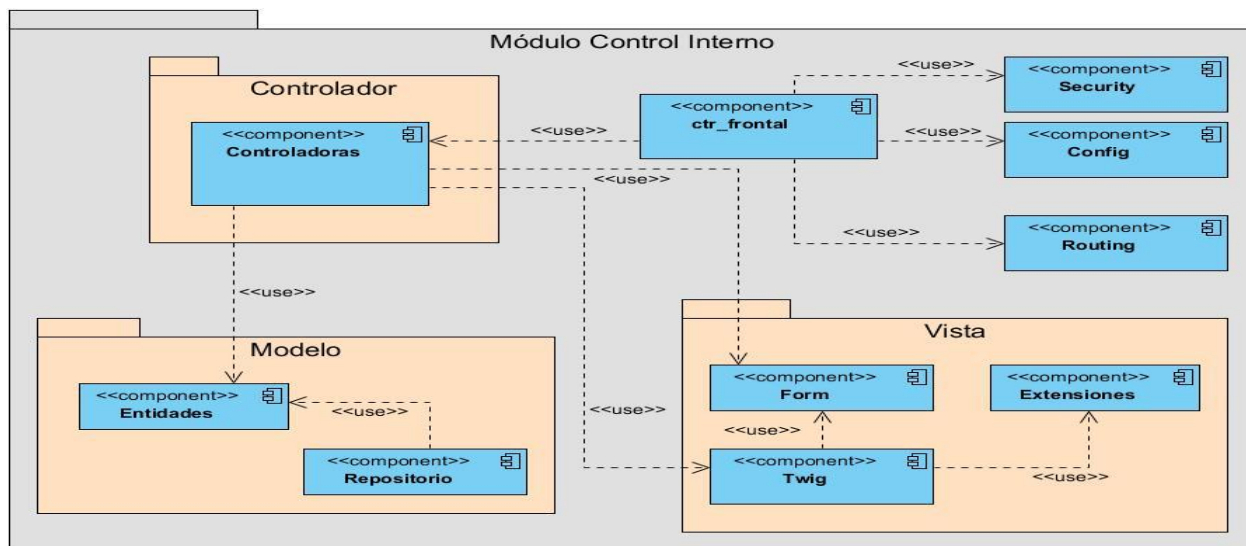
Teniendo en cuenta el capítulo anterior, es necesario definir cómo se desarrollará el sistema, pasando al presente capítulo en el que se detallarán aspectos de la implementación como la elaboración del diagrama de componentes y despliegue y las pruebas realizadas al Módulo CI. Se comprobará la calidad de los resultados mediante los artefactos elaborados durante el flujo de trabajo de pruebas y se validarán las variables de la investigación mediante la técnica de satisfacción al cliente ladov y las pruebas de aceptación.

### 3.1 Implementación

En esta disciplina a partir de los resultados del Análisis y diseño se construye el sistema (Rodríguez, 2014). Se describe cómo los elementos del modelo de diseño se implementan en términos de componentes, entre los que se encuentran datos, archivos, ejecutables y código.

#### 3.1.1 Diagrama de componentes

El diagrama de componentes representa cómo un sistema de software es dividido en componentes, ilustrando las dependencias entre estos. Este tipo de diagrama contiene además interfaces y sus relaciones, pudiendo contener también paquetes que se utilizan para agrupar elementos del modelo. Un diagrama de componentes permite visualizar con facilidad la estructura general del sistema y el comportamiento del servicio que estos componentes proporcionan y utilizan a través de las interfaces (Microsoft, 2014).



# Capítulo 3

Figura 20. Diagrama de componentes Fuente: Elaboración propia

El diagrama de componentes muestra la organización y las dependencias entre el conjunto de componentes que presenta el Módulo CI, sin ofrecer el diseño e implementación internas de los mismos. Permite la configuración correcta para llevar a cabo la funcionalidad deseada para el módulo.

## 3.1.2 Diagrama de despliegue

El diagrama de despliegue se realiza para un mejor entendimiento de cómo se ejecutará el sistema a nivel de hardware. Es un diagrama de objetos que representa la distribución física del sistema. El diagrama de despliegue es empleado para modelar el hardware utilizado en las implementaciones de sistemas y las relaciones entre sus componentes (Sarmiento, 2013).

A continuación se presenta el diagrama de despliegue de la solución. El mismo está compuesto por una PC<sup>14</sup> cliente conectada a través del protocolo de transferencia de hipertexto HTTPS<sup>15</sup> por el puerto 443 a un servidor web que se comunica con el servidor de bases de datos mediante el protocolo de Internet TCP/IP<sup>16</sup>, por el puerto 5432, además de una impresora que se conecta mediante USB<sup>17</sup> a dicha PC cliente.

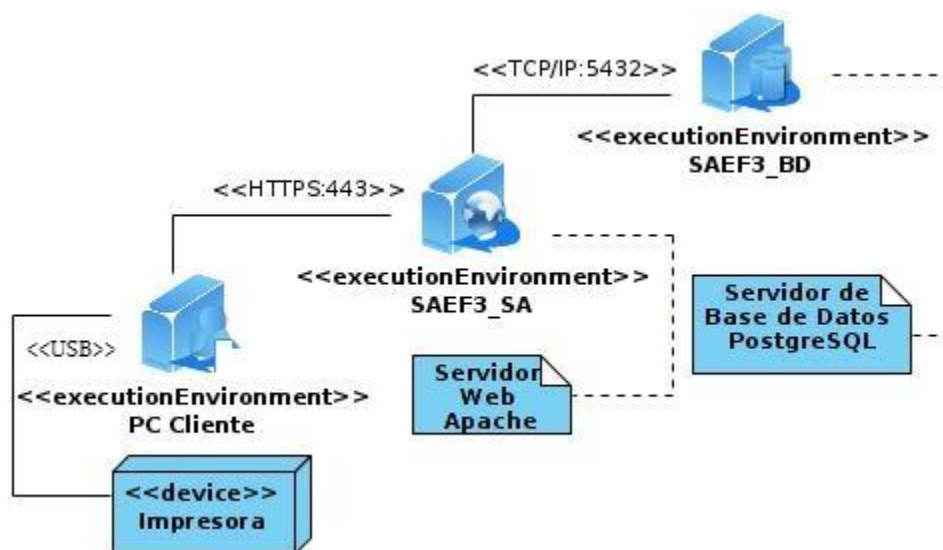


Figura 21. Diagrama de despliegue. Fuente:Elaboración propia

<sup>14</sup>Computadora Personal

<sup>15</sup>Protocolo De Transferencia De Hipertexto

<sup>16</sup>Transmisión de protocolo de Control y protocolo de Internet

<sup>17</sup>Serial Bus Universal

## Capítulo 3

### 3.1.3 Estándares de codificación

Los estándares de codificación son una serie de convenciones que deben seguir los desarrolladores manteniendo buenas prácticas definidas por la ingeniería de software, para obtener un código fácil de comprender y de alta calidad. Facilitan el mantenimiento de una aplicación. Posibilitan que un equipo de programadores mantenga un código legible sobre el que se efectuarán luego revisiones; con el objetivo de regular la calidad de la implementación estableciendo un estándar de desarrollo común por el cual se rige la programación del sistema (Microsoft, 2015).

Estilo de Codificación Utilizado:

#### División de líneas

Cuando una expresión ocupe más de una línea, esta se podrá romper o dividir en función de los siguientes criterios:

- ✓ Tras una coma.
- ✓ Se recomienda las rupturas de nivel superior a las de nivel inferior.
- ✓ Alinear la nueva línea con el inicio de la expresión al mismo nivel que la línea anterior.

Ejemplos:

```
private function CreateForm(AdecuacionGA $entity)
{
    $form = $this->createForm (new AdecuacionGAType (), $entity,
    array ( 'action' =>$this->generateUrl ('adecuacionga_create'),
    'attr' =>array ('id' =>'adecuacion_form'),
    'method' =>'POST',
    ));
    return $form;
}
```

#### Una declaración por línea

Se recomienda el uso de una declaración por línea, promoviendo así el uso de comentarios.

Ejemplo:

```
$items = 0;
$subitems = array ();
```

#### Declaración de clases / interfaces

Durante el desarrollo de clases / interfaces se deben seguir las siguientes reglas de formato:

- ✓ No incluir ningún espacio entre el nombre del método y el paréntesis inicial del listado de parámetros.

## Capítulo 3

- ✓ El carácter inicio de bloque ("{"") debe aparecer al final de la línea que contiene la sentencia de declaración.
- ✓ El carácter fin de bloque ("}") se sitúa en una nueva línea tabulada al mismo nivel que su correspondiente sentencia de inicio de bloque, excepto cuando la sentencia sea nula, en tal caso se situará detrás de "{".
- ✓ Los métodos se separarán entre sí mediante una línea en blanco.

```
class AdecuacionGA extends GuiaAutocontrol {
protected $area;
protected $informe;
public function __construct() {
$this->adecuacionesGaltem = new Doctrine\Common\Collections\ArrayCollection();
$this->adecuacionesGaSubitem = new \Doctrine\Common\Collections\ArrayCollection();
$this->componentes = new \Doctrine\Common\Collections\ArrayCollection();
}
...
}
```

### Clases e interfaces

Los nombres de clases deben ser sustantivos y deben tener la primera letra en mayúscula. Si el nombre es compuesto, cada palabra componente deberá comenzar con mayúscula. Los nombres serán simples y descriptivos. Debe evitarse el uso de acrónimos o abreviaturas, salvo en aquellos casos en los que dicha abreviatura sea más utilizada que la palabra que representa.

```
class GuiaAutocontrol{
...
}
```

### Métodos

Los métodos deben ser verbos escritos en minúscula. Cuando el método esté compuesto por varias palabras cada una de ella tendrá la primera letra en mayúscula.

```
public function adecuarGuiaAutocontrol ();
```

### Variables

Las variables se escribirán siempre en minúscula. Las variables compuestas tendrán la primera letra de cada palabra componente en mayúscula.

```
private $guiaAutocontrol;
```

# Capítulo 3

Los estándares de codificación son una serie de convenciones que deben seguir los desarrolladores manteniendo buenas prácticas definidas por la ingeniería de software, para obtener un código fácil de comprender y de alta calidad. Facilitan el mantenimiento de una aplicación. Posibilitan que un equipo de programadores mantenga un código legible sobre el que se efectuarán luego revisiones; con el objetivo de regular la calidad de la implementación estableciendo un estándar de desarrollo común por el cual se rige la programación del sistema

Con el uso de los estándares de codificación la implementación del Módulo CI se realizó con un código legible y fácil de comprender, con facilidades de mantenimiento y posteriores revisiones.

## **3.2 Pruebas de software**

Las pruebas de software son una serie de actividades que se realizan con el propósito de encontrar los posibles fallos de implementación, calidad o usabilidad de un programa u ordenador; probando el comportamiento del mismo. Las pruebas van dirigidas a componentes del sistema o al sistema de software en su totalidad, con el objetivo de medir el grado en que la aplicación cumple con los requerimientos. Al Módulo CI se le aplicarán pruebas internas y de aceptación definidas por la metodología.

### **3.2.1 Pruebas internas**

En esta disciplina se verificará el resultado de la implementación probando cada construcción, incluyendo tanto las construcciones internas como intermedias, así como las versiones finales. Se desarrollarán artefactos de prueba como diseños de casos de prueba (Rodríguez, 2014). Las pruebas realizadas al módulo fueron las pruebas de caja negra y caja blanca.

#### **Prueba de caja negra**

Las pruebas de caja negra se refieren a las pruebas que se llevan a cabo sobre la interfaz del software, por lo que los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce una salida correcta, así como que la integridad de la información externa se mantiene (M. Moreno, y otros, 2006). Esta prueba examina algunos aspectos del modelo, fundamentalmente del sistema, sin tener mucho en cuenta la estructura interna del software.

Las pruebas de caja negra pretenden encontrar errores como (M. Moreno, y otros, 2006):

- ✓ Funciones incorrectas o ausentes.
- ✓ Errores en la interfaz.
- ✓ Errores en la estructura de datos o en accesos a base de datos externas.



## Capítulo 3

- ✓ Errores de rendimiento.
- ✓ Errores de inicialización y de terminación

Los casos de prueba de caja negra pretenden demostrar que(M. Moreno, y otros, 2006):

- ✓ Las funciones del software son operativas.
- ✓ La entrada se acepta de forma correcta.
- ✓ Se produce una salida correcta.
- ✓ La integridad de la información externa se mantiene.

Para desarrollar la prueba de caja negra existen varias técnicas, entre ellas están(S. Pressman, 2000):

- ✓ Técnica de la Partición de Equivalencia: esta técnica divide el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del software.
- ✓ Técnica del Análisis de Valores Límites: esta técnica prueba la habilidad del programa para manejar datos que se encuentran en los límites aceptables.
- ✓ Técnica de Grafos de Causa-Efecto: es una técnica que permite al encargado de la prueba validar complejos conjuntos de acciones y condiciones.

En la presente investigación se utiliza la técnica Partición de Equivalencia ya que es una de las más efectivas, pues permite examinar los valores válidos e inválidos de las entradas existentes en el módulo, descubre de forma inmediata una clase de errores que, de otro modo, requerirían la ejecución de muchos casos antes de detectar el error genérico.

Se muestra seguidamente un ejemplo de casos de prueba para el módulo:

# Capítulo 3

Escenario	Descripción	Nombre	Área	Ítem	Respuesta del sistema	Flujo central
EC 11 Adicionar Adecuación de la Guía de Autocontrol	Permite adicionar una Adecuación de la Guía de Autocontrol.  Todos los campos son obligatorios.	V AdProgramación	V Dpto Programación	V 3, 5, 7, 14, 54	El sistema muestra el mensaje de confirmación: "La Adecuación ha sido creada satisfactoriamente.". El sistema cierra la interfaz. Se actualiza la lista de Adecuaciones.	1. Seleccionar del menú la opción "Control". 2. Selecciona la opción "Adecuación". 3. Seleccionar la opción "Adicionar". 4. Se muestra una pantalla con los siguientes datos a llenar: nombre, área, ítem. 6. Una vez introducidos los valores seleccionar la opción "Guardar". 7. Se actualiza el listado de las Adecuaciones.
		I	V Dpto Programación	V 3, 5, 7, 14, 54	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos o en blanco.	
		V AdProgramación	I	V 3, 5, 7, 14, 54	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos o en blanco.	
		V AdProgramación	V Dpto Programación	I	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos o en blanco.	
EC 12 Modificar Adecuación de la Guía de Autocontrol	Permite modificar valores existentes de una Adecuación seleccionada.	V AdProgramación	V Dpto Programación	V 3, 5, 7, 14, 54	El sistema muestra el mensaje de confirmación: "La Adecuación ha sido modificada satisfactoriamente.". El sistema cierra la interfaz. Se actualiza la lista de adecuaciones.	1. Ejecutar el flujo central del EC 11.(1,2) 2. Seleccionar la Adecuación que se desea modificar. 3. Seleccionar la opción "Modificar". 4. Se muestra una pantalla con los valores de la adecuación seleccionada. 5. Una vez introducidos los cambios deseados seleccionar entonces la opción "Modificar". 6. Se actualiza el listado de las Adecuaciones.
		I	V Dpto Programación	V 3, 5, 7, 14, 54	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos o en blanco.	
		V AdProgramación	I	V 3, 5, 7, 14, 54	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos o en blanco.	
		V AdProgramación	V Dpto Programación	I	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos o en blanco.	
EC 13 Eliminar Adecuación de la Guía de Autocontrol	Permite eliminar la(s) Adecuación(es) deseada(s).	V AdProgramación	V Dpto Programación	V 3, 5, 7, 14, 54	El sistema muestra el mensaje de confirmación: "Adecuación eliminada." El sistema cierra la interfaz.	1. Ejecutar el flujo central del EC 11.(1, 2) 2. Seleccionar la(s) Adecuación(es) que se desea(n) eliminar. 3. Seleccionar la opción "Mostrar". 4. Seleccionar la opción "Eliminar". 5. Se muestra un mensaje preguntándole al usuario que si está seguro que desea eliminar la(s) Adecuación(es) seleccionada(s). 6. Seleccionar entonces la opción "Aceptar". 7. Se actualiza el listado de las Adecuaciones.

Tabla 11. Caso de Prueba Gestionar adecuación de la Guía de Autocontrol. Fuente: Elaboración propia.

Para consultar los restantes casos de pruebas ver Anexo 5.

Con la aplicación de las pruebas de caja negra se validaron las funcionalidades implementadas en el Módulo CI. Estas se realizaron en tres iteraciones. En la primera iteración se encontraron 20 no conformidades donde 12 fueron de ortografías y 8 de validaciones, en la segunda iteración se encontraron 5 de ortografías y 3 de validaciones y finalmente se comprobó en la tercera iteración que el módulo estaba libre de no conformidades culminando así la fase de pruebas internas. En las mismas se detectaron errores mayormente de faltas de ortografías en los distintos tipos de mensajes, que no coincidían con los descritos en los casos de prueba, así como errores en las validaciones.

## Capítulo 3

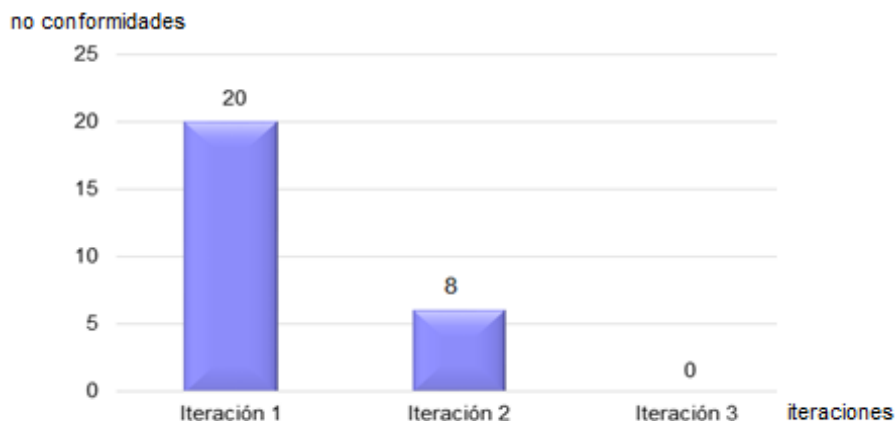


Figura 22. Resultado de no conformidades de la prueba de caja negra.

### **Prueba de caja blanca**

Las pruebas de caja blanca se basan en el minucioso examen de los detalles procedimentales. Se comprueban los caminos lógicos del software proponiendo casos de prueba que examinen que están correctas todas las condiciones y/o bucles para determinar si el estado real coincide con el esperado o afirmado. Esto genera gran cantidad de caminos posibles por lo que hay que dedicar esfuerzos a la determinación de las condiciones de prueba que se van a verificar (M. Moreno, y otros, 2006).

Entre las técnicas de caja blanca será utilizada la técnica “Camino Básico” que permite al diseñador de casos de prueba obtener una medida de la complejidad lógica de un diseño procedimental y usar esa medida como guía para la definición de un conjunto básico de caminos de ejecución (M. Moreno, y otros, 2006), además que en su completo funcionamiento se realizan las restantes técnicas como la de Bucles, de Condiciones y de Flujo de Datos. Para aplicar esta prueba se han definido una serie de pasos a seguir que a continuación se describen:

1. **Realizar la notación del grafo de flujo:** usando el código como base se realiza la representación del grafo de flujo (grafo del programa<sup>18</sup>), mediante una sencilla notación. Cada construcción estructurada tiene su correspondiente símbolo.
  - ✓ **Nodo:** cada círculo, denominado nodo, representa una o más sentencias procedimentales.

<sup>18</sup>Sirve como herramienta útil para ilustrar el método.

## Capítulo 3

- ✓ Arista: las flechas del grafo de flujo, denominadas aristas, representan flujo de control y son análogas a las flechas del diagrama de flujo.
  - ✓ Región: las áreas delimitadas por aristas y nodos se denominan regiones.
2. **Calcular la complejidad ciclomática:** es una métrica que proporciona una medición cuantitativa de la complejidad lógica de un programa. El valor calculado define el número de caminos independientes<sup>19</sup> del conjunto básico de un programa y proporciona el límite superior de pruebas que se deben realizar para asegurar que se ejecuta cada sentencia al menos una vez. Se calcula de tres formas:
    - ✓ La complejidad ciclomática,  $V(G)$ , de un grafo de flujo  $G$  se define como:  $V(G) = A - N + 2$ , donde  $A$  es el número de aristas del grafo de flujo y  $N$  es el número de nodos del mismo
    - ✓ La complejidad ciclomática,  $V(G)$ , de un grafo de flujo  $G$  también se define como  $V(G) = P + 1$ , donde  $P$  es el número de nodos predicado (son los nodos de los cuales parten dos o más aristas) contenidos en el grafo de flujo  $G$ .
    - ✓ El número de regiones ( $R$ ) del grafo de flujo coincide con la complejidad ciclomática.
  3. **Determinar un conjunto básico de caminos linealmente independientes:** el valor de  $V(G)$  permite determinar el número de caminos linealmente independientes de la estructura de control del programa.
  4. **Obtener los casos de prueba:** se realizan los casos de pruebas que forzarán la ejecución de cada camino del conjunto básico. (Aguilera, y otros, 2012)

### Aplicación de la prueba

A continuación se enumeran las sentencias de código del procedimiento realizado sobre el método "Crear adecuación de la Guía de Autocontrol", a partir del cual se creará el grafo de flujo para calcular la complejidad ciclomática del mismo.

---

<sup>19</sup>Camino del programa que introduce un nuevo conjunto de sentencias de proceso o una nueva condición. Está constituido por al menos una arista que no haya sido recorrida anteriormente a la definición del camino.

# Capítulo 3

```
public function createAction(Request $request)
{
    if($request->isXmlHttpRequest()) { 1
        $em = $this->getDoctrine()->getManager(); 2
        $data = $request->request->all(); 2

        $data = json_decode($data['datos'], true); 2
        $area = $em->getRepository('AppBundle:Area')->find($data['area']); 2

        $entity = new AdecuacionGA(); 2
        $entity->setNombre($data['nombre']); 2
        $entity->setArea($area); 2
        $em->persist($entity); 2

        $query = $em->createQuery("SELECT i FROM AppBundle:Item i WHERE i.id IN (:items)")
            ->setParameters(array('items' => $data['items'])); 2
        $items = $query->getResult(); 2

        $query = $em->createQuery("SELECT s FROM AppBundle:Subitem s WHERE s.id IN (:subitems)")
            ->setParameters(array('subitems' => $data['subitems'])); 2
        $subitems = $query->getResult(); 2

        if(count($items)>0) { 3
            foreach ($items as $item) { 4
                $entity->addItem($item); 5
                $em->persist($entity); 5
            }

            foreach ($subitems as $subitem) { 6
                $entity->addSubitem($subitem); 7
                $em->persist($entity); 7
            }
        }
        $em->flush(); 8
    }
    return new JsonResponse(array('success' => true, 'url' => $this->generateUrl('adecuacionga_show',
        array('id' => $entity->getId()), UrlGeneratorInterface::ABSOLUTE_URL)); 8
    }
}
```

Figura 23. Código fuente del método Crear adecuación de la Guía de Autocontrol. Fuente: Elaboración propia

A continuación se muestra el grafo de flujo asociado al código anterior:

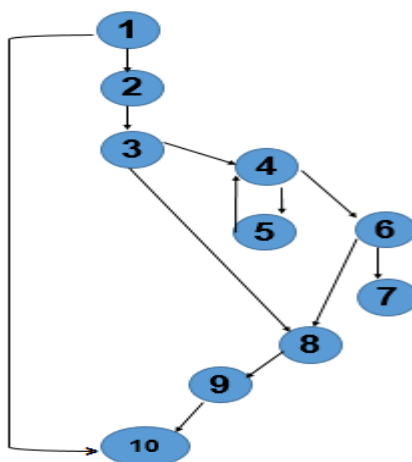


Figura 24. Grafo de flujo del método Crear adecuación de la Guía de Autocontrol. Fuente: Elaboración propia

Luego se calcula la complejidad ciclomática, mediante los métodos mencionados anteriormente:

## Capítulo 3

**Fórmula 1:**  $V(G) = (A - N) + 2$ .

**Resultado:**  $V(G) = (13 - 10) + 2, V(G) = 5$

**Fórmula 2:**  $V(G) = P + 1$ .

**Resultado:**  $V(G) = (4 + 1), V(G) = 5$

**Fórmula 3:**  $V(G) = R$ .

**Resultado:**  $V(G) = 5$

Cada una de las fórmulas  $V(G)$  representa el valor del cálculo de la complejidad ciclomática. A partir de los resultados obtenidos en cada una de ellas, se determina que la complejidad ciclomática del código analizado es 5, que a su vez es el número mínimo de caminos y de casos de prueba que se le pueden aplicar a dicho código. A continuación se muestran los caminos básicos por donde puede circular el flujo:

Número	Caminos básicos
1	1-2-3-4-5-4-6-7-6-8-9-10
2	1-2-3-4-6-7-6-8-9-10
3	1-2-3-4-5-4-6-8-9-10
4	1-2-3-8-9-10
5	1-10

*Tabla 12. Caminos básicos del flujo del método Crear adecuación de la Guía de Autocontrol. Fuente: Elaboración propia.*

Después de haber extraído los caminos básicos del flujo, se procede a ejecutar los casos de pruebas para este procedimiento, realizando al menos un caso de prueba por cada camino. Para realizarlos es necesario cumplir con las siguientes exigencias:

- ✓ **Descripción:** se hace la entrada de datos necesaria, validando que ningún parámetro obligatorio pase nulo al procedimiento o no se entre algún dato erróneo.
- ✓ **Condición de ejecución:** se especifica cada parámetro para que cumpla una condición deseada para ver el funcionamiento del procedimiento.
- ✓ **Entrada:** Se muestran los parámetros que entran al procedimiento.
- ✓ **Resultados Esperados:** se expone el resultado que se espera devuelva el procedimiento.

## Capítulo 3

A continuación se muestra el diseño de uno de los casos de prueba realizado al primer camino básico. Para consultar los demás casos de pruebas ver Anexo 5.

### Caso de prueba para el camino básico 1:

Camino:1-2-3-4-5-4-6-7-6-8-9-10	
<b>Descripción</b>	Los datos de entrada son correctos y cumplen con el formato indicado.
<b>Condición de ejecución</b>	<b>Campos válidos:</b> El nombre de la adecuación será Adecuación del Decanato. El área será Decanato. El ítem será la elaboración del plan anual de actividades. El Subítem será las funciones definidas de cada área y puesto de trabajo.
<b>Entrada</b>	<b>Nombre de la Adecuación:</b> Adecuación del Decanato. <b>Área:</b> Decanato. <b>Ítem:</b> La elaboración del plan anual de actividades. <b>Subítem:</b> Las funciones definidas de cada área y puesto de trabajo.
<b>Resultados esperados</b>	Se espera que se cree correctamente la adecuación.

Tabla 13. Caso de Prueba del camino básico 1. Fuente: Elaboración propia

Luego de aplicar los distintos casos de pruebas derivados de la prueba de caja blanca “Camino Básico”, se pudo comprobar que el flujo de trabajo de la función analizada está correcto pues, según las condiciones de entrada especificadas, se obtienen los resultados esperados.

### 3.2.2 Pruebas de Aceptación

Es la disciplina que realizará la prueba final antes del despliegue del módulo. Su objetivo es verificar que el software está listo y que puede ser usado por usuarios finales para ejecutar aquellas funciones y tareas para las cuales el software fue construido (Rodríguez, 2014).

Las pruebas de aceptación son realizadas por el cliente verificando que el módulo cumple con los requerimientos especificados para validar las variables por las que se decidió iniciar la investigación.

### Validación de las variables de la investigación

## Capítulo 3

A continuación se explica cómo se evalúan las variables disponibilidad y control con el uso del Módulo CI.

Antes de conocer el resultado de la validación de las variables es necesario definir en primer paso los conceptos de disponibilidad y control de la información.

La **disponibilidad de la información** asegura que el acceso a los datos o a los recursos de información por personal autorizado se produzca correctamente. Es decir, la disponibilidad garantiza que los sistemas funcionan cuando se les necesita (Belt Ibérica S.A, 2013).

Se entiende por **control**, medir y corregir las actividades de subordinados para asegurarse que los eventos se ajustan a los planes; es el proceso de verificar para determinar si se están cumpliendo los planes o no, si existe un progreso hacia los objetivos y metas. El control es necesario para corregir cualquier desviación (Comunidad Zona económica, 2013).

Variables	Dimensiones	Proceso actual	Proceso con la utilización del módulo implementado
Disponibilidad de la información	Acceso a datos	La documentación se encuentra en formato duro, lo que puede provocar desactualización y pérdida de la información.	La documentación está almacenada de forma digital en la herramienta, evitando la desactualización. La información guardada persiste en una base de datos, evitándose la pérdida de información.
		La documentación se encuentra almacenada en estantes, lo que dificulta su búsqueda para la consulta de su información.	La documentación se encuentra disponible en el sistema, tanto durante la ejecución de los procesos como una vez culminado los mismos.
		El acceso a los datos se realiza a través del personal del VDEA.	El acceso a los datos puede ser realizado por los usuarios según el permiso que tengan



## Capítulo 3

			en el sistema, en cualquier momento.
Control de la información	Adecuación/Supervisión	Es engorroso adecuar o supervisar eficientemente a las áreas subordinadas considerando que son varias y las adecuaciones o supervisiones se realizan manualmente.	Con el módulo desarrollado se facilita la adecuación o supervisión a la información gestionada por las áreas de un modo más ágil.
		Generar los reportes solicitados al VDEA se dificulta debido a la gran cantidad de información que controlan, y la omisión de datos trae consigo errores en la información.	Los reportes son generados por el sistema mediante llamadas a la base de datos que permiten obtener la información necesaria.

*Tabla 14. Resultado de la validación de las variables disponibilidad y control de la información. Fuente: Elaboración propia.*

Para calcular el grado de satisfacción del cliente con la solución del Módulo CI respecto al control y disponibilidad de la información, se aplicó la técnica ladov. Durante la valoración participaron como clientes, los Vicedecanos de Economía y Administración de todas las facultades de la UCI, para un total de siete personas.

### Técnica ladov

La técnica ladov se compone de cinco preguntas claves: tres cerradas y dos abiertas, las cuales se reformulan en la investigación para valorar el grado de satisfacción de los clientes sobre la tema en específico. Una vez establecidas las preguntas se conforma el “cuadro lógico de ladov” y el número resultante de la interrelación de las tres preguntas, indica la posición de los sujetos en la escala de satisfacción. La escala de satisfacción está dada por los criterios (N. V., 1970).

1. Máxima satisfacción.
2. Más satisfecho que insatisfecho.
3. No definida.
4. Más insatisfecho que satisfecho.

## Capítulo 3

5. Máxima insatisfacción.
6. Contradictoria.

¿Le gustaría hacer uso del Módulo Control Interno del SAEF3, para desarrollar los procesos asociados al control que se realiza a cada una de las áreas por parte del VDEA?	¿Considera usted oportuno continuar ejecutando los procesos asociados al control interno que se realiza a cada una de las áreas por parte del VDEA manualmente, a pesar del gran volumen de información que se genera?								
	No			No sé			Sí		
	¿El Módulo Control Interno del SAEF3 contribuye a mejorar el control y disponibilidad de la información del VDEA acorde a sus necesidades?								
	Sí	No sé	No	Sí	No sé	No	Sí	No sé	No
Me gusta mucho	1	2	6	2	2	6	6	6	6
No me gusta mucho	2	2	3	2	3	3	6	3	6
Me da lo mismo	3	3	3	3	3	3	3	3	3
Me disgusta más de lo que me gusta	6	3	6	3	4	4	3	4	4
No me gusta nada	6	6	6	6	4	4	6	4	6
No sé qué decir	2	3	6	3	3	3	6	3	4

Tabla 15. Cuadro lógico de ladov. Fuente: Elaboración propia.

Para obtener el índice de satisfacción grupal (ISG) se trabaja con los diferentes niveles de satisfacción que se expresan en la escala numérica que oscila entre +1 y - 1 de la siguiente forma:

Índice de satisfacción	Escala
Máxima satisfacción	+1
Más satisfecho que insatisfecho	0,5
No definido y contradictorio	0
Más insatisfecho que satisfecho	-0,5
Máxima insatisfacción	-1

Tabla 16. Resultado de aplicación de la técnica ladov. Fuente: Elaboración propia.

La satisfacción grupal (ISG) se calcula por la siguiente fórmula:

$$ISG = \frac{A(+1) + B(+0,5) + C(0) + D(-0,5) + E(-1)}{N}$$

Donde:

A representa el número de sujetos con índice individual 1

B representa el número de sujetos con índice individual 2

C representa el número de sujetos con índice individual 3 ó 6

## Capítulo 3

D representa el número de sujetos con índice individual 4

E representa el número de sujetos con índice individual 5

N representa el número total de sujetos del grupo

Como resultado se obtuvo:

$$ISG = \frac{6 * 1 + 1 * 0,5}{7}$$

De manera que el ISG = 0,93

Los resultados de la satisfacción individual según las categorías empleadas fueron los siguientes:

Nivel de satisfacción	Cantidad	%
Máxima satisfacción	6	93, %
Más satisfecho que insatisfecho	1	7%
No definida	0	0

Tabla 17. Resultado de aplicación de la técnica ladov. Fuente: Elaboración propia.

Al procesar las respuestas a las encuestas en el cuadro lógico de ladov, se obtiene un grado de satisfacción grupal de 0.93, lo cual se traduce en una clara satisfacción con el uso del Módulo Control Interno del SAEF3.

En el criterio respecto al control y disponibilidad de la información en el VDEA a través del uso de la solución propuesta, hubo una concordancia de un 100% en que contribuye a su mejora. De igual manera el 100% de los participantes manifestó que le gustaría mucho hacer uso del Módulo Cidel SAEF3 para desarrollar los procesos asociados al CI del VDEA. Mientras que un 7% no sabe si sería oportuno continuar ejecutando los procesos asociados al Cique se realiza a cada una de las áreas del VDEA.

Las preguntas abiertas que se formularon fueron:

- ¿Qué valoraciones le sugiere al Módulo CI del SAEF3 respecto al control y disponibilidad de la información asociada al VDEA?
- ¿Qué elemento(s) usted adicionaría a la solución que se propone?

Entre las valoraciones positivas obtenidas como respuestas a las preguntas abiertas, se recopilaron criterios como los siguientes:

- ✓ El módulo permite un control más eficiente de los procesos asociados al CI que se realiza a cada una de las áreasde la facultad.

## Capítulo 3

- ✓ El módulo permite agilizar la elaboración del Informe y el Plan de Medidas a partir de la aplicación de la Guía de Autocontrol.
- ✓ EL módulo garantiza que exista concordancia entre el Informe y el Plan de Medidas.
- ✓ El módulo le garantiza a los jefes de áreas rapidez para acceder a la información.
- ✓ EL módulo permite realizar un seguimiento constante a las medidas.
- ✓ Se recomienda que se informatice los procesos asociados a la ejecución y control del componente Gestión y Prevención de riesgos.
- ✓ El módulo debería exportar los Informes y Planes de Medidas a diferentes formatos.

La aplicación de la técnica de ladov aportó información significativa respecto al grado de satisfacción del cliente. Los resultados obtenidos y los criterios emitidos validan la fortaleza de la propuesta, reflejándose una valoración muy positiva del cliente con la solución.

Luego de aplicar estas pruebas quedó evidenciado el grado de aceptación del Módulo Control Interno evidenciándose esto con la carta de aceptación redactada por el cliente.

Luego de los resultados obtenidos en la validación de las variables y la aplicación de la técnica ladov, el cliente emite su carta de aceptación, lo que evidencia su satisfacción con el Módulo CI. Para ver detalles del acta de aceptación del Módulo CI consultar Anexo 7.

### **3.4 Conclusiones parciales**

- ✓ Se realizaron los diagramas de componentes y de despliegue para mostrar una vista de cómo quedó la aplicación a nivel de componentes y a nivel de hardware.
- ✓ Con las pruebas de caja blanca y caja negra se comprobó que todas las funcionalidades especificadas y aprobadas por el cliente fueron implementadas de forma correcta.
- ✓ Se diseñaron casos de pruebas para realizar pruebas funcionales a la aplicación los cuales arrojaron en todos los casos resultados satisfactorios.
- ✓ Se validaron las variables de la investigación demostrando el correcto funcionamiento de la aplicación y el grado de aceptación del cliente.
- ✓ Con las pruebas y validaciones realizadas la herramienta desarrollada cumple con las especificaciones y requisitos definidos por los clientes en la etapa de concepción del sistema.

## CONCLUSIONES

Con la realización del presente trabajo se desarrolló el Módulo CI para el Sistema de Administración y Economía de la Facultad 3, contribuyendo de esta manera a que la información que genera la aplicación de la Guía de Autocontrol en las diferentes áreas, no presente problemas de difícil acceso, deterioro y pérdidas de datos, logrando un mayor control y disponibilidad de la misma. Es por ello que al finalizar la presente investigación se puede afirmar que:

- ✓ El estudio de los conceptos fundamentales y los sistemas de CI existentes permitió determinar los elementos necesarios para, unidos con las exigencias del Vicedecanato de Economía y Administración mediante la entrevista realizada, conformar un conjunto de funcionalidades esenciales que debía cumplir la aplicación.
- ✓ El diseño de la solución permitió obtener un modelo en el que se visualizó la manera en que debía ser implementado el sistema y de esta forma facilitar el cumplimiento de las funcionalidades del mismo.
- ✓ A partir de esta implementación se obtuvo una aplicación funcional capaz de satisfacer las necesidades del cliente y resolver la gestión de la información del proceso de CI de la Facultad 3.
- ✓ Los resultados de las métricas TOC y RC demuestran que se validó el diseño realizado y la aplicación de pruebas de caja negra y caja blanca permitió validar las funcionalidades del módulo. Estas demuestran que la herramienta desarrollada cumple con las especificaciones y requisitos definidos en la etapa de concepción del sistema.
- ✓ Se validaron las variables definidas en la investigación mediante la comparación de indicadores, los cuales demostraron las ventajas del uso del módulo en comparación a la realización manual del proceso CI que se lleva a cabo en la Facultad 3. A partir de esta validación y los resultados de la técnica de ladov se obtuvo el acta de aceptación del Módulo CI por parte del cliente, dejando claro que el mismo cumple con todas las funcionalidades requeridas.

## **RECOMENDACIONES**

- ✓ Exportar los informes y planes de medidas en distintos formatos.
- ✓ Mantener un constante seguimiento a las leyes por las que se rige el proceso de control interno para en caso de posibles modificaciones no afecte el funcionamiento del módulo.
- ✓ Informatizar los procesos asociados a la ejecución y control del componente Gestión y Prevención de riesgos.

# Referencias

## REFERENCIAS

- Aguilera, Revuelta , Cardoso, Grimón y Deysi Lenadra. 2012.***Módulo Quejas, Peticiones o Denuncias del Sistema de Informatización de la Gestión Fiscal fase II.* La habana : s.n., 2012.
- Aguiluz, Javier. 2013.** Symfony. [En línea] 2013. [Citado el: 12 de 11 de 2014.] <http://symfony.es/que-es-symfony..>
- Álvarez, Miguel Angel. 2001.** DesarrolloWeb: Que es PHP. [En línea] 9 de 5 de 2001. [Citado el: 7 de 12 de 2014.] <http://www.desarrolloweb.com/articulos/392.php..>
- Belt Ibérica S.A. 2013.** Belt Ibérica, soluciones de seguridad global. *Belt Ibérica, soluciones de seguridad global.* [En línea] 2013. <http://www.belt.es>.
- Blanco. 1988.** Facultad de Ciencias Contables, Económicas y Administrativas. [En línea] 1988. <http://fccea.unicauca.edu.co/old/tgarf/tgarfse83.html>.
- Company, UBM plc. 2014.** PR Newswire:PhpStorm 2.0: Conquiste PHP. [En línea] 2014. [Citado el: 6 de 11 de 2014.]
- Comunidad Zonaeconomica. 2013.** Zona Económica. *Zona Económica.* [En línea] 2013. <http://www.zonaeconomica.com/control>.
- Contraloria General De La República. 2011.***RESOLUCION No. 60/11.* La Habana : s.n., 2011. pág. 39, Artículo.
- Cra. Daniela Biasco, Cra. Madelón Eiroa, Cra. Rosalía Nogueira. 2008.***ESTÁNDARES INTERNACIONALES DE CONTROL INTERNO APLICADOS A INSTITUCIONES DE INTERMEDIACIÓN FINANCIERA.* Uruguay : s.n., 2008.
- CTCP. 1999.** Facultad de Ciencias Contables, Económicas y Administrativas. [En línea] 1999. <http://fccea.unicauca.edu.co/old/tgarf/tgarfse83.html..>
- Desoft Guantánamo. 2011..** EcuRed. [En línea] 6 de 7 de 2011. <http://www.ecured.cu/index.php/Farola>.
- E. Wieggers, Karl . 2003.***Software Requirements, Second Edition.* 2003. ISBN:0735618798.
- Eguiluz, Javier. 2012.***Desarrollo Web Ágil de Symfony 2.* 2012.
- Expósito, Erly Delgado. 2009.***Sistema Informático de Apoyo a la Evaluación del Control Interno.* Matanzas : Red de Revistas Científicas de América Latina, el Caribe, España y Portugal, 2009. Vol. 3.
- Fuentes, Alcalde, Fernández, Bechara y Rodríguez, Torres. 2012.** Metodología para la implementación del sistema de control interno. [aut. libro] Bechara Fernández, Torres Rodríguez Alcalde Fuentes. s.l. : Observatorio de la Economía Latinoamericana, 2012.
- Ibañez, P. C. 2008.***El informe COSO y sus repercusiones: un enfoque moderno del control interno.* 2008.
- ICMP. 1982.** Facultad de Ciencias Contables, Económicas y Administrativas. [En línea] 1982. <http://fccea.unicauca.edu.co/old/tgarf/tgarfse83.html>.
- Larman, Craig. 1999.** 04\_Parte\_IV\_Fase\_del\_Disenio\_1. Mexico : s.n., 1999.
- . 2003.*UML y Patrones. 2da Edición.* 2003.
- . 2003.*UML y patrones.2da edicion.* 2003.
- López, José María. 2012.** Softonic. [En línea] 2012. <http://mindmanager.softonic.com>.
- M. Moreno, Ana , Juristo, Natalia y Vegas, Siras. 2006.***TÉCNICAS DE EVALUACIÓN DE SOFTWARE.* 2006.

## Referencias

- Martinez, Rafael. 2010.** PostgreSQL-es. [En línea] 2 de 10 de 2010. [Citado el: 4 de 12 de 2014.] [http://www.postgresql.org.es/sobre\\_postgresql](http://www.postgresql.org.es/sobre_postgresql).
- Microsoft. 2014.** Developer Network. [En línea] septiembre de 2014. <https://msdn.microsoft.com/es-es/library/dd409390.aspx>.
- . **2015.** Microsoft Developer Network. *Revisiones de código y estándares de codificación*. [En línea] 2015.
- N. V., Kuzmina. 1970.** *Metódicas Investigativas de la actividad pedagógica*. Leningrado : s.n., 1970.
- N.V, Kuzmina. 1970.** *Metódicas Investigativas de la actividad pedagógica*. Leningrado : s.n., 1970.
- Oracle. 2013.** Sitio oficial Netbeans. [En línea] 27 de 11 de 2013. <https://netbeans.org/features/platform/index.html>.
- Ould, Martyn. 2011.** INGTI. [En línea] 2011.
- Patricio Letelier, M<sup>a</sup> Carmen Penadés. 2006.** *Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP)*. Valencia : Universidad Politécnica de Valencia, 2006.
- Raghavan, Sridhar, Zelesnik, Gregory y Ford, Gary A. 1994.** *Lecture Notes on Requirements Elicitation*. Pittsburgh (E.E.U.U.) : Institute (Carnegie Mellon University), 1994.
- 2004.** Red Bean. [En línea] 2004. [Citado el: 4 de 12 de 2014.] <http://svnbook.red-bean.com/nightly/es/svn-ch-1-sect-1.htm>.
- Red Hat Enterprise. 2003.** Massachusetts Institute of Technology: Servidor Apache HTTP. [En línea] 2003. [Citado el: 14 de 11 de 2014.] <http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-rg-es-4/s1-apache-config.html>.
- Rodriguez, Tamara. 2014.** *Metodología de desarrollo para la Actividad productiva de la UCI*. La habana : s.n., 2014.
- Romero, Javier. 2012.** GestioPolis. [En línea] 31 de 8 de 2012.
- S. Pressman, Roger. 2000.** *Ingeniería del Software: Un enfoque práctico*. 2000.
- . **2010.** *Software Engineering. A Practitioners's Approach. Seventh*. Nueva York : McGraw-Hill, 2010. 978-0-07-337397-7.
- . **2005.** *Un enfoque práctico*. 2005.
- S. Roger, Pressman. 2005.** *Ingeniería del Software: Un enfoque práctico*. s.l. : Mc Graw Hill, 2005.
- Sanchez, Alvaro Fontela. 2013.** OpenWebCMS: Que es bootstrap. [En línea] 2013. [Citado el: 28 de 11 de 2014.]
- Sarmiento, Johana. 2013.** [En línea] 2013. <http://umldiagramadespliegue.blogspot.com/>.
- Sinfopac. 2014.** Sinfopac internacional. [En línea] 2014. [http://www.datevsinfopac.com/category.php?id\\_category=31](http://www.datevsinfopac.com/category.php?id_category=31).
- SoftExpert Software for Performance Excellence. 2011.** SoftExpert. [En línea] 2011. <http://www.softexpert.es/planificacion-control-auditorias.php>.
- Sommerville, Ian. 2007.** *Ingeniería de Software. 8va Edición*. Boston : Addison-Wesley, 2007. 9780321313799.
- 2010.** Técnicas de Pruebas. [En línea] 2010. [https://docs.google.com/viewer?a=v&q=cache:L0\\_moMWzYTYJ:indalog.ual.es/mtorres/LP/Prueba.pdf+pruebas+de+caja+negra&hl=es&gl=cu&pid=bl&srcid=ADGEEShyg3ux79zqPpID1O92U](https://docs.google.com/viewer?a=v&q=cache:L0_moMWzYTYJ:indalog.ual.es/mtorres/LP/Prueba.pdf+pruebas+de+caja+negra&hl=es&gl=cu&pid=bl&srcid=ADGEEShyg3ux79zqPpID1O92U)



## Referencias

U\_M3GdGAlm2GwqSjSr2t\_-ZPK92JW5\_hBCcumq1FcxUBeqrpgyuQ6QzRHEujVuzZ1T961tR-Omog9-fZtsh1JqlqFE.

**The jQuery Foundation. 2012.** Sitio Oficial: JQuery. [En línea] 2012. [Citado el: 10 de 12 de 2014.] <http://jquery.com/>.

**Trujillo, Cesar. 1993..LEY 87 DE 1993.** Santafé de Bogotá, D.C : s.n., 1993.

**2014.** Visual-paradigm. [En línea] 17 de 11 de 2014. <http://www.visual-paradigm.com/>.

**Wesley, Addison. 2004.** *Mike Cohn, "User Stories Applied"*. s.l. : ISBN 0-321-20568-5, 2004.

## **GLOSARIO**

**Guía de Autocontrol (GA):** documento que contiene los 5 componentes registrados en la Resolución 60 de la Contraloría General de la República, por los que se rigen las empresas cubanas para realizar el CI, con cada una de las normas correspondientes a dicho componente, así como sus ítems.

**Componente:** se aplica al elemento que forma parte de la GA.

**Norma:** es una regla que debe ser respetada y que permite ajustar ciertas conductas o actividades.

**Ítem:** escada una de las partes individuales que conforman la norma.

**Subítem:** escada una de las partes individuales que conforman el ítem.

**Adecuación:** es el proceso de adaptación de la GA a un área específica ante el cambio de ciertas condiciones preexistentes.

**Aplicación:** es la acción y el efecto de aplicar la GA a un área.

**Supervisión:** es la inspección al trabajo de las áreas realizado por la Vicedecana de Economía y Administración.

## ANEXOS

### Anexo 1. Descripción de los procesos de negocio.

Descripción del proceso de negocio Aplicar Guía de Autocontrol

<b>Objetivo</b>	Realizar la aplicación la Guía de Autocontrol a cada una de las áreas.
<b>Evento que lo genera</b>	Supervisión interna o externa.
<b>Pre condiciones</b>	Se cuenta con la Guía de Autocontrol de cada una de las áreas.
<b>Marco legal</b>	Resolución 60 de fecha 1ro de marzo del 2011.
<b>Reglas de negocio</b>	La Guía de Autocontrol será aplicada al menos dos veces en el año.
<b>Responsable</b>	Vicedecana Economía y Administración. Jefe de área.
<b>Clientes internos</b>	Vicedecana Economía y Administración. Jefes de áreas. Trabajadores de las áreas.
<b>Clientes externos</b>	Supervisores externos.
<b>Entradas</b>	Guía de Autocontrol.
<b>Flujo de eventos</b>	
<b>Flujo básico</b>	
	1. Seleccionar la Guía de Autocontrol (GA) según al área a la que se le aplicará. La Vicedecana Economía y Administración (VEA) selecciona la GA aplicable según el área en cuestión.
	2. Aplicar la GA al área. La VEA o el Jefe de área (Controlador) pasan por el área y realiza el control o la supervisión aplicando la GA. En caso de la aplicación tenga algún error ver flujo alterno <u>2.a Aplicación Incorrecta</u> .
	3. Elaborar Informe con la descripción de las deficiencias encontradas por cada componente de la GA. Luego de aplicar la GA se obtienen las deficiencias por cada componente y estas son explicadas detalladamente en un informe.
	4. Elaborar Plan de Medidas que responden a las deficiencias encontradas. Luego de identificar y explicar las deficiencias, se elabora el Plan de Medidas, donde se crean las medidas que darán solución a las deficiencias detectadas.
	5. Unificar documentación resultante luego de aplicar la GA. La VEA controla la documentación obtenida luego de aplicar la GA, el Informe y el Plan de Medidas.
	6. Entregar la documentación del resultado de la aplicación de la GA a los Jefes de áreas. La VEA le entrega el Informe y Plan de Medidas a cada Jefe de área.
	7. Recibir documentación. Cada Jefe de área recibe su Informe y Plan de Medidas como resultado de la aplicación de la GA a su área.
	8. Informar los resultados de la aplicación de la GA con los trabajadores de área. El Jefe de área se reúne con todos los trabajadores de su área para comunicar el resultado de la aplicación de la GA en su área.
<b>Pos-condiciones</b>	

1.	Se creó el Informe de deficiencias detectadas por cada componente.
1.	Se creó el Plan de Medidas
<b>Salidas</b>	
1.5.1	Informe de deficiencias
	Plan de Medidas
<b>Flujos paralelos</b>	
	N/A
<b>Pos-condiciones</b>	
	N/A
<b>Salidas</b>	
	N/A
<b>Flujos alternos</b>	
<b>2.a Aplicación Incorrecta</b>	
2.a	La VEA revisa la aplicación de la GA e informa que la misma tiene errores dejando de evaluarse un ítem.
	Vuelve al paso 2 del flujo básico
<b>Pos-condiciones</b>	
	N/A
<b>Salidas</b>	
	N/A
<b>Asuntos pendientes</b>	
	N/A

Tabla 18. Descripción del proceso de negocio Aplicar Guía de Autocontrol. Fuente: Elaboración propia

Diagrama del proceso de negocio Aplicar Guía de Autocontrol

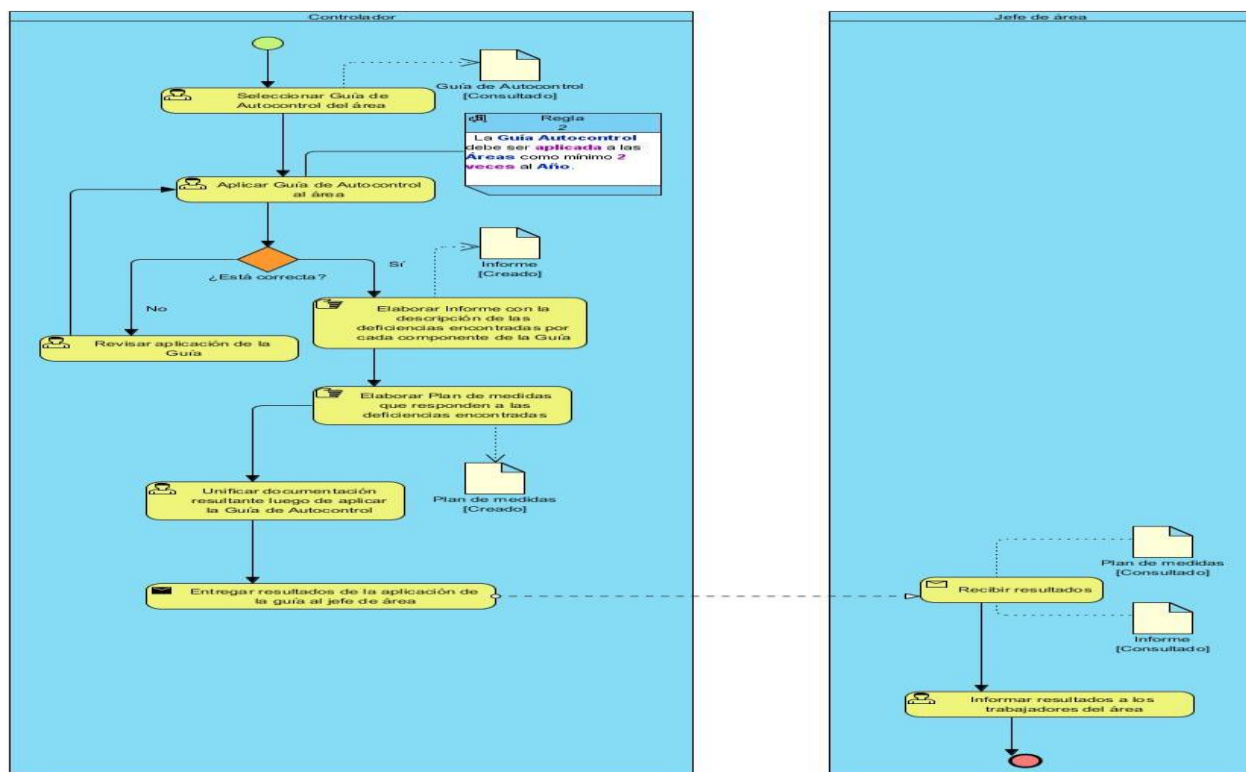


Figura 25. Diagrama de proceso de negocio Aplicar Guía de Autocontrol. Fuente: Elaboración propia

Descripción del proceso de negocio Realizar seguimiento del Plan de Medidas

<b>Objetivo</b>	Chequear que las medidas que se establecieron en el Plan de Medidas se hayan cumplido en la fecha establecida.
<b>Evento que lo genera</b>	Acercamiento de la fecha establecida para el cumplimiento de la medida. Una supervisión o control en el área.
<b>Pre condiciones</b>	Se crea el Plan de Medidas con fecha de cumplimiento de cada una de ellas. Cada medida debe tener un responsable y un ejecutante.
<b>Marco legal</b>	Resolución de 60 de fecha 1ero de marzo de 2011.
<b>Reglas de negocio</b>	N/A
<b>Responsable</b>	Programador.
<b>Clientes internos</b>	Vicedecana de Economía y Administración. Jefe de área.
<b>Clientes externos</b>	N/A
<b>Entradas</b>	Plan de Medidas
<b>Flujo de eventos</b>	
<b>Flujo básico</b>	

1. Recoger Plan de medidas del área. La Vicedecana de Administración y Economía (VEA) recoge el Plan de medidas correspondiente al área a controlar.

2. Chequear el cumplimiento de las medidas establecidas. La VEA chequea que las medidas plasmadas en el Plan de Medidas se cumplieron en dicha área.
3. Solicitar evidencias del cumplimiento de las medidas. La VEA solicita al Jefe de área las evidencias de que la o las medidas plasmadas en el Plan de medidas se cumplieron.
4. Entregar evidencias del cumplimiento de las medidas. El Jefe de área consulta el Plan de Medidas correspondiente al área y entrega las evidencias de las medidas cumplidas.
5. Revisar evidencias. La VEA revisa las evidencias para ver si las medidas se cumplieron en la fecha establecida y con la calidad requerida.
6. Enviar resultado de cumplimiento. La VEA manda los resultados del control del área, especificando las medidas que se cumplieron, las que se incumplieron y las que están en proceso de cumplimiento.
7. Recibir resultados. El Jefe de área recibe los resultados del control realizado a su área con las especificaciones correspondientes.
8. Enviar resultados al responsable. El Jefe de área envía los resultados a los responsables correspondientes de cada medida.

**Pos-condiciones**

1. Se controló el área.
2. El Plan de medidas y las evidencias asociadas a dichas medidas se revisaron.
3. Se detectaron las medidas incumplidas, cumplidas o en proceso de cumplimiento.

**Salidas**

**Flujos paralelos**

**Pos-condiciones**

N/A

**Salidas**

N/A

**Flujos alternos**

N/A

**Pos-condiciones**

N/A

**Salidas**

N/A

**Asuntos pendientes**

N/A

*Tabla 19. Descripción del proceso de negocio Realizar seguimiento del Plan de Medidas. Fuente: Elaboración propia.*

Diagrama del proceso de negocio Realizar seguimiento del Plan de Medidas

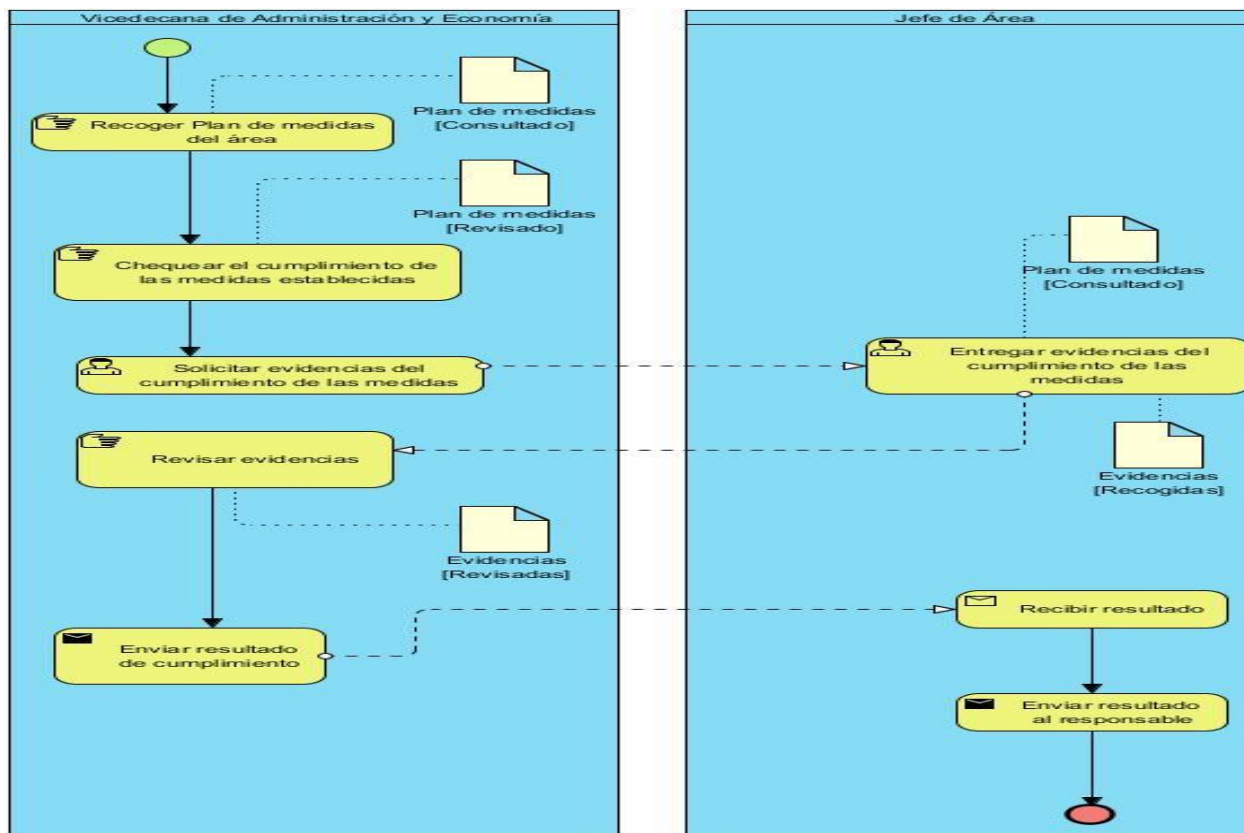


Figura 26. Diagrama del proceso de negocio Realizar seguimiento del Plan de Medidas. Fuente: Elaboración propia

## Anexo 2. Diccionario de datos

### Área

Descripción		Son las áreas que pertenecen a la Facultad 3 y a las que se les realiza el Control Interno.				
Atributos						
Nombre	Descripción	Tipo	¿Puede ser nulo?	¿Es único?	Restricciones	
nombre	Guarda el nombre del área	String	No	No	Clases válidas Caracteres alfabéticos a...z, A...Z Caracteres numéricos 0...9	Clases no válidas Caracteres Especiales, /[!~@#%*&*() _+.
descripcion	Guarda la descripción de lo que hace el área.	Text	No	No	Caracteres alfabéticos a...z, A...Z Caracteres numéricos 0...9	Caracteres Especiales, /[!~@#%*&*() _+.
id_area	Atributo identificador del área a la que se le realiza el CI	Integer	No	Si	Caracteres alfabéticos A...Z, a...z Caracteres Especiales, /[!~@#%*&*() _+. Números decimales 1.1, 24.45, etc Números fraccionarios 2/7, 23/50, etc	

Tabla 20. Concepto Área. Fuente: Elaboración propia.

## Usuario

Descripción		Se refiere al usuario que va a acceder al Módulo CI.				
Atributos						
Nombre	Descripción	Tipo	¿Puede ser nulo?	¿Es unico?	Restricciones	
					Clases válidas	Clases no válidas
user	Este atributo guarda el usuario de la persona que accederá al Módulo CI	String	No	Si	Caracteres alfabéticos A...Z, a...z	Caracteres numéricos 0...9 Caracteres Especiales
		String	No	Si	Caracteres alfabéticos A...Z, a...z Caracteres numéricos 0...9 Caracteres Especiales	N/A
contraseña	Este atributo guarda la contraseña del usuario que accederá al Módulo CI	Integer	No	Si	Caracteres numéricos 0...9 Caracteres Especiales	Caracteres alfabéticos A..Z, a...z Caracteres Especiales
id_usuario	Atributo identificador del usuario que se va a registrar en el Módulo CI	Integer	No	Si	Caracteres numéricos 0...9	Caracteres alfabéticos A..Z, a...z Caracteres Especiales

Tabla 21. Concepto Usuario. Fuente: Elaboración propia.

## Supervisión

Descripción		Se refiere a la supervisión que se le realiza a un área por parte de la Vicedecana de Economía y Administración o por una auditoría externa.				
Atributos						
Nombre	Descripción	Tipo	¿Puede ser nulo?	¿Es unico?	Restricciones	
					Clases válidas	Clases no válidas
id_supervision	Atributo identificador de la supervisión que se le realiza a un área determinada.	Integer	No	Si	Caracteres numéricos 0...9	Caracteres alfabéticos A..Z, a...z Caracteres Especiales Números decimales Números fraccionarios

Tabla 22. Concepto Supervisión. Fuente: Elaboración propia.

## Informe



Descripción		Es el documento donde se explican las deficiencias encontradas luego de haber aplicado la GA a un área.				
Atributos						
Nombre	Descripción	Tipo	¿Puede ser nulo?	¿Es único?	Restricciones	
fecha_ini	Este atributo guarda la fecha de inicio de la GA	Date	No	No	Clases válidas Caracteres numéricos 0...9 Caracter especial /	Clases no válidas Caracteres alfabéticos A...Z, a...z Caracteres Especiales_2[!~@#%&^&*()_+.
fecha_fin	Este atributo guarda la fecha de finalización de la GA	Date	No	No	Caracteres numéricos 0...9 Caracter especial /	Caracteres alfabéticos A...Z, a...z Caracteres Especiales_2[!~@#%&^&*()_+.
fecha_informe	Este atributo guarda la fecha de creación del informe	Date	No	No	Caracteres numéricos 0...9 Caracter especial /	Caracteres alfabéticos A...Z, a...z Caracteres Especiales_2[!~@#%&^&*()_+.
introduccion	Este atributo guarda los datos de la introducción del informe	Text	No	No	Caracteres alfabéticos A...Z, a...z	Caracteres Especiales_2[!~@#%&^&*()_+. Caracteres numéricos 0...9
resultados_componentes	Este atributo guarda los resultados por cada componente después de haber aplicado la GA	Text	No	No	Caracteres alfabéticos A...Z, a...z Caracteres numéricos	Caracteres Especiales_2[!~@#%&^&*()_+.
especificacion	Este atributo guarda la especificación de cada resultado por componente	Text	No	No	Caracteres alfabéticos A...Z, a...z Caracteres numéricos 0...9	Caracteres Especiales_2[!~@#%&^&*()_+.
informacion	Este atributo guarda la información adicional que se quiera documentar sobre la aplicación de la GA en esa área	Text	No	Si	Caracteres alfabéticos A...Z, a...z	Caracteres numéricos 0...9 Caracteres Especiales_2[!~@#%&^&*()_+.
subordinado_a	Este atributo guarda el responsable de revisar el informe una vez realizado	String	No	No	Caracteres alfabéticos A...Z, a...z	Caracteres numéricos 0...9 Caracteres Especiales_2[!~@#%&^&*()_+.
conclusiones	Este atributo guarda las conclusiones plasmadas en el informe	Text	No	No	Caracteres alfabéticos A...Z, a...z	Caracteres numéricos 0...9 Caracteres Especiales_2[!~@#%&^&*()_+.
anexo	Este atributo guarda cualquier archivo anexado al informe	Text	Si	No	Caracteres numéricos 0...9 Caracteres Especiales_2[!~@#%&^&*()_+. Caracteres alfabéticos A...Z, a...z	N/A
nombre_jefe_maximo	Este atributo guarda el nombre del responsable máximo que aprueba la creación del informe	String	No	Si	Caracteres alfabéticos A...Z, a...z	Caracteres numéricos 0...9 Caracteres Especiales_2[!~@#%&^&*()_+.

Tabla 23. Concepto Informe. Fuente: Elaboración propia.

Plan de medidas

Descripción		Es el documento que contiene la información del Plan de Medidas correspondiente a un área, creado después de haber aplicado la GA en dicha área.				
Atributos						
Nombre	Descripción	Tipo	¿Puede ser nulo?	¿Es único?	Restricciones	
					Clases válidas	Clases no válidas
fecha_ini	Este atributo guarda la fecha de inicio de la Guía de Autocontrol	Date	No	No	Caracteres numéricos 0...9 Caracter especial /	Caracteres alfabéticos A..Z, a..z Caracteres Especiales [!~@#%&*()_+.
fecha_fin	Este atributo guarda la fecha de fin de la GA	Date	No	No	Caracteres numéricos 0...9 Caracter especial /	Caracteres alfabéticos A..Z, a..z Caracteres Especiales [!~@#%&*()_+.
fecha_plan_medidas	Este atributo guarda la fecha de creación del Plan de Medidas	Date	No	No	Caracteres numéricos 0...9 Caracter especial /	Caracteres alfabéticos A..Z, a..z Caracteres Especiales [!~@#%&*()_+.
subordinado_a	Este atributo guarda el nombre del responsable de revisar el Plan de Medidas una vez creado	String	Si	No	Caracteres alfabéticos A..Z, a..z	Caracteres numéricos 0...9 Caracteres Especiales [!~@#%&*()_+.
numero_item_n o_consecutivo	Este atributo guarda el número del ítem (no consecutivo) correspondiente con el que tiene en la GA	Integer	No	Si	Caracteres numéricos 0...9	Caracteres alfabéticos A..Z, a..z Caracteres Especiales [!~@#%&*()_+.

nombre_deficiencia	Este atributo guarda el nombre de la deficiencia encontrada después de haber aplicado la GA a un área determinada	String	No	Si	Caracteres alfabéticos A...Z, a...z	Caracteres numéricos 0...9 Caracteres Especiales / ? [ ! ~ @ # \$ % ^ & * () _ + .
medida_deficiencia	Este atributo guarda la medida tomada a una deficiencia encontrada después de haber aplicado la GA	Text	No	No	Caracteres alfabéticos A...Z, a...z	Caracteres numéricos 0...9 Caracteres Especiales / ? [ ! ~ @ # \$ % ^ & * () _ + .
responsable_medida	Este atributo guarda el responsable de velar porque la medida se cumpla ante la fecha establecida	String	No	Si	Caracteres alfabéticos A...Z, a...z	Caracteres numéricos 0...9 Caracteres Especiales / ? [ ! ~ @ # \$ % ^ & * () _ + .
ejecutante_medida	Este atributo guarda el encargado de ejecutar la medida y hacer que se cumpla	String	No	Si	Caracteres alfabéticos A...Z, a...z	Caracteres numéricos 0...9 Caracteres Especiales / ? [ ! ~ @ # \$ % ^ & * () _ + .
fecha_cumplimiento_medida	Este atributo guarda la fecha de cumplimiento de la medida	Date	No	No	Caracteres numéricos 0...9 Caracter especial /	Caracteres alfabéticos A...Z, a...z Caracteres Especiales / ? [ ! ~ @ # \$ % ^ & * () _ + .
elementos_a_mejorar	Este atributo guarda el nombre de la tabla donde se guardarán los elementos a mejorar	Text	Si	No	Caracteres alfabéticos A...Z, a...z	Caracteres numéricos 0...9 Caracteres Especiales / ? [ ! ~ @ # \$ % ^ & * () _ + .
mejoras_a_realizar	Este atributo guarda las mejoras a realizar en algunos ítems que no están incumplidos pero que les faltan algunos detalles	Text	Si	No	Caracteres alfabéticos A...Z, a...z	Caracteres numéricos 0...9 Caracteres Especiales / ? [ ! ~ @ # \$ % ^ & * () _ + .

Tabla 24. Concepto Plan de Medidas. Fuente: Elaboración propia.

## Seguimiento

<b>Descripción</b>	Es el seguimiento que se le realiza al Plan de Medidas para que las medidas y las mejoras se cumplan en la fecha establecida.					
<b>Atributos</b>						
<b>Nombre</b>	<b>Descripción</b>	<b>Tipo</b>	<b>¿Puede ser nulo?</b>	<b>¿Es único?</b>	<b>Restricciones</b>	
					Clases válidas	Clases no válidas
id_seguimiento	Atributo identificador del seguimiento que se le realiza a un Plan de Medidas	Integer	No	Si	Caracteres numéricos 0...9	Caracteres alfabéticos A...Z, a...z. Caracteres Especiales, / ? [ ! ~ @ # \$ % ^ & * () _ + . Números decimales 1.1, 24.45, etc Números fraccionarios 2/7, 23/50, etc

Tabla 25. Concepto Seguimiento. Fuente: Elaboración propia.

## Mejoras

Descripción		Son las mejoras a realizar a los ítems no cumplidos de un área después de aplicar la GA.				
Atributos						
Nombre	Descripción	Tipo	¿Puede ser nulo?	¿Es único?	Restricciones	
nombre	Este atributo guarda el nombre de la mejora a realizar	String	No	SI	Clases válidas Caracteres alfabéticos A..Z, a..z	Clases no válidas Caracteres numéricos 0...9 Caracteres Especiales, /[ ] !~@#%*&'() _+-
		Text	No	SI	Caracteres alfabéticos A..Z, a..z	Caracteres numéricos 0...9 Caracteres alfabéticos A..Z, a..z Caracteres Especiales, /[ ] !~@#%*&'() _+-
descripcion	Este atributo guarda la descripción de la mejora a realizar por cada ítem	Integer	No	SI	Caracteres numéricos 0...9	Caracteres alfabéticos A..Z, a..z Caracteres Especiales, /[ ] !~@#%*&'() _+- Números decimales 1.1, 24.45, etc Números fraccionarios 2/7, 23/50, etc
		Text	No	SI	Caracteres alfabéticos A..Z, a..z	Caracteres numéricos 0...9 Caracteres alfabéticos A..Z, a..z Caracteres Especiales, /[ ] !~@#%*&'() _+-
id_mejora	Atributo Identificador de la mejora a realizar por cada ítem	Integer	No	SI	Caracteres numéricos 0...9	Caracteres alfabéticos A..Z, a..z Caracteres Especiales, /[ ] !~@#%*&'() _+- Números decimales 1.1, 24.45, etc Números fraccionarios 2/7, 23/50, etc

Tabla 26. Concepto Mejoras. Fuente: Elaboración propia.

## Medidas

Descripción		Son las medidas establecidas por cada deficiencia encontrada en un área después de haber aplicado la GA.				
Atributos						
Nombre	Descripción	Tipo	¿Puede ser nulo?	¿Es único?	Restricciones	
nombre	Este atributo guarda el nombre de la medida	String	No	SI	Clases válidas Caracteres alfabéticos A..Z, a..z	Clases no válidas Caracteres numéricos 0...9 Caracteres Especiales, /[ ] !~@#%*&'() _+-
		Text	No	SI	Caracteres alfabéticos A..Z, a..z	Caracteres numéricos 0...9 Caracteres Especiales, /[ ] !~@#%*&'() _+-
descripcion	Este atributo guarda la descripción de la medida	Text	SI	No	Caracteres alfabéticos A..Z, a..z	Caracteres numéricos 0...9 Caracteres Especiales, /[ ] !~@#%*&'() _+-
		Integer	No	SI	Caracteres numéricos 0...9	Caracteres alfabéticos A..Z, a..z Caracteres Especiales, /[ ] !~@#%*&'() _+- Números decimales 1.1, 24.45, etc Números fraccionarios 2/7, 23/50, etc
estado_cumplimiento	Este atributo guarda el estado de cumplimiento de la medida tomada: cumplida, incumplida y en cumplimiento	Integer	No	SI	Caracteres numéricos 0...9	Caracteres alfabéticos A..Z, a..z Caracteres Especiales, /[ ] !~@#%*&'() _+- Números decimales 1.1, 24.45, etc Números fraccionarios 2/7, 23/50, etc
id_medida	Atributo Identificador de la medida	Integer	No	SI	Caracteres numéricos 0...9	Caracteres alfabéticos A..Z, a..z Caracteres Especiales, /[ ] !~@#%*&'() _+- Números decimales 1.1, 24.45, etc Números fraccionarios 2/7, 23/50, etc

Tabla 27. Concepto Medidas. Fuente: Elaboración propia.

## Deficiencias

Descripción		Son las deficiencias encontradas por cada uno de los ítems no cumplidos después de haber aplicado la GA a un área.				
Atributos						
Nombre	Descripción	Tipo	¿Puede ser nulo?	¿Es único?	Restricciones	
nombre	Guarda el nombre de la deficiencia	String	No	No	Clases válidas Caracteres alfabéticos a...z, A...Z	Clases no válidas Caracteres Especiales, /3[!~@#%*&*0 _+. Caracteres numéricos o...9
		Text	No	No	Caracteres alfabéticos a...z, A...Z	Caracteres Especiales, /3[!~@#%*&*0 _+. Caracteres numéricos o...9
id_deficiencia	Atributo identificador de la deficiencia	Integer	No	Si	Caracteres numéricos o...9	Caracteres alfabéticos A...Z, a...z, Caracteres Especiales, /3[!~@#%*&*0 _+. Números decimales 1.1, 24.45, etc. Números fraccionarios 2/7, 23/50, etc.
		Integer	No	Si	Caracteres numéricos o...9	Caracteres alfabéticos A...Z, a...z, Caracteres Especiales, /3[!~@#%*&*0 _+. Números decimales 1.1, 24.45, etc. Números fraccionarios 2/7, 23/50, etc.

Tabla 28. Concepto Deficiencias. Fuente: Elaboración propia.

## Guía de Autocontrol

Descripción		La GA es un documento que se adecua según el área a la que se le vaya a realizar el CI. Contiene los 5 componentes registrados en la Resolución 60 de la Contraloría General de la República, con cada una de las normas correspondientes a dicho componente, además de los ítems a evaluar en dicho control.				
Atributos						
Nombre	Descripción	Tipo	¿Puede ser nulo?	¿Es único?	Restricciones	
nombre	Guarda el nombre de la guía	String	No	Si	Clases válidas Caracteres alfabéticos a...z, A...Z	Clases no válidas Caracteres Especiales, /3[!~@#%*&*0 _+. Caracteres numéricos o...9
		Integer	No	Si	Caracteres numéricos o...9	Caracteres alfabéticos A...Z, a...z, Caracteres Especiales, /3[!~@#%*&*0 _+. Números decimales 1.1, 24.45, etc. Números fraccionarios 2/7, 23/50, etc.
id_guia	Atributo identificador de la guía	Integer	No	Si	Caracteres numéricos o...9	Caracteres alfabéticos A...Z, a...z, Caracteres Especiales, /3[!~@#%*&*0 _+. Números decimales 1.1, 24.45, etc. Números fraccionarios 2/7, 23/50, etc.
		Integer	No	Si	Caracteres numéricos o...9	Caracteres alfabéticos A...Z, a...z, Caracteres Especiales, /3[!~@#%*&*0 _+. Números decimales 1.1, 24.45, etc. Números fraccionarios 2/7, 23/50, etc.

Tabla 29. Concepto Guía de Autocontrol. Fuente: Elaboración propia

## Componente

<b>Descripción</b>	Se refiere a los componentes que establece la GA definidos en la Resolución 60 de la Contraloría General de la República para realizar el Control Interno.					
<b>Atributos</b>						
<b>Nombre</b>	<b>Descripción</b>	<b>Tipo</b>	<b>¿Puede ser nulo?</b>	<b>¿Es único?</b>	<b>Restricciones</b>	<b>Clases no válidas</b>
nombre	Este atributo guarda el nombre del componente	String	No	Si	Clases válidas Caracteres alfabéticos a...z, A...Z	Caracteres Especiales, /[!~@#%*&*()_+. Caracteres numéricos 0...9
descripcion	Este atributo guarda la descripción de los componentes de la GA	Text	No	No	Caracteres alfabéticos a...z, A...Z	Caracteres Especiales, /[!~@#%*&*()_+. Caracteres numéricos 0...9
id_componente	Atributo identificador del componente	Integer	No	Si	Caracteres numéricos 0...9	Caracteres alfabéticos A..Z, a..z. Caracteres Especiales, /[!~@#%*&*()_+. Números decimales 1.1, 24.45, etc Números fraccionarios 2/7, 23/50, etc

Tabla 30. Concepto Componente. Fuente: Elaboración propia.

## Norma

<b>Descripción</b>	Son las normas correspondientes a cada componente de la GA definidas en la Resolución 60 de la Contraloría General de la República.					
<b>Atributos</b>						
<b>Nombre</b>	<b>Descripción</b>	<b>Tipo</b>	<b>¿Puede ser nulo?</b>	<b>¿Es único?</b>	<b>Restricciones</b>	<b>Clases no válidas</b>
nombre	Este atributo guarda el nombre de la norma definido en la Resolución 60	String	No	Si	Caracteres alfabéticos A...Z, a...z	Caracteres numéricos 0...9 Caracteres Especiales, /[!~@#%*&*()_+.
descripcion	Este atributo guarda la descripción de la norma de la GA	Text	No	Si	Caracteres alfabéticos A...Z, a...z	Caracteres numéricos 0...9 Caracteres Especiales, /[!~@#%*&*()_+.
id_norma	Atributo identificador de la norma	Integer	No	Si	Caracteres numéricos 0...9	Caracteres alfabéticos A..Z, a..z. Caracteres Especiales, /[!~@#%*&*()_+. Números decimales 1.1, 24.45, etc Números fraccionarios 2/7, 23/50, etc

Tabla 31. Concepto Norma. Fuente: Elaboración propia.

## Ítem



<b>Descripción</b>	La adecuación se le realiza a la GA a un área determinada. Contiene los componentes de la Resolución 60 de la Contraloría General de la República, con cada una de sus normas y los ítems que se evalúan en esa área.					
<b>Atributos</b>						
<b>Nombre</b>	<b>Descripción</b>	<b>Tipo</b>	<b>¿Puede ser nulo?</b>	<b>¿Es único?</b>	<b>Restricciones</b>	
id_adecuacion	Atributo identificador de la adecuación que se está realizando a un área determinada.	Integer	No	Si	<b>Clases válidas</b> Caracteres numéricos 0...9	<b>Clases no válidas</b> Caracteres alfabéticos, A..Z, a..z. Caracteres Especiales, /[]!;~@#%&*( ) _+. Números decimales 1.1, 24.45, etc. Números fraccionarios 2/7, 23/50, etc

Tabla 34. Concepto Adecuación. Fuente: Elaboración propia.

### Anexo 3. Descripción de requisitos por Historias de usuario.

#### HU Gestionar Adecuación de la Guía de Autocontrol

##### HU-02 Modificar Adecuación de la Guía de Autocontrol

<b>Número:</b> HU-02		<b>Nombre del requisito:</b> Modificar Adecuación de la Guía de Autocontrol	
<b>Programador:</b> Yanirosa Almenares		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 5 días	
<b>Riesgo en Desarrollo:</b> Alta		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El usuario podrá seleccionar dentro de los componentes y sus normas, los ítems adecuados al área, podrá seleccionar y deseleccionar tantas veces que sea necesario los ítems, modificando la Adecuación hecha previamente.			



**Prototipo de interfaz:**

Tabla 35. Historia de Usuario Modificar Adecuación de la Guía de Autocontrol. Fuente: Elaboración propia

**HU-03 Eliminar Adecuación de la Guía de Autocontrol**

<b>Número:</b> HU-03	<b>Nombre del requisito:</b> Eliminar Adecuación de la Guía de Autocontrol
<b>Programador:</b> Yanirosa Almenares	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 3 días
<b>Riesgo en Desarrollo:</b> Alta	<b>Tiempo Real:</b> 1 semana
<b>Descripción:</b> El usuario podrá eliminar la Adecuación de la Guía de Autocontrol realizada a un área determinada.	

**Prototipo de interfaz:**

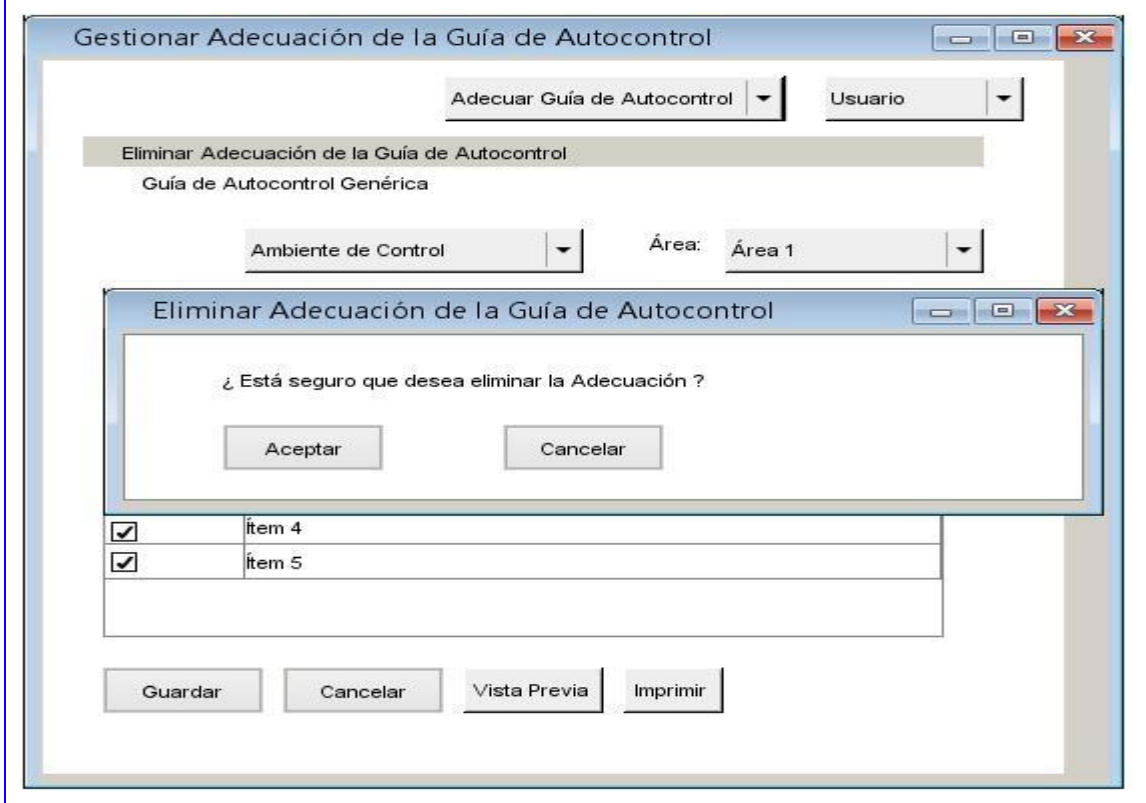


Tabla 36. Historia de Usuario Eliminar Adecuación de la Guía de Autocontrol. Fuente: Elaboración propia

**HU Gestionar Guía de Autocontrol**

**HU-01 Adicionar Guía de Autocontrol**

<b>Número:</b> HU-01	<b>Nombre del requisito:</b> Adicionar Guía de Autocontrol
<b>Programador:</b> Yanirosa Almenares	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 3 días
<b>Riesgo en Desarrollo:</b> Alta	<b>Tiempo Real:</b> 1 semana
<b>Descripción:</b> El usuario podrá adicionar una Guía de Autocontrol.	

**Prototipo de interfaz:**

Tabla 37. Historia de Usuario Adicionar Guía de Autocontrol. Fuente: Elaboración propia

**HU-02 Modificar Guía de Autocontrol**

<b>Número:</b> HU-02	<b>Nombre del requisito:</b> Modificar Guía de Autocontrol
<b>Programador:</b> Yanirosa Almenares	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 3 días
<b>Riesgo en Desarrollo:</b> Alta	<b>Tiempo Real:</b> 1 semana
<b>Descripción:</b> El usuario podrá modificar la Guía de Autocontrol.	

**Prototipo de interfaz:**

Tabla 38. Historia de Usuario Modificar Guía de Autocontrol. Fuente: Elaboración propia

**HU-03 Eliminar Guía de Autocontrol**

<b>Número:</b> HU-03		<b>Nombre del requisito:</b> Eliminar Guía de Autocontrol	
<b>Programador:</b> Yanirosa Almenares		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 3 días	
<b>Riesgo en Desarrollo:</b> Alta		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El usuario podrá eliminar la Guía de Autocontrol.			

**Prototipo de interfaz:**



Tabla 39. Historia de Usuario Eliminar Guía de Autocontrol. Fuente: Elaboración propia

**HU Gestionar Componente**

**HU-01 Adicionar componente**

<b>Número:</b> HU-01		<b>Nombre del requisito:</b> Adicionar componente	
<b>Programador:</b> Yanirosa Almenares		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 3 días	
<b>Riesgo en Desarrollo:</b> Alta		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El usuario podrá adicionar los componentes al sistema.			

**Prototipo de interfaz:**

Tabla 40. Historia de Usuario Adicionar Componente. Fuente: Elaboración propia

**HU-02 Modificar componente**

<b>Número:</b> HU-02		<b>Nombre del requisito:</b> Modificar componente	
<b>Programador:</b> Yanirosa Almenares		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 3 días	
<b>Riesgo en Desarrollo:</b> Alta		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El sistema muestra el listado de los componentes que existen. El usuario selecciona el componente que podrá modificar los datos.			

**Prototipo de interfaz:**

Tabla 41. Historia de Usuario Modificar Componente. Fuente: Elaboración propia

**HU-03 Eliminar componente**

<b>Número:</b> HU-03	<b>Nombre del requisito:</b> Eliminar componente
<b>Programador:</b> Yanirosa Almenares	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 3 días
<b>Riesgo en Desarrollo:</b> Alta	<b>Tiempo Real:</b> 1 semana
<b>Descripción:</b> El sistema muestra el listado de los componentes que existen y el usuario selecciona el/los componente(s) que podrá eliminar.	

**Prototipo de interfaz:**

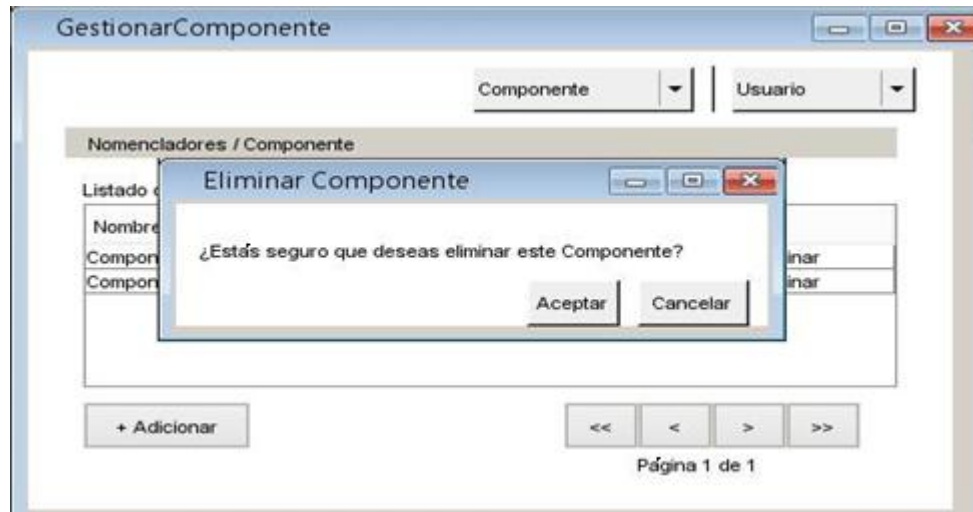


Tabla 42. Historia de Usuario Eliminar Componente. Fuente: Elaboración propia

**HU-04 Listar componente**

<b>Número:</b> HU-04	<b>Nombre del requisito:</b> Listar componente
<b>Programador:</b> Yanirosa Almenares	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 3 días
<b>Riesgo en Desarrollo:</b> Alta	<b>Tiempo Real:</b> 1 semana
<b>Descripción:</b> El sistema muestra el listado de los componentes que existen y el usuario selecciona el/los componente(s).	



**Prototipo de interfaz:**



Tabla 43. Historia de Usuario Listar Componente. Fuente: Elaboración propia

**HU Gestionar Norma**

**HU-01 Adicionar norma**

<b>Número:</b> HU-01		<b>Nombre del requisito:</b> Adicionar norma	
<b>Programador:</b> Yanirosa Almenares		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 3 días	
<b>Riesgo en Desarrollo:</b> Alta		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El sistema muestra el listado de los componentes que existen. El usuario selecciona previamente un componente y podrá adicionar las normas correspondientes a dicho componente.			

**Prototipo de interfaz:**

Tabla 44. Historia de Usuario Adicionar Norma. Fuente: Elaboración propia

**HU-02 Modificar norma**

<b>Número:</b> HU-02		<b>Nombre del requisito:</b> Modificar norma	
<b>Programador:</b> Yanirosa Almenares		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 3 días	
<b>Riesgo en Desarrollo:</b> Alta		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El sistema muestra el listado de los componentes que existen. El usuario busca el componente, dentro del cual están sus normas correspondientes y selecciona la norma a la cual podrá modificar sus datos.			

**Prototipo de interfaz:**

Tabla 45. Historia de Usuario Modificar Norma. Fuente: Elaboración propia

**HU-03 Eliminar norma**

<b>Número:</b> HU-03		<b>Nombre del requisito:</b> Eliminar norma	
<b>Programador:</b> Yanirosa Almenares		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 3 días	
<b>Riesgo en Desarrollo:</b> Alta		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El sistema muestra el listado de los componentes que existen. El usuario busca el componente, dentro del cual están sus normas correspondientes selecciona la/s norma(s) que podrá eliminar.			

**Prototipo de interfaz:**



Tabla 46. Historia de Usuario Eliminar Norma. Fuente: Elaboración propia

**HU-04 Listar norma**

<b>Número:</b> HU-04	<b>Nombre del requisito:</b> Listar norma
<b>Programador:</b> Yanirosa Almenares	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 3 días
<b>Riesgo en Desarrollo:</b> Alta	<b>Tiempo Real:</b> 1 semana
<b>Descripción:</b> El sistema muestra el listado de las normas que existen y el usuario selecciona la/s norma(s).	

Prototipo de interfaz:

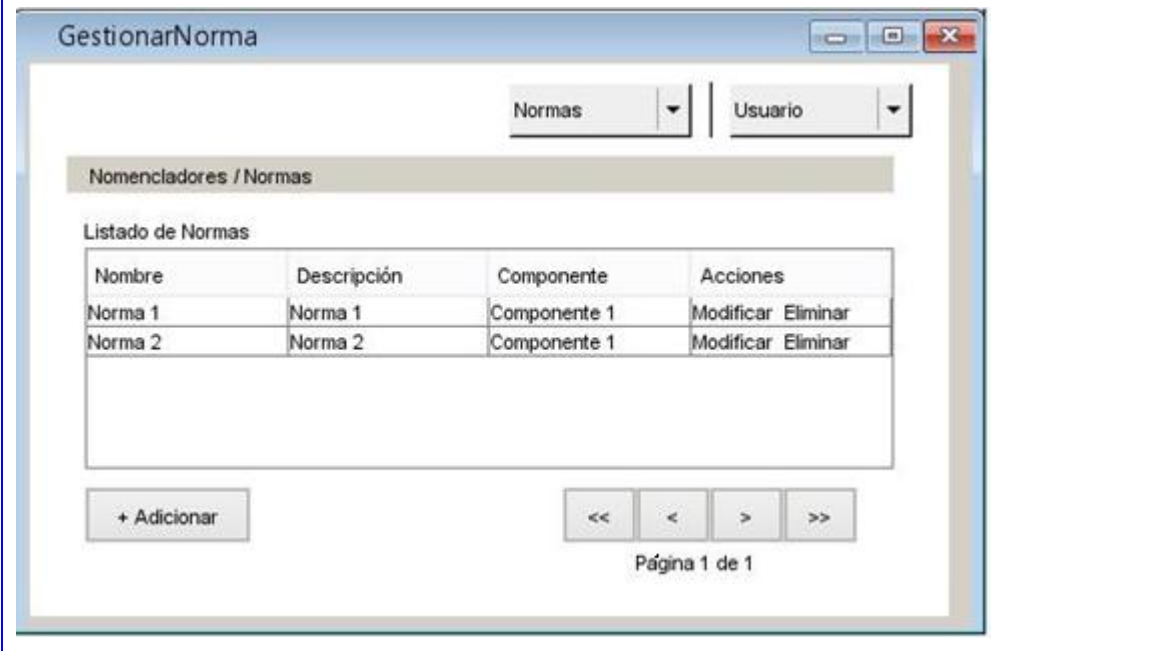


Tabla 47. Historia de Usuario Listar Norma. Fuente: Elaboración propia

HU Gestionar Ítem

HU-01 Adicionar ítem

<b>Número:</b> HU-01		<b>Nombre del requisito:</b> Adicionar ítem	
<b>Programador:</b> Yanirosa Almenares		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 3 días	
<b>Riesgo en Desarrollo:</b> Alta		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El usuario selecciona previamente un componente y luego selecciona una de sus normas donde podrá adicionar los ítems correspondientes a dicha norma.			

**Prototipo de interfaz:**

Tabla 48. Historia de Usuario Adicionar Ítem. Fuente: Elaboración propia

**HU-02 Modificar ítem**

<b>Número:</b> HU-01	<b>Nombre del requisito:</b> Adicionar ítem
<b>Programador:</b> Yanirosa Almenares	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 3 días
<b>Riesgo en Desarrollo:</b> Alta	<b>Tiempo Real:</b> 1 semana
<b>Descripción:</b> El usuario selecciona previamente un componente y luego selecciona una de sus normas donde podrá adicionar los ítems correspondientes a dicha norma.	

**Prototipo de interfaz:**

Tabla 49. Historia de Usuario Modificar Ítem. Fuente: Elaboración propia

**HU-03 Eliminar ítem**

<b>Número:</b> HU-03		<b>Nombre del requisito:</b> Eliminar ítem	
<b>Programador:</b> Yanirosa Almenares		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 3 días	
<b>Riesgo en Desarrollo:</b> Alta		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El sistema muestra el listado de los componentes que existen. El usuario busca el componente, posteriormente busca la norma, dentro de la cual están sus ítems correspondientes y selecciona el/los ítem(s) que podrá eliminar.			

**Prototipo de interfaz:**

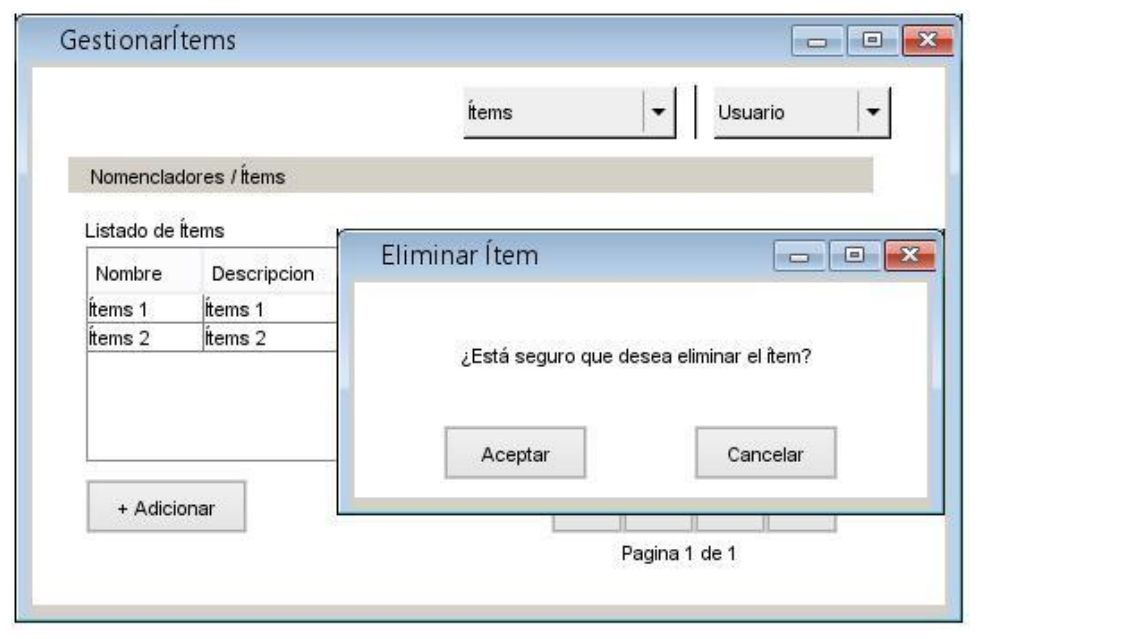


Tabla 50. Historia de Usuario Eliminar Ítem. Fuente: Elaboración propia

**HU-04 Listar ítem**

<b>Número:</b> HU-04		<b>Nombre del requisito:</b> Listar ítems	
<b>Programador:</b> Yanirosa Almenares		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 3 días	
<b>Riesgo en Desarrollo:</b> Alta		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El sistema muestra el listado de los ítems que existen y el usuario selecciona el/los ítem(s).			



**Prototipo de interfaz:**

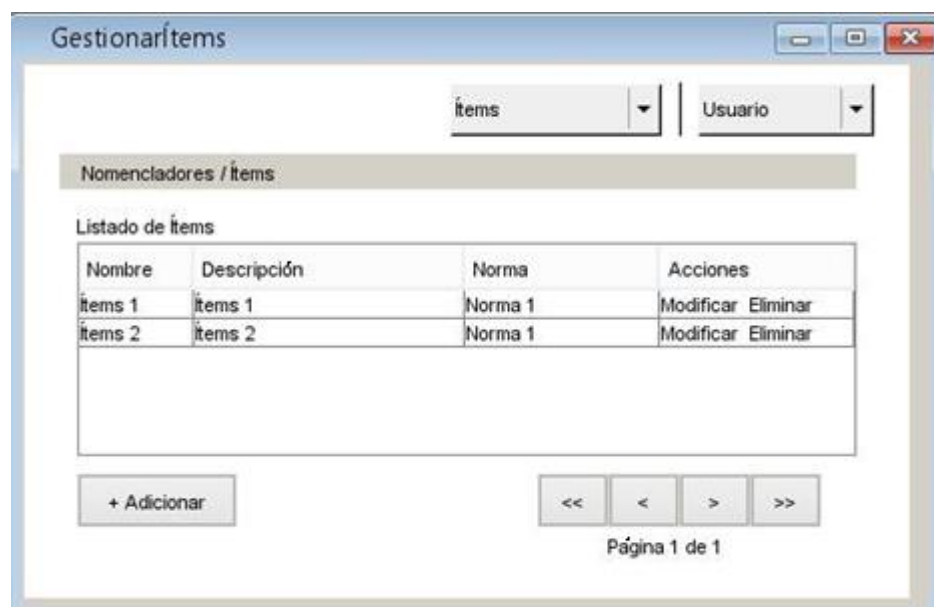


Tabla 51. Historia de Usuario Listar Ítem. Fuente: Elaboración propia

**HU- Aplicar GA**

<b>Número:</b> HU-01		<b>Nombre del requisito:</b> Aplicar GA	
<b>Programador:</b> Yeisel Pérez Rivera		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 5 días	
<b>Riesgo en Desarrollo:</b> N/A		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> Permite al usuario aplicar la Guía de Autocontrol después de haber sido adecuada al área.			
<b>Observaciones:</b> N/A			
<b>Prototipo de interfaz:</b>			

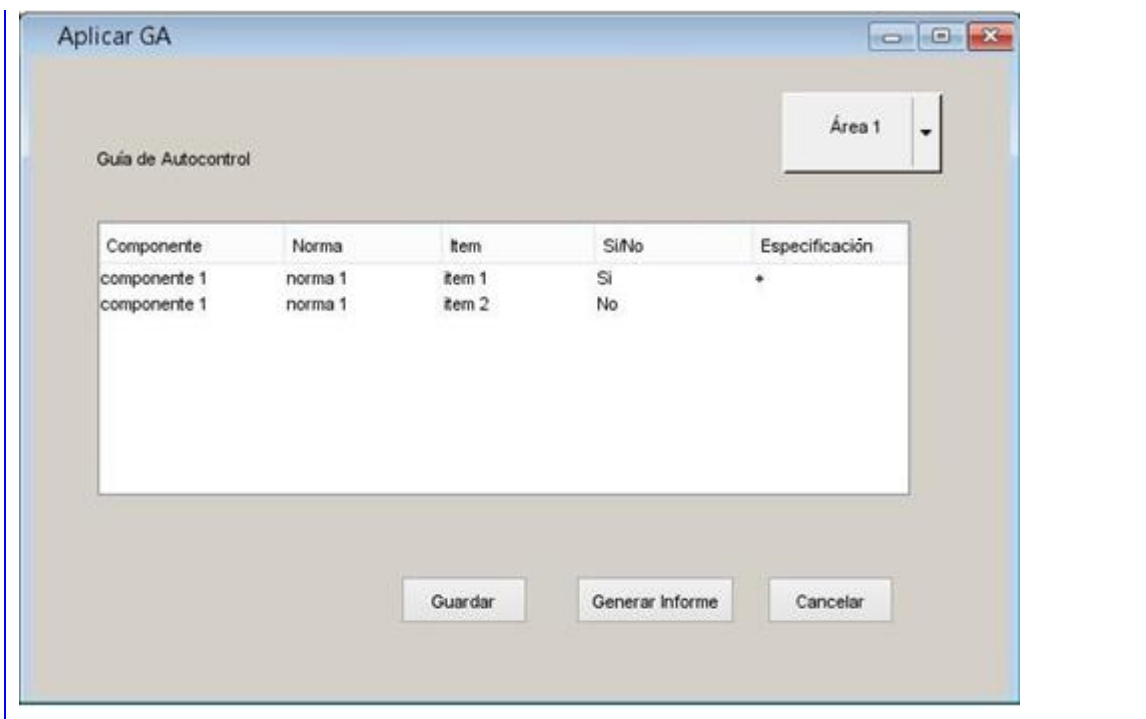


Tabla 52. Historia de Usuario Aplicar GA. Fuente: Elaboración propia

### HU- Generar Informe de aplicación de la GA

<b>Número:</b> HU-01		<b>Nombre del requisito:</b> Generar Informe de aplicación de la GA	
<b>Programador:</b> Yeisel Pérez Rivera		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 8 días	
<b>Riesgo en Desarrollo:</b> N/A		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El usuario podrá generar el informe con la explicación detallada de cada una de las deficiencias encontradas en el área, una vez realizada la aplicación de la GA.			
<b>Observaciones:</b> N/A			
<b>Prototipo de interfaz:</b>			

Generar Informe

Fecha inicio\_GA: dd/mm/aa      Fecha fin\_GA: dd/mm/aa      Fecha Informe: dd/mm/aa

Subordinado a: \_\_\_\_\_

Introducción

Resultados

Componente	Norma	Cantidad de ítems incumplidos	Si/No	Deficiencias	Especificación	Causas	Condiciones
componente 1	norma 1	ítem 1	Si		+		
componente 2	norma 1	ítem 2	No				

Información

Conclusiones

Anexo: \_\_\_\_\_ Examinar...

Nombre del Jefe máximo: \_\_\_\_\_ Cargo: \_\_\_\_\_ Firma: \_\_\_\_\_

Generar Informe      Cancelar

Tabla 53. Historia de Usuario Generar Informe de la aplicación de la GA. Fuente: Elaboración propia

### HU- Generar Plan de Medidas

<b>Número:</b> HU-01	<b>Nombre del requisito:</b> Generar Plan de Medidas		
<b>Programador:</b> Yeisel Pérez Rivera	<b>Iteración Asignada:</b> 1		
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 8 días		
<b>Riesgo en Desarrollo:</b> N/A	<b>Tiempo Real:</b> 1 semana		
<b>Descripción:</b> El usuario podrá generar el Plan de Medidas que tendrá cada las			

medidas correspondientes a cada deficiencia encontrada en el área, una vez realizada la aplicación de la GA.

**Observaciones:** N/A

**Prototipo de interfaz:**

Tabla 54. Historia de Usuario Generar Plan de Medidas. Fuente: Elaboración propia

### HU- Generar Seguimiento del Plan de Medidas

<b>Número:</b> HU-04	<b>Nombre del requisito:</b> Generar Seguimiento del Plan de Medidas
<b>Programador:</b> Yeisel Pérez Rivera	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 5 días
<b>Riesgo en Desarrollo:</b> N/A	<b>Tiempo Real:</b> 1 semana
<b>Descripción:</b> El usuario podrá realizar un seguimiento del Plan de Medidas para comprobar si se le están dando cumplimiento a las deficiencias encontradas en el	

área.

**Observaciones:** N/A

**Prototipo de interfaz:**

Medidas	Deficiencia	Ítem	% Cumplimiento	Fecha	Estado de cumplimiento
medida 1	deficiencia 1	# ítem	%	dd/mm/aa	cumplido
medida 2	deficiencia 1	# ítem	%	dd/mm/aa	incumplido
medida 3	deficiencia 2	# ítem	%	dd/mm/aa	en proceso

Tabla 55. Historia de Usuario Generar Seguimiento del Plan de Medidas. Fuente: Elaboración propia

## HU- Gestionar Área

### HU-01 Adicionar área

<b>Número:</b> HU-01		<b>Nombre del requisito:</b> Adicionar área	
<b>Programador:</b> Yeisel Pérez Rivera		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Media		<b>Tiempo Estimado:</b> 1 días	
<b>Riesgo en Desarrollo:</b> N/A		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El usuario podrá adicionar un área al sistema.			
<b>Observaciones:</b> N/A			
<b>Prototipo de interfaz:</b>			

Tabla 56. Historia de Usuario Adicionar área. Fuente: Elaboración propia

### HU-02 Modificar área

<b>Número:</b> HU-02		<b>Nombre del requisito:</b> Modificar área	
<b>Programador:</b> Yeisel Pérez Rivera		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Media		<b>Tiempo Estimado:</b> 1 día	
<b>Riesgo en Desarrollo:</b> N/A		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El usuario podrá realizarle modificaciones a un área determinada.			
<b>Observaciones:</b> N/A			
<b>Prototipo de interfaz:</b>			

The image shows a software dialog box titled "Modificar Área". It has a standard Windows-style title bar with minimize, maximize, and close buttons. The main area contains three text input fields: "Nombre del Área" (top left), "Jefe de Área" (middle left), and "Descripción" (right side). At the bottom right, there are two buttons: "Modificar" and "Cancelar".

Tabla 57. Historia de Usuario Modificar área. Fuente: Elaboración propia

### HU-03 Eliminar área

<b>Número:</b> HU-03		<b>Nombre del requisito:</b> Eliminar área	
<b>Programador:</b> Yeisel Pérez Rivera		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Media		<b>Tiempo Estimado:</b> 1 día	
<b>Riesgo en Desarrollo:</b> N/A		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> Una vez reflejada el área en el sistema, la misma se podrá eliminar si se desea.			
<b>Observaciones:</b> N/A			
<b>Prototipo de interfaz:</b>			

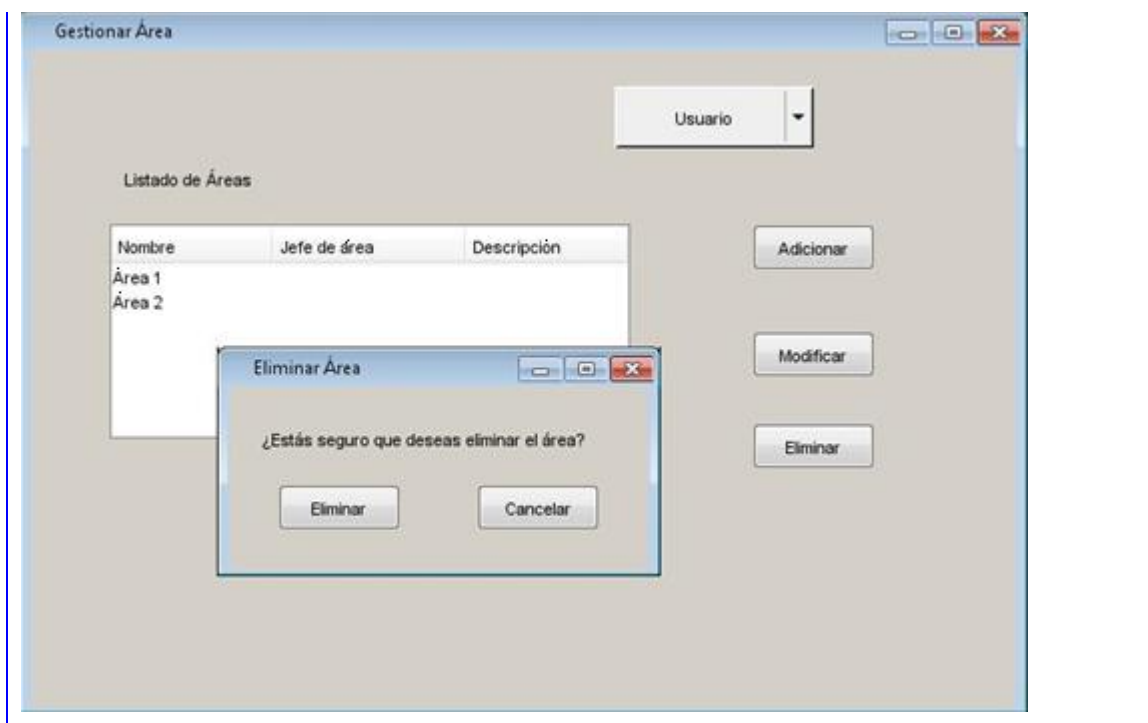


Tabla 58. Historia de Usuario Eliminar área. Fuente: Elaboración propia

## HU- Gestionar Deficiencia

### HU-01 Adicionar deficiencia

<b>Número:</b> HU-01		<b>Nombre del requisito:</b> Adicionar deficiencia	
<b>Programador:</b> Yeisel Pérez Rivera		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 6 día	
<b>Riesgo en Desarrollo:</b> N/A		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El usuario podrá adicionar la deficiencia encontrada al Plan de Medidas.			
<b>Observaciones:</b> N/A			
<b>Prototipo de interfaz:</b>			



Tabla 59. Historia de Usuario Adicionar deficiencia. Fuente: Elaboración propia

### HU-02 Modificar deficiencia

<b>Número:</b> HU-01		<b>Nombre del requisito:</b> Modificar deficiencia	
<b>Programador:</b> Yeisel Pérez Rivera		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 6 día	
<b>Riesgo en Desarrollo:</b> N/A		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El usuario podrá modificar la deficiencia encontrada al Plan de Medidas.			
<b>Observaciones:</b> N/A			

**Prototipo de interfaz:**

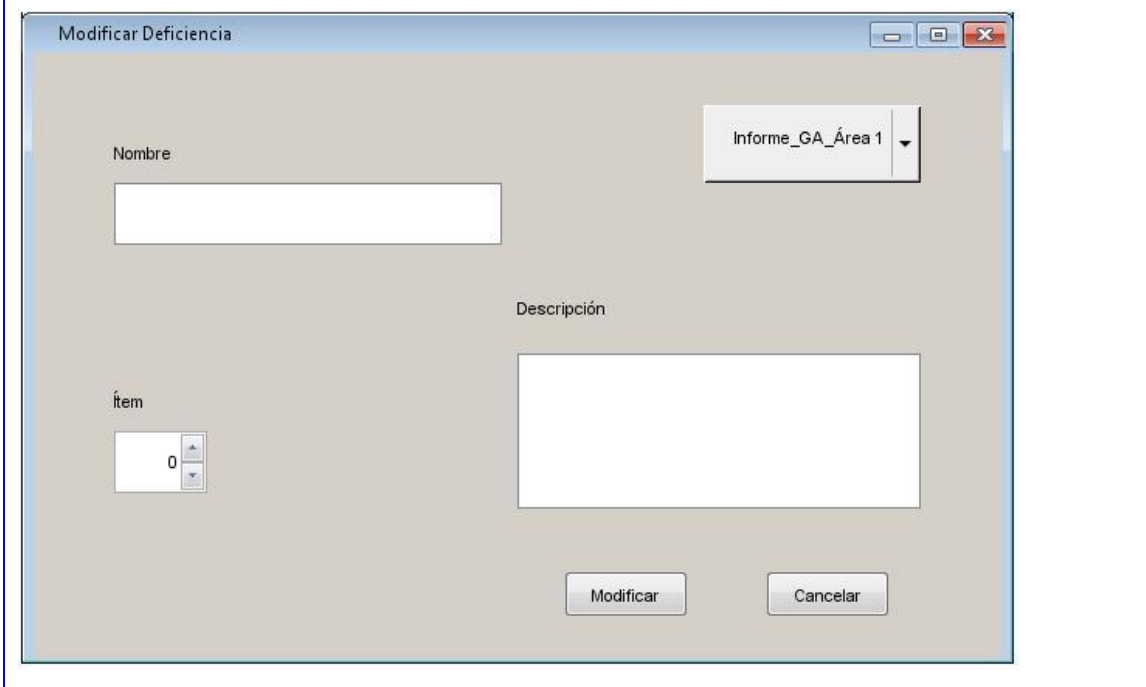


Tabla 60. Historia de Usuario Modificar deficiencia. Fuente: Elaboración propia

**HU-03 Eliminar deficiencia**

<b>HU-03 Eliminar deficiencia</b>	
<b>Número:</b> HU-03	<b>Nombre del requisito:</b> Eliminar deficiencia
<b>Programador:</b> Yeisel Pérez Rivera	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 6 día
<b>Riesgo en Desarrollo:</b> N/A	<b>Tiempo Real:</b> 1 semana
<b>Descripción:</b> El usuario podrá eliminar la deficiencia encontrada en el Plan de Medidas.	
<b>Observaciones:</b> N/A	
<b>Prototipo de interfaz:</b>	

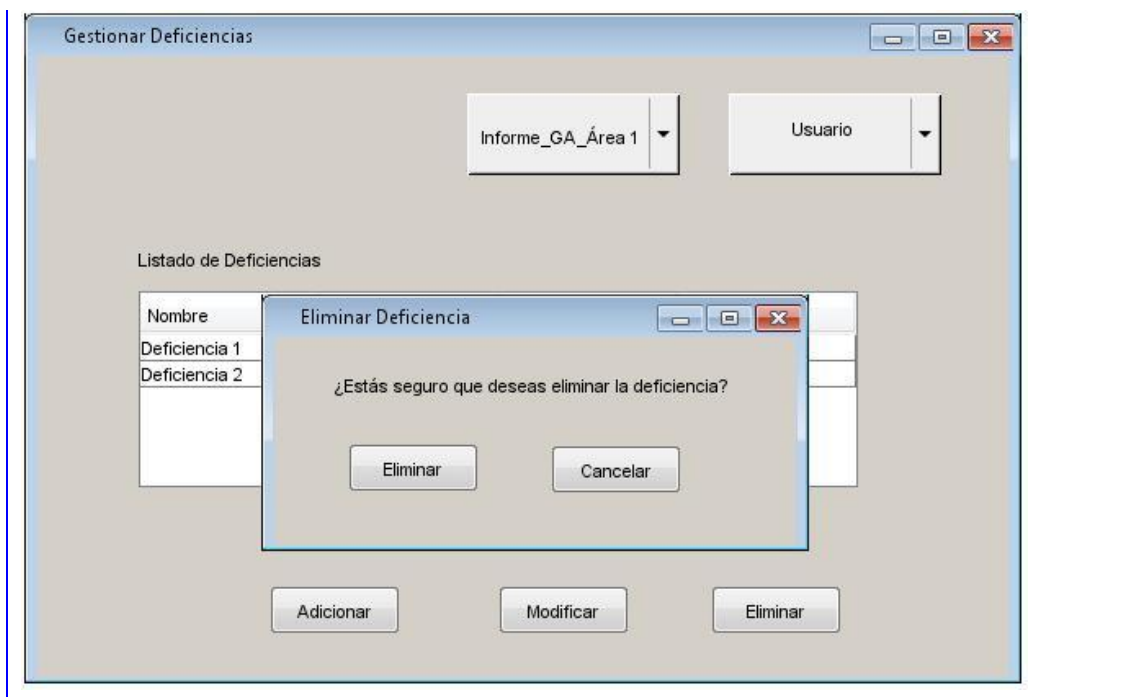


Tabla 61. Historia de Usuario Eliminar deficiencia. Fuente: Elaboración propia

## HU- Gestionar Medida

### HU-01 Adicionar medida

<b>Número:</b> HU-01		<b>Nombre del requisito:</b> Adicionar medida	
<b>Programador:</b> Yeisel Pérez Rivera		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 4 días	
<b>Riesgo en Desarrollo:</b> N/A		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El usuario podrá adicionar la medida al Plan de medidas.			
<b>Observaciones:</b> N/A			
<b>Prototipo de interfaz:</b>			

Tabla 62. Historia de Usuario Adicionar medida. Fuente: Elaboración propia

### HU-02 Modificar medida

<b>Número:</b> HU-02		<b>Nombre del requisito:</b> Modificar medida	
<b>Programador:</b> Yeisel Pérez Rivera		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 4 días	
<b>Riesgo en Desarrollo:</b> N/A		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El usuario podrá modificar las medidas reflejadas en el Plan de medidas.			
<b>Observaciones:</b> N/A			
<b>Prototipo de interfaz:</b>			

The image shows a software dialog box titled "Modificar Medida". It features a "Ítem" spinner control with the value "0", a "Deficiencia" dropdown menu, a "Nombre" text input field, and a "Descripción" text area. At the bottom right, there are two buttons: "Modificar" and "Cancelar".

Tabla 63. Historia de Usuario Modificar medida. Fuente: Elaboración propia

### HU-03 Eliminar medida

<b>Número:</b> HU-03		<b>Nombre del requisito:</b> Eliminar Medida	
<b>Programador:</b> Yeisel Pérez Rivera		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta		<b>Tiempo Estimado:</b> 4 días	
<b>Riesgo en Desarrollo:</b> N/A		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> Una vez reflejada una medida en el Plan de medidas la misma se podrá eliminar si se desea.			
<b>Observaciones:</b> N/A			
<b>Prototipo de interfaz:</b>			

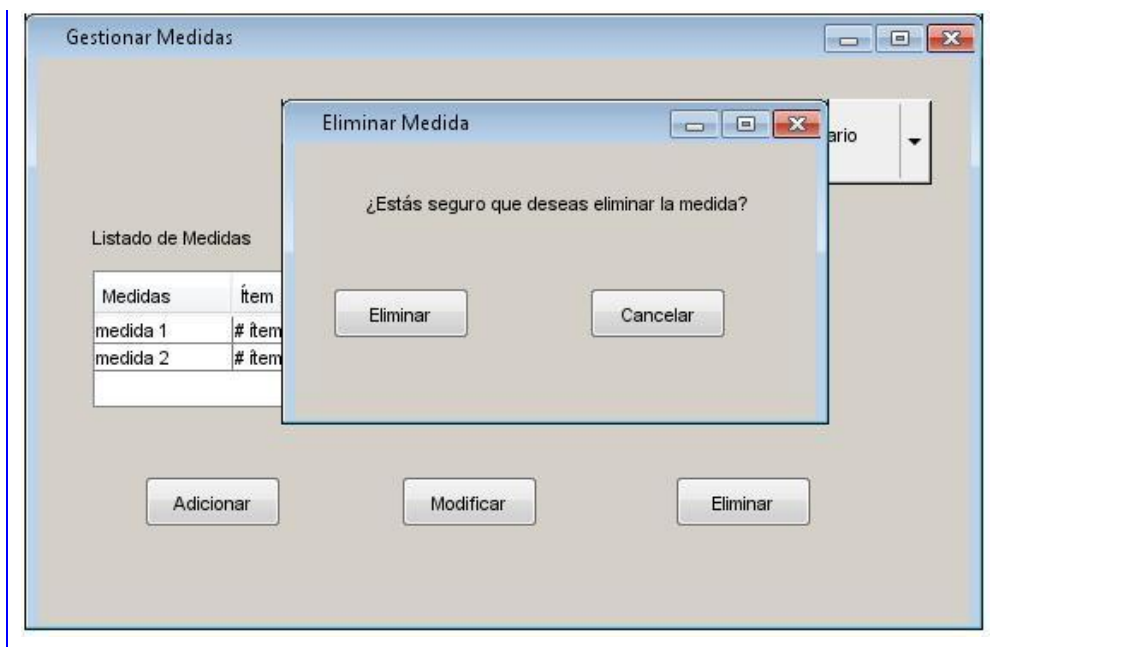


Tabla 64. Historia de Usuario Eliminar medida. Fuente: Elaboración propia

## HU- Gestionar Mejora

### HU-01 Adicionar mejora

<b>Número:</b> HU-18		<b>Nombre del requisito:</b> Adicionar mejora	
<b>Programador:</b> Yeisel Pérez Rivera		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Media		<b>Tiempo Estimado:</b> 2 días	
<b>Riesgo en Desarrollo:</b> N/A		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El usuario podrá adicionar las mejoras al Plan de medidas.			
<b>Observaciones:</b> N/A			
<b>Prototipo de interfaz:</b>			

Tabla 65. Historia de Usuario Adicionar mejora. Fuente: Elaboración propia

### HU-02 Modificar mejora

<b>Número:</b> HU-19		<b>Nombre del requisito:</b> Modificar mejora	
<b>Programador:</b> Yeisel Pérez Rivera		<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Media		<b>Tiempo Estimado:</b> 2 días	
<b>Riesgo en Desarrollo:</b> N/A		<b>Tiempo Real:</b> 1 semana	
<b>Descripción:</b> El usuario podrá modificar las mejoras reflejadas en el Plan de medidas.			
<b>Observaciones:</b> N/A			
<b>Prototipo de interfaz:</b>			

Tabla 66. Historia de Usuario Modificar mejora. Fuente: Elaboración propia

### HU-03 Eliminar mejora

<b>HU-03 Eliminar mejora</b>	
<b>Número:</b> HU-20	<b>Nombre del requisito:</b> Eliminar mejora
<b>Programador:</b> Yeisel Pérez Rivera	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Media	<b>Tiempo Estimado:</b> 2 días
<b>Riesgo en Desarrollo:</b> N/A	<b>Tiempo Real:</b> 1 semana
<b>Descripción:</b> Una vez reflejada una mejora en el Plan de medidas la misma se podrá eliminar si se desea.	
<b>Observaciones:</b> N/A	



**Prototipo de interfaz:**

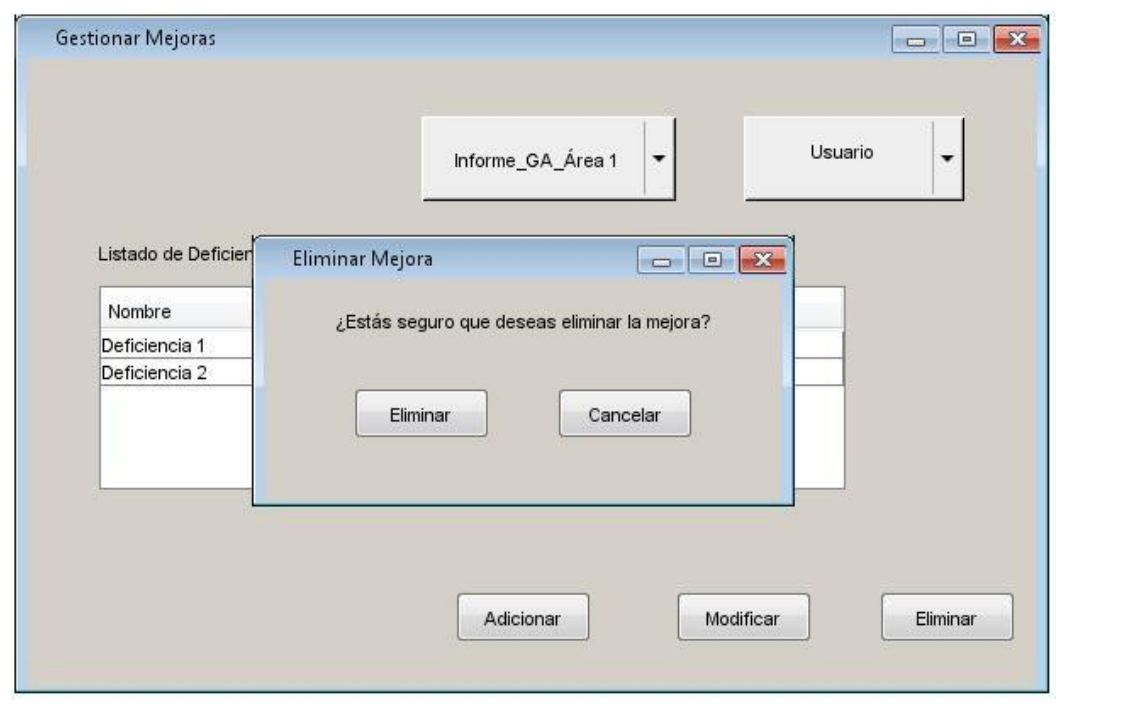


Tabla 67. Historia de Usuario Eliminar mejora. Fuente: Elaboración propia

**HU- Autenticar Usuario**

<b>Número:</b> HU-01	<b>Nombre del requisito:</b> Autenticar usuario
<b>Programador:</b> Yanirosa Almenares	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 3 días
<b>Riesgo en Desarrollo:</b> Alta	<b>Tiempo Real:</b> 1 semana
<b>Descripción:</b> Permite al usuario autenticarse en el sistema.	

**Prototipo de interfaz:**

Tabla 68. Historia de Usuario Autenticar usuario. Fuente: Elaboración propia

**HU- Gestionar Usuario**

**HU-01 Adicionar usuario**

<b>Número:</b> HU-01	<b>Nombre del requisito:</b> Adicionar usuario
<b>Programador:</b> Yanirosa Almenares	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 3 días
<b>Riesgo en Desarrollo:</b> Alta	<b>Tiempo Real:</b> 1 semana
<p><b>Descripción:</b> El administrador encargado de la inserción que debe tener los permisos necesarios y estar autenticado en el sistema, podrá adicionar usuarios al sistema.</p>	

**Prototipo de interfaz:**

Tabla 69. Historia de Usuario Adicionar usuario. Fuente: Elaboración propia

**HU-01 Modificar usuario**

<b>Número:</b> HU-02	<b>Nombre del requisito:</b> Modificar usuario
<b>Programador:</b> Yanirosa Almenares	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 3 días
<b>Riesgo en Desarrollo:</b> Alta	<b>Tiempo Real:</b> 1 semana
<p><b>Descripción:</b> El sistema muestra el listado de los usuarios que existen y el administrador busca el usuario, lo selecciona y modifica sus datos.</p>	

**Prototipo de interfaz:**

Tabla 70. Historia de Usuario Modificar usuario. Fuente: Elaboración propia

**HU-01 Eliminar usuario**

<b>Número:</b> HU-03	<b>Nombre del requisito:</b> Eliminar usuario
<b>Programador:</b> Yanirosa Almenares	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 3 días
<b>Riesgo en Desarrollo:</b> Alta	<b>Tiempo Real:</b> 1 semana
<b>Descripción:</b> El sistema muestra el listado de los usuarios que existen y el administrador busca el/los usuario(s)que desea eliminar.	

Prototipo de interfaz:



Tabla 71. Historia de Usuario Eliminar usuario. Fuente: Elaboración propia

Anexo 4. Diagrama de clases de diseño

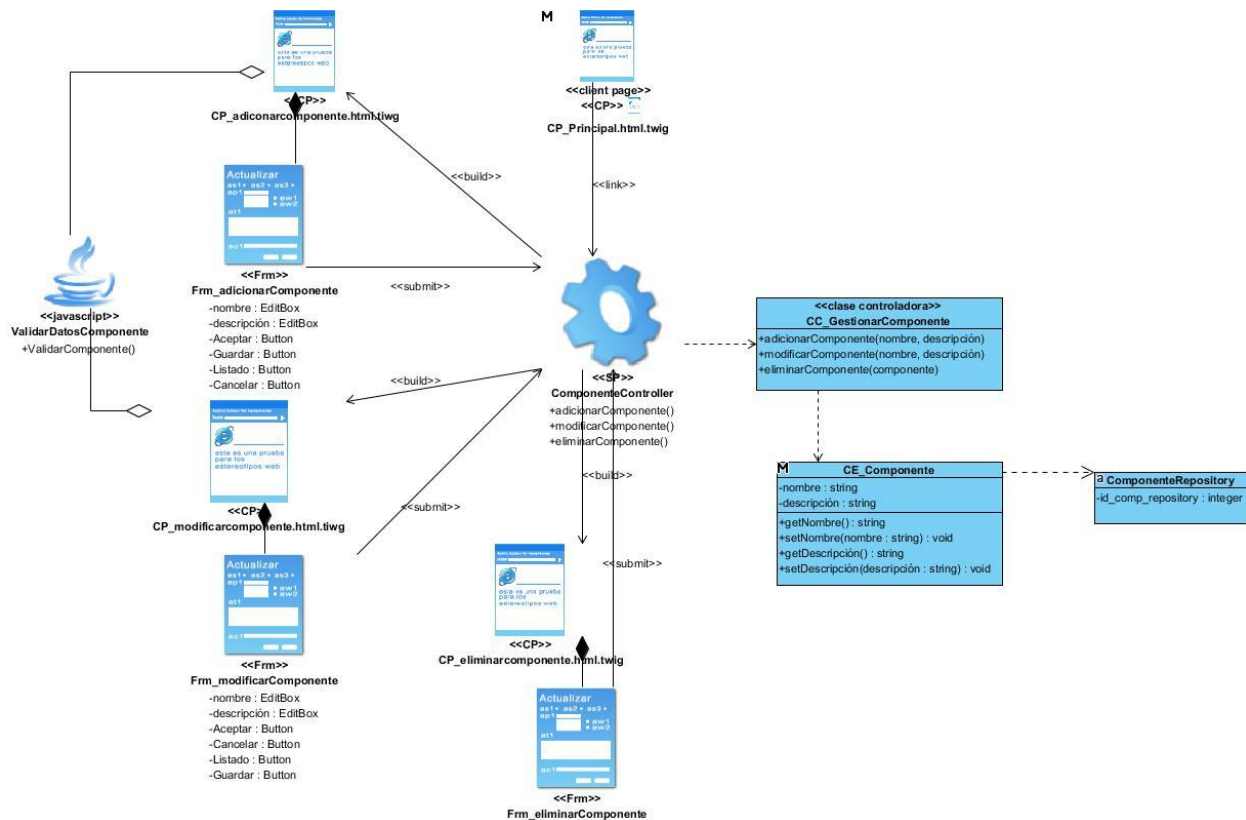


Figura 27. Diagrama de clases de diseño. Escenario Gestionar Componente. Fuente: Elaboración propia.

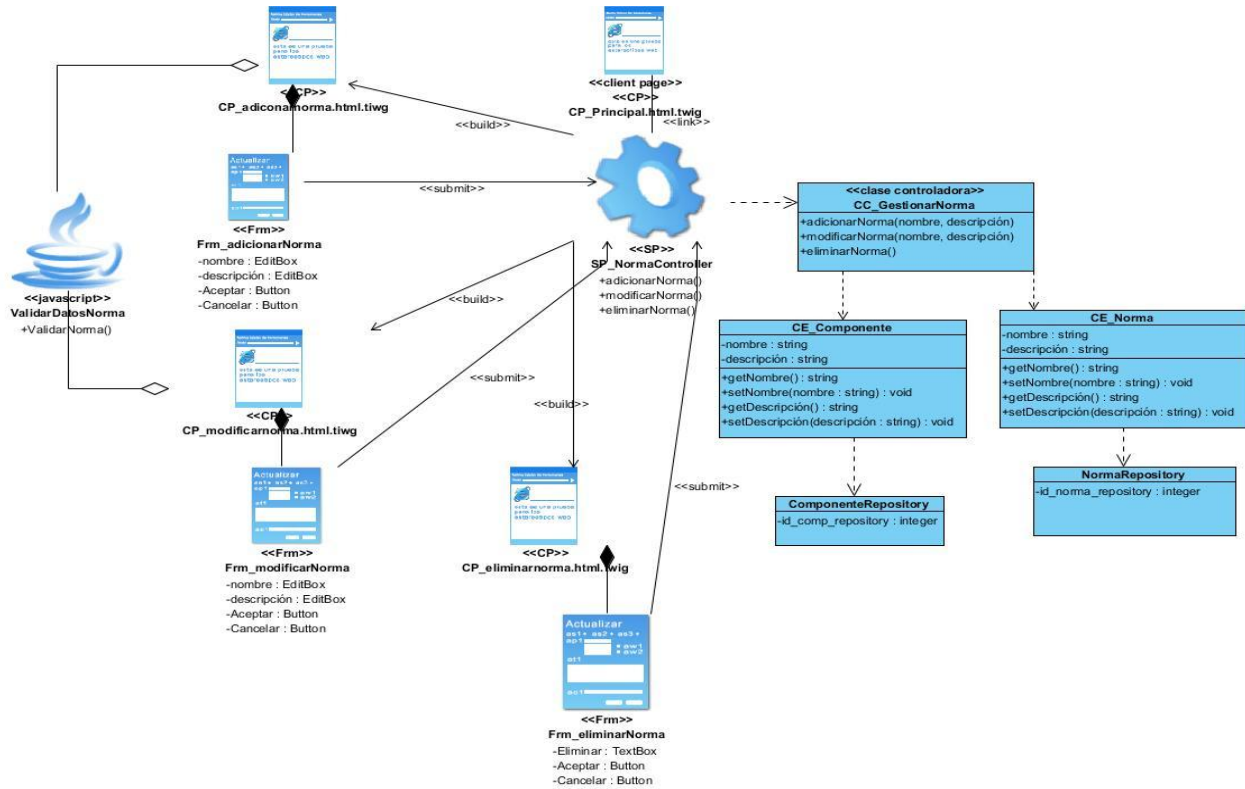


Figura 28. Diagrama de clases de diseño. Escenario Gestionar Norma. Fuente: Elaboración propia.

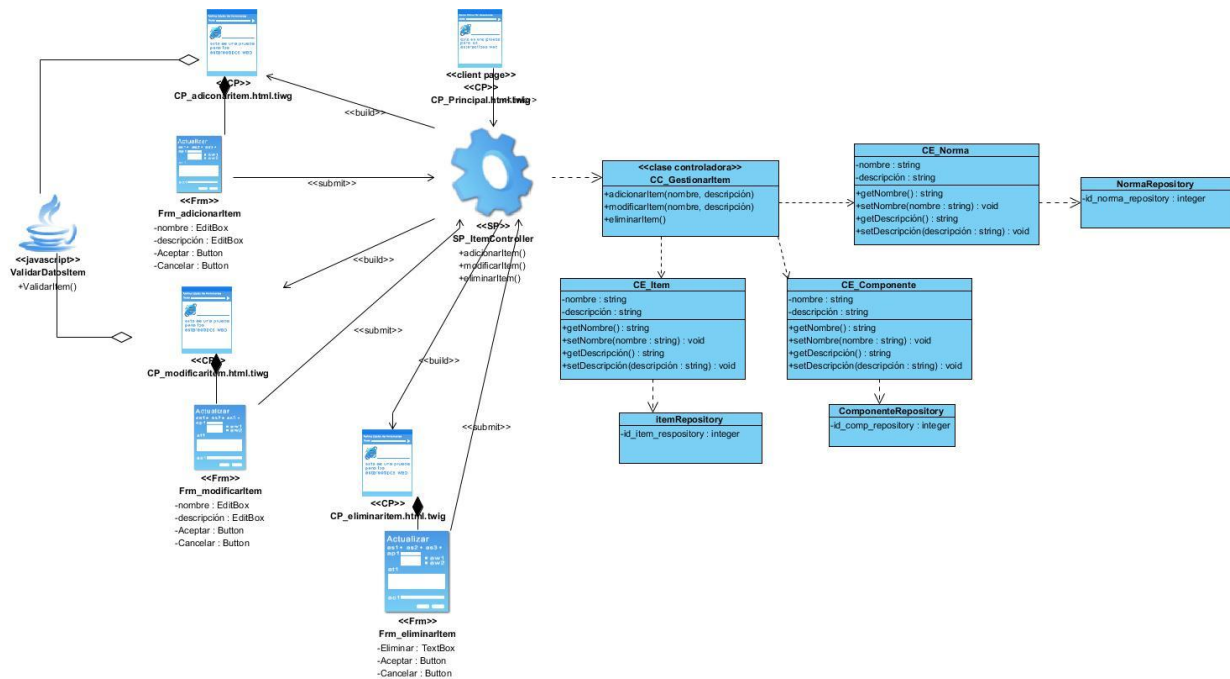


Figura 29. Diagrama de clases de diseño. Escenario Gestionar Ítem. Fuente: Elaboración propia.

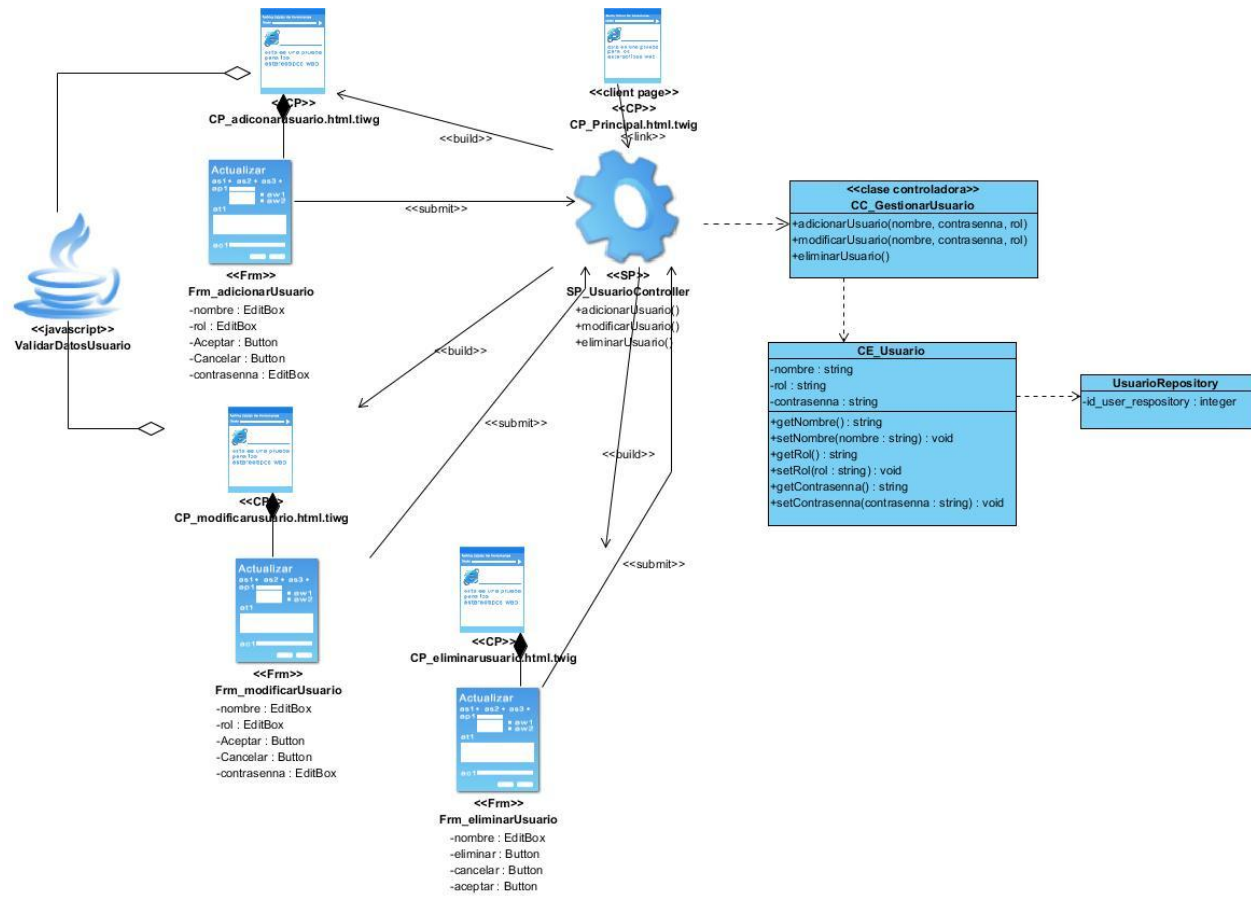


Figura 30. Diagrama de clases de diseño. Escenario Gestionar Usuario. Fuente: Elaboración propia.

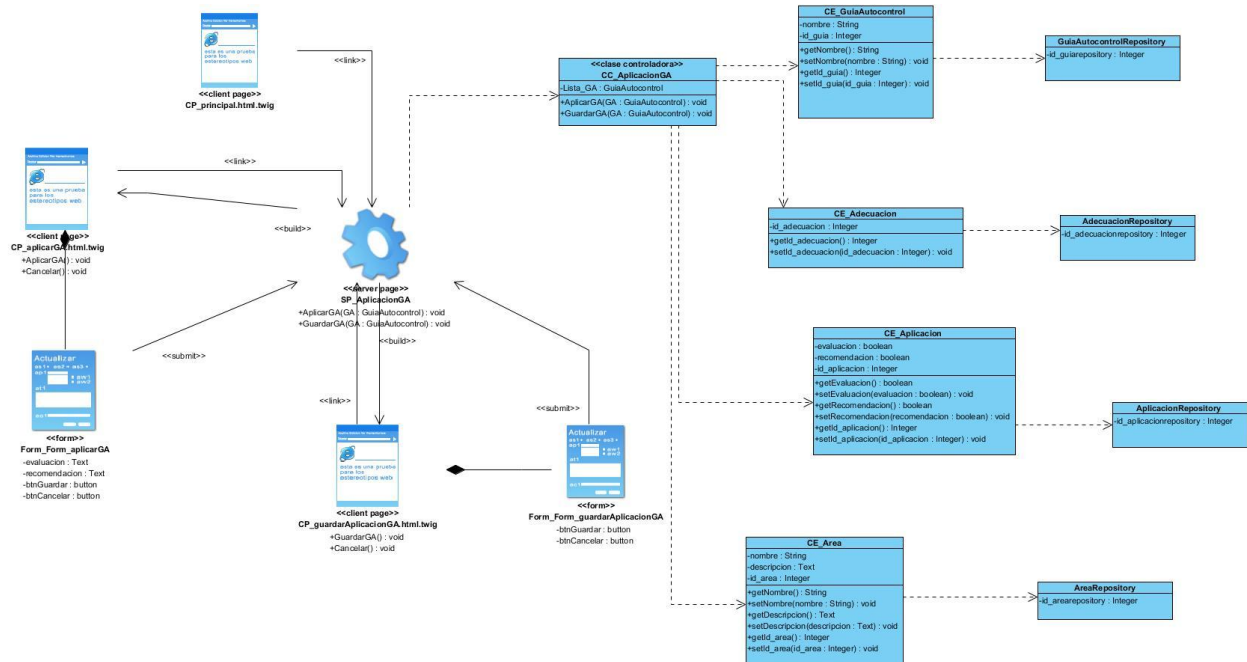


Figura 31. Diagrama de clases de diseño. Escenario Aplicar Guía de Autocontrol. Fuente: Elaboración propia.

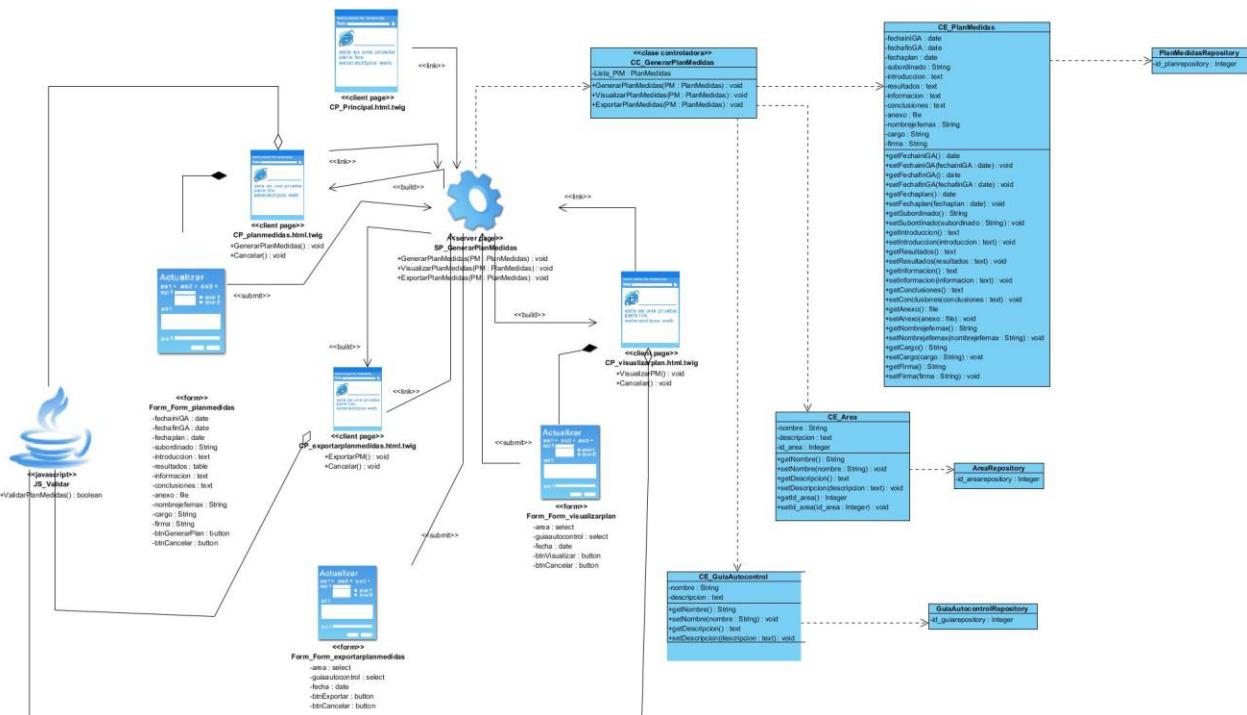


Figura 32. Diagrama de clases de diseño. Escenario Generar Plan de Medidas. Fuente: Elaboración propia.





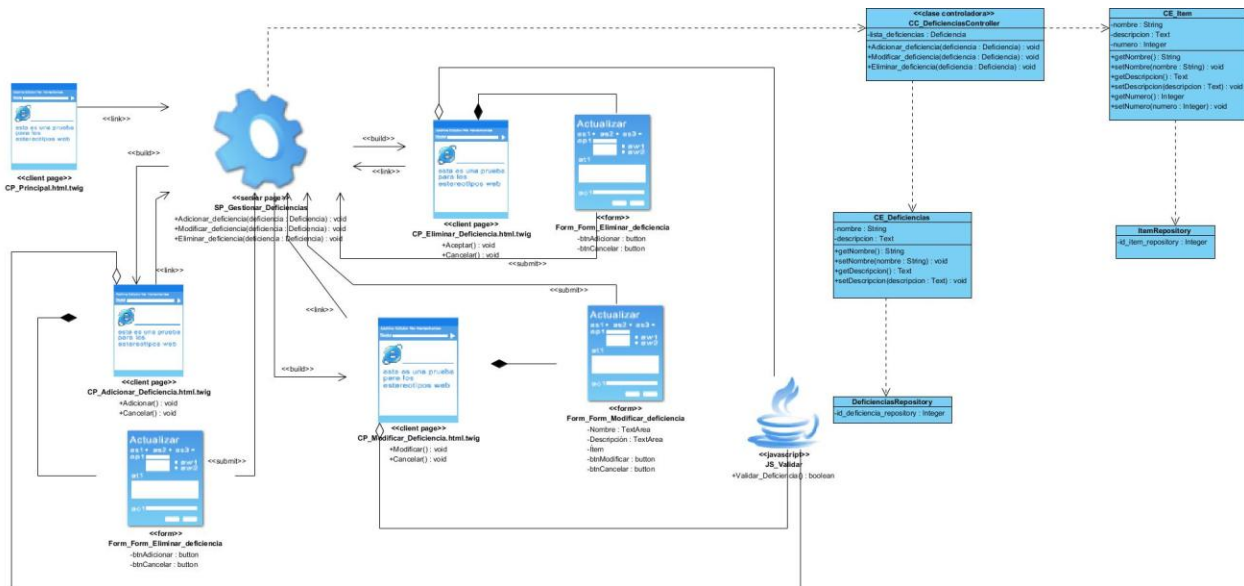


Figura 35. Diagrama de clases de diseño. Escenario Gestionar Deficiencia. Fuente: Elaboración propia.

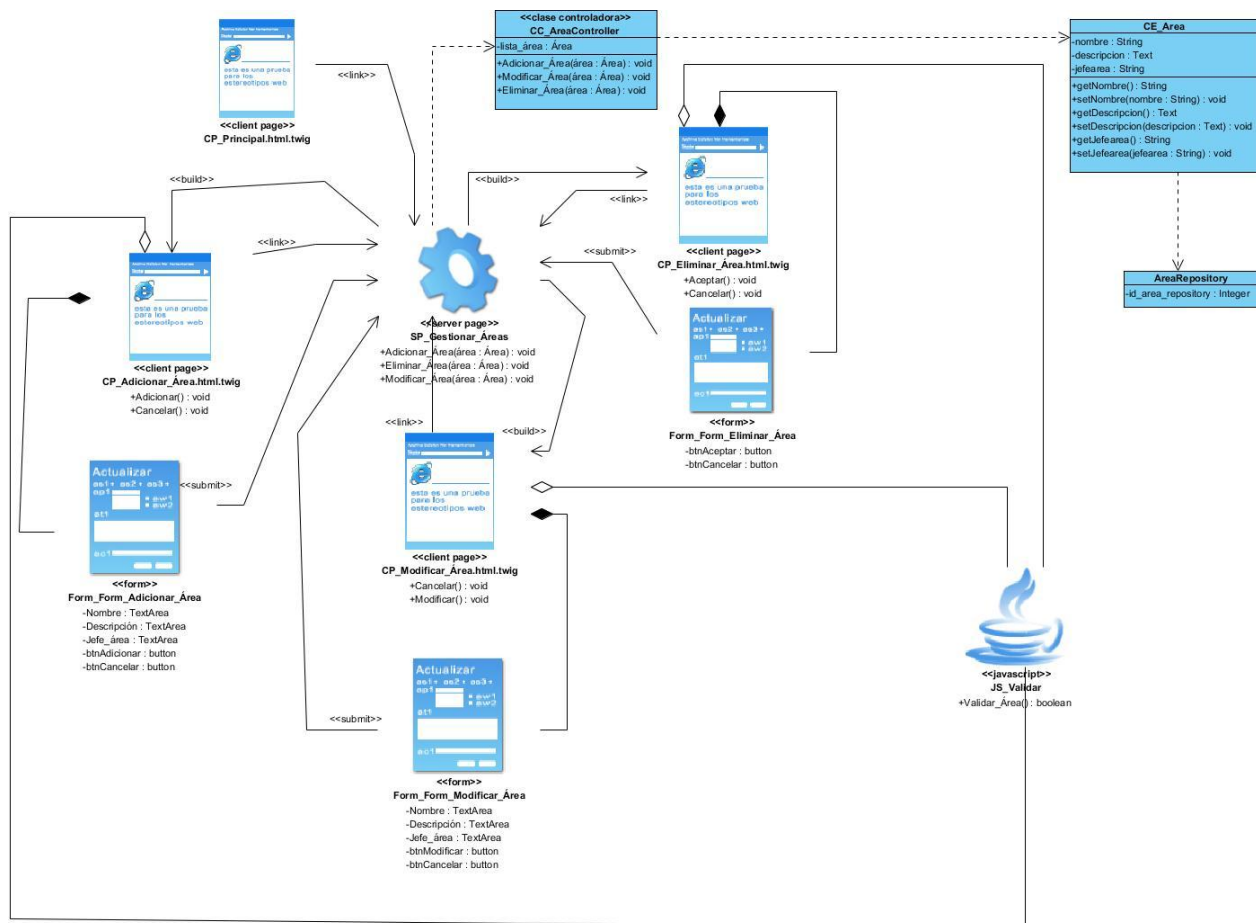


Figura 36. Diagrama de clases de diseño. Escenario Gestionar Área. Fuente: Elaboración propia.

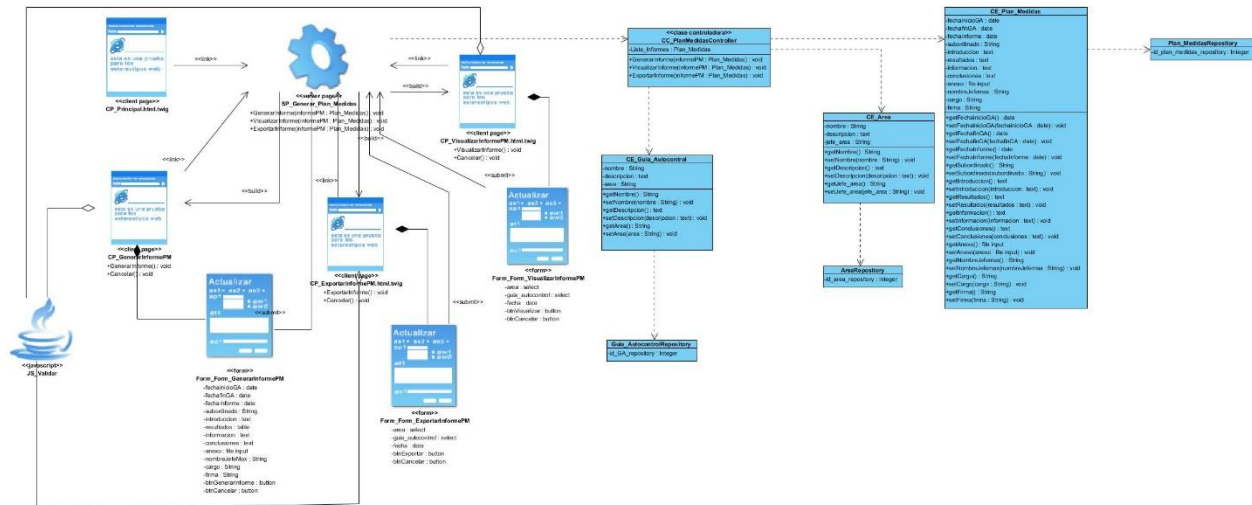


Figura 37. Diagrama de clases de diseño. Escenario Generar Informe. Fuente: Elaboración propia.

## Anexo 5. Casos de Pruebas

### Caso de Prueba Gestionar Guía de Autocontrol

Escenario	Descripción	Nombre	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Adicionar Guía de Autocontrol.	Permite adicionar una Guía de Autocontrol. Todos los campos son obligatorios.	V guía	V esta es una guía	El sistema muestra el mensaje de confirmación: "La Guía de Autocontrol a sido creada satisfactoriamente.". El sistema cierra la interfaz. Se actualiza la lista de las Guía de Autocontrol.	<ol style="list-style-type: none"> <li>1. Seleccionar del menú la opción "Nomencladores".</li> <li>2. Selecciona la opción "Guía de Autocontrol".</li> <li>3. Seleccionar la opción "Adicionar".</li> <li>4. Se muestra una pantalla con los siguientes datos a llenar: nombre, descripción.</li> <li>6. Una vez introducido los valores seleccionar entonces la opción "Aceptar".</li> <li>7. Se actualiza el listado de las Guía de Autocontrol.</li> </ol>
		I guía	V esta es una guía	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	
		V guía	I esta es una guía	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	
EC 1.2 Modificar Guía de Autocontrol.	Permite modificar valores existentes de la Guía de Autocontrol.	V guía uno	V esta es una guía modificada	El sistema muestra el mensaje de confirmación: "La Guía de Autocontrol ha sido modificada satisfactoriamente.". El sistema cierra la interfaz.	<ol style="list-style-type: none"> <li>1. Ejecutar el flujo central del EC 1.1.</li> <li>2. Seleccionar la Guía de Autocontrol. que se desea modificar.</li> <li>3. Seleccionar la opción "Modificar".</li> <li>4. Se muestra una pantalla con los valores de la Guía de Autocontrol seleccionada.</li> <li>5. Una vez introducido los cambios deseados seleccionar entonces la opción "Modificar".</li> <li>6. Se actualiza el listado de las Guía de Autocontrol.</li> </ol>
		I guía	V esta es una guía modificada	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	
		V guía uno	I esta es una guía	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	
EC 1.3 Eliminar Guía de Autocontrol.	Permite eliminar la/las Guía de Autocontrol deseada(s).	V guía	V esta es una guía	El sistema muestra el mensaje de confirmación: "Guía de Autocontrol eliminada". El sistema cierra la interfaz.	<ol style="list-style-type: none"> <li>1. Ejecutar el flujo central del EC 1.1.</li> <li>2. Seleccionar la/las Guía de Autocontrol que se desea(n) eliminar.</li> <li>3. Seleccionar la opción "Mostrar".</li> <li>4. Seleccionar la opción "Eliminar".</li> <li>5. Se muestra un mensaje preguntándole al usuario que si está seguro que desea eliminar las Guía de Autocontrol seleccionada.</li> <li>6. Seleccionar entonces la opción "Aceptar".</li> <li>7. Se actualiza el listado de las Guía de Autocontrol.</li> </ol>

Tabla 72. Caso de Prueba Gestionar Guía de Autocontrol. Fuente: Elaboración propia.

### Caso de Prueba Gestionar Componente

Escenario	Descripción	Nombre	Descripción	Guía Autocontrol	Respuesta del sistema	Flujo central
EC 1.1 Adicionar componente.	Permite adicionar componente en el listado de estos.  Todos los campos son obligatorios.	V Ambiente de Control	V Establece las pautas para el funcionamiento legal y armónico.	V Guía de Autocontrol General.	V El sistema muestra el mensaje de confirmación: "El componente ha sido creado satisfactoriamente." El sistema cierra la interfaz. Se actualiza la lista de componentes.	1. Seleccionar del menú la opción "Nomencladores". 2. Selecciona la opción "Componente". 3. Seleccionar la opción "Adicionar". 4. Se muestra una pantalla con los siguientes datos a llenar: nombre, descripción, Guía de Autocontrol. 6. Una vez introducido los valores seleccionar entonces la opción "Guardar". 7. Se actualiza el listado de componentes.
		I Ambiente de Control233	V Establece las pautas para el funcionamiento legal y armónico.	V Guía de Autocontrol General.	V El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	
		V Ambiente de Control	I Establece las pautas para el funcionamiento legal y armónico.	V Guía de Autocontrol General.	V El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	
		V Ambiente de Control	V Establece las pautas para el funcionamiento legal y armónico.	I Guíae34 de Autocontrol General.	V El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	
EC 1.2 Modificar componente.	Permite modificar valores existentes de un componente seleccionado.	V Ambiente de Control	V Establece las pautas para el funcionamiento legal y armónico.	V Guía de Autocontrol General.	V El sistema muestra el mensaje de confirmación: "El componente ha sido modificado satisfactoriamente." El sistema cierra la interfaz. Se actualiza la lista de componentes.	1. Ejecutar el flujo central del EC 1.1. 2. Seleccionar el componente que se desea modificar. 3. Seleccionar la opción "Modificar". 4. Se muestra una pantalla con los valores del componente seleccionado. 5. Una vez introducido los cambios deseados seleccionar entonces la opción "Modificar". 6. Se actualiza el listado de los componentes.
		I Ambiente de Control233	V Establece las pautas para el funcionamiento legal y armónico.	V Guía de Autocontrol General.	V El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	
		V Ambiente de Control	I Establece las pautas para el funcionamiento legal y armónico.	V Guía de Autocontrol General.	V El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	
		V Ambiente de Control	V Establece las pautas para el funcionamiento legal y armónico.	I Guíae34 de Autocontrol General.	V El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	
EC 1.3 Eliminar componente.	Permite eliminar el/los componente(s) deseado(s).	V Ambiente de Control	V Estable las pautas para el funcionamiento legal y armónico.	V Guía de Autocontrol General.	V El sistema muestra el mensaje de confirmación: "Componente eliminado." El sistema cierra la interfaz.	1. Ejecutar el flujo central del EC 1.1. 2. Seleccionar el/los componente(s) que se desea(n) eliminar. 3. Seleccionar la opción "Mostrar". 4. Seleccionar la opción "Eliminar". 5. Se muestra un mensaje preguntándole al usuario que si está seguro que desea eliminar los componentes seleccionados. 6. Seleccionar entonces la opción "Aceptar". 7. Se actualiza el listado de los componentes.

Tabla 73. Caso de Prueba Gestionar Componente. Fuente: Elaboración propia.

### Caso de Prueba Gestionar Norma

Escenario	Descripción	Componente	Nombre	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Adicionar norma.	Permite adicionar norma en el listado de estas.  Todos los campos son obligatorios.	V Ambiente de Control	V norma	V esta es una norma	El sistema muestra el mensaje de confirmación: "La norma ha sido creada satisfactoriamente.". El sistema cierra la interfaz. Se actualiza la lista de normas.	1. Seleccionar del menú la opción "Nomencladores". 2. Selecciona la opción "Norma". 3. Seleccionar la opción "Adicionar". 4. Se muestra una pantalla con los siguientes datos a llenar: nombre, descripción. 6. Una vez introducido los valores seleccionar entonces la opción "Aceptar". 7. Se actualiza el listado de normas.
		I Ambiente de Controlerr443	V norma	V esta es una norma	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos	
		V Ambiente de Control	I norm4	V esta es una norma	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos	
		V Ambiente de Control	V norma	I esta es una norma4	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos	
EC 1.2 Modificar norma.	Permite modificar valores existentes de una norma seleccionada.	V Ambiente de Control	V norma	V esta es una norma	El sistema muestra el mensaje de confirmación: "La norma ha sido modificada satisfactoriamente.". El sistema cierra la interfaz. Se actualiza la lista de normas.	1. Ejecutar el flujo central del EC 1.1. 2. Seleccionar la norma que se desea modificar. 3. Seleccionar la opción "Modificar". 4. Se muestra una pantalla con los valores de la norma seleccionada. 5. Una vez introducido los cambios deseados seleccionar entonces la opción "Modificar". 6. Se actualiza el listado de las normas.
		I Ambiente de Controlerr443	V norma	V esta es una norma	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos	
		V Ambiente de Control	I norm4	V esta es una norma	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos	
		V Ambiente de Control	V norma	I esta es una norma4	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos	
EC 1.3 Eliminar norma	Permite eliminar la/s norma(s) deseada(s).	V Ambiente de Control	V norma	V esta es una norma	El sistema muestra el mensaje de confirmación: Norma eliminada". El sistema cierra la interfaz.	1. Ejecutar el flujo central del EC 1.1. 2. Seleccionar la/s norma(s) que se desea(n) eliminar. 3. Seleccionar la opción "Mostrar". 4. Seleccionar la opción "Eliminar". 5. Se muestra un mensaje preguntándole al usuario que si está seguro que desea eliminar la/s norma(s) seleccionada(s). 6. Seleccionar entonces la opción "Aceptar". 7. Se actualiza el listado de las normas.

Tabla 74. Caso de Prueba Gestionar Norma. Fuente: Elaboración propia.

## Caso de Prueba Gestionar Ítem

Escenario	Descripción	Nombre	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Adicionar ítem.	Permite adicionar ítem en el listado de estos.  Todos los campos son obligatorios.	V ítem	V este es un ítem	El sistema muestra el mensaje de confirmación: "El ítem ha sido creado satisfactoriamente.". El sistema cierra la interfaz. Se actualiza la lista de ítems.	1. Seleccionar del menú la opción "Nomencladores". 2. Selecciona la opción "ítem". 3. Seleccionar la opción "Adicionar". 4. Se muestra una pantalla con los siguientes datos a llenar: nombre, descripción, 5. Una vez introducido los valores seleccionar entonces la opción "Aceptar". 6. Se actualiza el listado de ítems.
		V ítem	I este es un ítem 2222@	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	
		I ítem213	V este es un ítem	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	
EC 1.2 Modificar ítem.	Permite modificar valores existentes de un ítem seleccionado.	V ítem modificado	V este es un ítem modificado	El sistema muestra el mensaje de confirmación: "El ítem ha sido modificado satisfactoriamente.". El sistema cierra la interfaz.	1. Ejecutar el flujo central del EC 1.1. 2. Seleccionar el ítem que se desea modificar. 3. Seleccionar la opción "Modificar". 4. Se muestra una pantalla con los valores del ítem seleccionado. 5. Una vez introducido los cambios deseados seleccionar entonces la opción "Modificar". 6. Se actualiza el listado de los ítems.
		V ítem	I este es un ítem mal 123lm@#	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	
		I íte23lmb	V este es un ítem modificado	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores	
EC 1.3 Eliminar ítem	Permite eliminar el/los ítem(s) deseado(s).	V ítem	V este es un ítem	El sistema muestra el mensaje de confirmación: "Ítem eliminado.". El sistema cierra la interfaz.	1. Ejecutar el flujo central del EC 1.1. 2. Seleccionar el/los ítem(s) que se desea(n) eliminar. 3. Seleccionar la opción "Mostrar". 4. Seleccionar la opción "Eliminar". 5. Se muestra un mensaje preguntándole al usuario que si está seguro que desea eliminar los ítems seleccionados. 6. Seleccionar entonces la opción "Aceptar". 7. Se actualiza el listado de los ítems.

Tabla 75. Caso de Prueba Gestionar Ítem. Fuente: Elaboración propia.

## Caso de Prueba Gestionar Área

Escenario	Descripción	Nombre	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Adicionar área.	Permite adicionar área en el listado de estas.  Todos los campos son obligatorios.	V Decanato  V Decanato  I Decan4to	V En esta área se encuentran los máximos dirigentes de la facultad y en ella se controlan los acuerdos y  I En esta área 2222 se encuentran los máximos dirigentes de la facultad y en ella se controlan los  V En esta área se encuentran los máximos dirigentes de la facultad y en ella se controlan los acuerdos y actividades	El sistema muestra el mensaje de confirmación: "El área ha sido creada satisfactoriamente." El sistema cierra la interfaz. Se actualiza la lista de áreas.  El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.  El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	1. Seleccionar del menú la opción "Nomencladores". 2. Selecciona la opción "Área". 3. Seleccionar la opción "Adicionar". 4. Se muestra una pantalla con los siguientes datos a llenar: nombre, descripción. 6. Una vez introducido los valores seleccionar entonces la opción "Guardar". 7. Se actualiza el listado de áreas.
EC 1.2 Modificar área.	Permite modificar valores existentes de un área seleccionada.	V Decanato uno  V Decanato  I Decan4to	V esta es un área modificada  I En esta área 2222 se encuentran los máximos dirigentes de la facultad y en ella se controlan los acuerdos y actividades asignadas a la facultad.  V En esta área se encuentran los máximos dirigentes de la facultad y en ella	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.  El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.  El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	1. Ejecutar el flujo central del EC 1.1. 2. Seleccionar el área que se desea modificar. 3. Seleccionar la opción "Modificar". 4. Se muestra una pantalla con los valores del área seleccionado. 5. Una vez introducido los cambios deseados seleccionar entonces la opción "Modificar". 6. Se actualiza el listado de las áreas.
EC 1.3 Eliminar área	Permite eliminar ellas áreas(s) deseada(s).	V Decanato uno	V esta es un área modificada	El sistema muestra el mensaje de confirmación: "Área eliminado." El sistema cierra la interfaz.	1. Ejecutar el flujo central del EC 1.1. 2. Seleccionar ellas área(s) que se desea(n) eliminar. 3. Seleccionar la opción "Mostrar". 4. Seleccionar la opción "Eliminar". 5. Se muestra un mensaje preguntándole al usuario que si está seguro que desea eliminar las áreas seleccionadas. 6. Seleccionar entonces la opción "Aceptar".

Tabla 76. Caso de Prueba Gestionar Área. Fuente: Elaboración propia.

## Caso de Prueba Gestionar Usuario

Escenario	Descripción	Nombre	Email	Área	Contraseña	Respuesta del sistema	Flujo central	
EC 1.1 Adicionar usuario.	Permite adicionar usuario en el listado de estos.  Todos los campos son obligatorios.	V	usuario	usuario@uclcu	área uno	V	usuario	<ol style="list-style-type: none"> <li>1. Seleccionar del menú la opción "Gestión".</li> <li>2. Seleccionar la opción "Gestionar usuarios".</li> <li>3. Seleccionar la opción "Adicionar".</li> <li>4. Se muestra una pantalla con los siguientes datos a llenar: nombre, email, área, contraseña.</li> <li>6. Una vez introducido los valores seleccionar entonces la opción "Aceptar".</li> <li>7. Se actualiza el listado de usuarios</li> </ol>
		I	usuario56773	usuario@uclcu	área uno	V	usuario	
		V	usuario	user.*@uclcu	área uno	V	usuario	
		V	usuario	usuario@uclcu	área****668	V	usuario	
EC 1.2 Modificar usuario.	Permite modificar valores existentes de un usuario seleccionado.	user	user@uclcu	área dos	V	usuario	<ol style="list-style-type: none"> <li>1. Ejecutar el flujo central del EC 1.1.</li> <li>2. Seleccionar el usuario que se desea modificar.</li> <li>3. Seleccionar la opción "Modificar".</li> <li>4. Se muestra una pantalla con los valores del usuario seleccionado.</li> <li>5. Una vez introducido los cambios deseados seleccionar entonces la opción "Modificar".</li> <li>6. Se actualiza el listado de los usuarios.</li> </ol>	
		I	usuario56773	usuario@uclcu	área uno	V		usuario
		V	usuario	user.*@uclcu	área uno	V		usuario
		V	usuario	usuario@uclcu	área****668	V		usuario
EC 1.3 Eliminar usuario.	Permite eliminar el/los usuario(s) deseado(s).	usuario	usuario@uclcu	área uno	V	usuario	<ol style="list-style-type: none"> <li>1. Ejecutar el flujo central del EC 1.1.</li> <li>2. Seleccionar el/los usuario(s) que se desea(n) eliminar.</li> <li>3. Seleccionar la opción "Mostrar".</li> <li>4. Seleccionar la opción "Eliminar".</li> <li>5. Se actualiza el listado de los usuarios.</li> </ol>	

Tabla 77. Caso de Prueba Gestionar Usuario. Fuente: Elaboración propia.

## Caso de Prueba Gestionar Deficiencia



Escenario	Descripción	Nombre	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Adicionar deficiencia.	Permite adicionar deficiencia en el listado de estas.  Todos los campos son obligatorios.	V deficiencia  I deficiencia56  V deficiencia	V esta es una deficiencia  V esta es una deficiencia  I esta es una deficiencia4	El sistema muestra el mensaje de confirmación: "La deficiencia ha sido creada satisfactoriamente.". El sistema cierra la interfaz. Se actualiza la lista de deficiencias.  El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos  El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos	1. Seleccionar del menú la opción "Gestión". 2. Selecciona la opción "Deficiencias". 3. Seleccionar la opción "Adicionar". 4. Se muestra una pantalla con los siguientes datos a llenar: nombre, descripción 6. Una vez introducido los valores seleccionar entonces la opción "Aceptar". 7. Se actualiza el listado de deficiencias.
EC 1.2 Modificar deficiencia.	Permite modificar valores existentes de una deficiencia seleccionada.	V deficiencia uno  I deficiencia56  V deficiencia	V esta es una deficiencia modificada  V esta es una deficiencia  I esta es una deficiencia4	El sistema muestra el mensaje de confirmación: "La deficiencia ha sido modificada satisfactoriamente.". El sistema cierra la interfaz.  El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos  El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos	1. Ejecutar el flujo central del EC 1.1. 2. Seleccionar la deficiencia que se desea modificar. 3. Seleccionar la opción "Modificar". 4. Se muestra una pantalla con los valores de la deficiencia seleccionada. 5. Una vez introducido los cambios deseados seleccionar entonces la opción "Modificar". 6. Se actualiza el listado de las deficiencias.
EC 1.3 Eliminar deficiencia	Permite eliminar la/s deficiencia(s) deseada(s).	V deficiencia uno	V esta es una deficiencia modificada	El sistema muestra el mensaje de confirmación: "Deficiencia eliminada". El sistema cierra la interfaz.	1. Ejecutar el flujo central del EC 1.1. 2. Seleccionar la/s deficiencia(s) que se desea(n) eliminar. 3. Seleccionar la opción "Mostrar". 4. Seleccionar la opción "Eliminar". 5. Se muestra un mensaje preguntándole al usuario que si está seguro que desea eliminar la/s deficiencia(s) seleccionada(s). 6. Seleccionar entonces la opción "Aceptar". 7. Se actualiza el listado de las deficiencias.

Tabla 78. Caso de Prueba Gestionar Deficiencia. Fuente: Elaboración propia.

## Caso de Prueba Gestionar Medida

Escenario	Descripción	Nombre	Descripción	Deficiencia	Respuesta del sistema	Flujo central
EC 1.1 Adicionar medidas.	Permite adicionar la medida en el listado de estas.  Todos los campos son obligatorios.	V medida	V esta es una medida	V deficiencia 1	El sistema muestra el mensaje de confirmación: "La medida ha sido adicionada satisfactoriamente.". El sistema cierra la interfaz. Se actualiza la lista de medidas.	1. Seleccionar del menú la opción "Gestión". 2. Selecciona la opción "Gestionar Medidas". 3. Seleccionar la opción "Adicionar". 4. Se muestra una pantalla con los siguientes datos a llenar: nombre, descripción, deficiencia. 5. Una vez introducido los valores seleccionar entonces la opción "Aceptar". 7. Se actualiza el listado de deficiencias.
		I medidaRR56	V esta es una medida	V deficiencia 1	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores	
		V medida	I mesta es una 24*	V deficiencia 1	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores	
		V medida	V esta es una medida	I deficiencia1223*	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	
EC 1.2 Modificar medida.	Permite modificar valores existentes de una medida seleccionada.	V medida	V esta es una medida modificada	V12 deficiencia 1	El sistema muestra el mensaje de confirmación: "La medida ha sido modificada satisfactoriamente.". El sistema cierra la interfaz.	1. Ejecutar el flujo central del EC 1.1. 2. Seleccionar la medida que se desea modificar. 3. Seleccionar la opción "Modificar". 4. Se muestra una pantalla con los valores de la medida seleccionada. 5. Una vez introducido los cambios deseados seleccionar entonces la opción "Modificar". 6. Se actualiza el listado de las medidas.
		I medidaRR56	V esta es una medida	V deficiencia 1	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	
		V medida	I mesta es una 24*	V deficiencia 1	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores	
		V medida	V esta es una medida	I deficiencia1223*	El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores	
EC 1.3 Eliminar medida.	Permite eliminar la/las medidas(s) deseada(s).	V medida	V esta es una medida modificada	V1.1 deficiencia 1	El sistema muestra el mensaje de confirmación: "Medida eliminada.". El sistema cierra la interfaz.	1. Ejecutar el flujo central del EC 1.1. 2. Seleccionar la/las medidas(s) que se desea(n) eliminar. 3. Seleccionar la opción "Mostrar". 4. Seleccionar la opción "Eliminar". 5. Se muestra un mensaje preguntándole al usuario que si está seguro que desea eliminar las medidas seleccionadas. 6 Seleccionar entonces la opción "Aceptar". 7. Se actualiza el listado de las medidas.

Tabla 79. Caso de Prueba Gestionar Medida. Fuente: Elaboración propia.

## Caso de Prueba Gestionar Mejora

Escenario	Descripción	Nombre	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Adicionar mejora.	Permite adicionar una mejora.  Todos los campos son obligatorios.	V mejora  I mejora 233  V mejora	V esta es una mejora  V esta es una mejora  I esta es "67" 1 una mejora	El sistema muestra el mensaje de confirmación: "La mejora ha sido creada satisfactoriamente." El sistema cierra la interfaz. Se actualiza la lista de las mejoras.  El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.  El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	1. Seleccionar del menú la opción "Gestión". 2. Selecciona la opción "Mejoras". 3. Seleccionar la opción "Adicionar". 4. Se muestra una pantalla con los siguientes datos a llenar: nombre, descripción, 6. Una vez introducido los valores seleccionar entonces la opción "Aceptar". 7. Se actualiza el listado de las mejoras.
EC 1.2 Modificar mejora.	Permite modificar valores existentes de la mejora.	V mejora uno  I mejora 233  V mejora	V esta es 1 una mejora  V esta es una mejora  I esta es "67" 1 una mejora	El sistema muestra el mensaje de confirmación: "La mejora ha sido modificada satisfactoriamente." El sistema cierra la interfaz  El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.  El sistema muestra un mensaje informando que existen campos erróneos. Se muestra un mensaje de error en rojo debajo del campo con valores incorrectos.	1. Ejecutar el flujo central del EC 1.1. 2. Seleccionar la mejora que se desea modificar. 3. Seleccionar la opción "Modificar". 4. Se muestra una pantalla con los valores de la mejora seleccionada. 5. Una vez introducido los cambios deseados seleccionar entonces la opción "Modificar". 6. Se actualiza el listado de las mejoras.
EC 1.3 Eliminar mejora.	Permite eliminar la/s mejoras(s).	V mejora	V esta es una mejora	El sistema muestra el mensaje de confirmación: "Mejora eliminada". El sistema cierra la interfaz.	1. Ejecutar el flujo central del EC 1.1. 2. Seleccionar la/s mejora que se desea(n) eliminar. 3. Seleccionar la opción "Mostrar". 4. Seleccionar la opción "Eliminar". 5. Se muestra un mensaje preguntándole al usuario que si está seguro que desea eliminar la mejora seleccionada. 6. Seleccionar entonces la opción "Aceptar". 7. Se actualiza el listado de las mejoras.

Tabla 80. Caso de Prueba Gestionar Mejora. Fuente: Elaboración propia.

**Caso de prueba para el camino básico 2:**

Camino:1-2-3-4-6-7-6-8-9-10	
<b>Descripción</b>	Los datos de entrada son correctos y cumplen con el formato indicado.
<b>Condición de ejecución</b>	<b>Campos válidos:</b> El nombre de la adecuación será Adecuación del Decanato. El área será Decanato. El ítem será nulo. El Subítem será las funciones definidas de cada área y puesto de trabajo.
<b>Entrada</b>	<b>Nombre de la Adecuación:</b> Adecuación del Decanato. <b>Área:</b> Decanato. <b>Ítem:</b> nulo.

	<b>Subítem:</b> Las funciones definidas de cada área y puesto de trabajo.
<b>Resultados esperados</b>	Se espera que se le notifique al usuario que el ítem no puede ser nulo.

Tabla 81. Caso de Prueba del camino básico 2. Fuente: Elaboración propia

**Caso de prueba para el camino básico 3:**

<b>Camino: 1-2-3-4-5-4-6-8-9-10</b>	
<b>Descripción</b>	Los datos de entrada son correctos y cumplen con el formato indicado.
<b>Condición de ejecución</b>	<b>Campos válidos:</b> El nombre de la adecuación será Adecuación del Decanato. El área será Decanato. El ítem será la elaboración del plan anual de actividades. El Subítem nulo.
<b>Entrada</b>	<b>Nombre de la Adecuación:</b> Adecuación del Decanato. <b>Área:</b> Decanato. <b>Ítem:</b> La elaboración del plan anual de actividades. <b>Subítem:</b> nulo.
<b>Resultados esperados</b>	Se espera que se le notifique al usuario que el subítem no puede ser nulo.

Tabla 82. Caso de Prueba del camino básico 3. Fuente: Elaboración propia

**Caso de prueba para el camino básico 4:**

<b>Camino: 1-2-3-8-9-10</b>	
<b>Descripción</b>	Los datos de entrada son correctos y cumplen con el formato indicado.
<b>Condición de ejecución</b>	<b>Campos válidos:</b> El nombre de la adecuación será Adecuación del Decanato. El área será Decanato. El ítem será nulo. El Subítem será nulo.
<b>Entrada</b>	<b>Nombre de la Adecuación:</b> Adecuación del Decanato.

	<p><b>Área:</b> Decanato.</p> <p><b>Ítem:</b>nulo.</p> <p><b>Subítem:</b>nulo.</p>
<b>Resultados esperados</b>	Se espera que se le notifique al usuario que el ítem y el subítem no pueden ser nulos.

Tabla 83. Caso de Prueba del camino básico 4. Fuente: Elaboración propia

**Caso de prueba para el camino básico 5:**

<b>Camino: 1-10</b>	
<b>Descripción</b>	Los datos de entrada son correctos y cumplen con el formato indicado.
<b>Condición de ejecución</b>	<p><b>Campos válidos:</b></p> <p>El nombre de la adecuación será nulo.</p> <p>El área será nulo.</p> <p>El ítem será nulo.</p> <p>El Subítem será nulo.</p>
<b>Entrada</b>	<p><b>Nombre de la Adecuación:</b>nulo.</p> <p><b>Área:</b>nulo.</p> <p><b>Ítem:</b>nulo.</p> <p><b>Subítem:</b>nulo.</p>
<b>Resultados esperados</b>	Se espera que se le notifique al usuario que para realizar una adecuación de la Guía de Autocontrol ninguno de los campos pueden ser nulos.

Tabla 84. Caso de Prueba del camino básico 5. Fuente: Elaboración propia

**Anexo 6. Entrevista realizada:**

**ENTREVISTA**

**Objetivo de la entrevista:** reunir información relevante para el desarrollo del módulo CI del Sistema de Administración y Economía de la Facultad 3 y conocer todo lo referente a cómo se maneja el CI en el país.

**Datos del entrevistado:**

Nombre y apellidos: Msc. Ana Marys García Rodríguez

Cargo que ocupa: Vicedecana de Administración y Economía de la Facultad 3.

## Preguntas:

1. ¿Qué es CI?
2. ¿Para qué se realiza el CI?
3. ¿Qué conceptos maneja el CI de la Facultad?
4. ¿Qué es una Guía de Autocontrol (GA)?
5. ¿Qué es un componente?
6. ¿Qué es una norma?
7. ¿Qué es un ítem?
8. ¿Cómo se realiza el CI en la Facultad?
9. ¿Quiénes son las personas encargadas de realizar el CI en la Facultad?
10. ¿Qué es adecuar la GA?
11. ¿Cómo se aplica la GA?
12. ¿Cuántas veces se aplica la GA en el año a cada una de las áreas?
13. ¿Qué se obtiene luego de aplicar la Guía de Autocontrol a un área?
14. ¿Qué es el Informe y qué contiene?
15. ¿Qué es el Plan de Medidas y qué contiene?
16. ¿Qué se realiza con esos documentos?
17. ¿Cómo se le da seguimiento al Plan de Medidas?
18. ¿Qué son las evidencias?
19. ¿Dónde se archivan esas evidencias?
20. ¿Qué es un expediente de CI?
21. ¿Qué usuarios serán definidos para el módulo CI?
22. ¿Qué colores se deben tener en cuenta para la interfaz visual del módulo?



## Anexo 7. Acta de aceptación del Módulo CI

## ACTA DE ACEPTACIÓN

En cumplimiento del **Convenio de colaboración en la Universidad de las Ciencias Informáticas** y en función de la ejecución del trabajo de diploma: Módulo Control Interno para el Sistema de Administración y Economía de la Facultad 3, se hace entrega de los productos que se relacionan a continuación:

- Expediente de proyecto generado según la metodología.
- Código de la aplicación y Backup de la Base de Datos.

La Parte Cliente, luego de haber revisado los productos de trabajo determina que se **sí se aceptan**.

Entrega	Recibe
<b>Nombres y apellidos:</b> Yeisel Pérez Rivera y Yanirosa Almenares Ramirez	<b>Nombre y apellidos:</b> MSc. Ana Marys Garcia Rodríguez
<b>Cargos:</b> Diplomantes	<b>Cargo:</b> Vicedecana de Economía y Administración de la Facultad 3
<b>Firmas:</b> 	<b>Firma:</b> 

### Observador independiente

**Nombre y Apellidos:** Ing. Idel Jorge Sánchez González

**Cargo:** Tutor

**Firma:** 

**Fecha:** 08/06/2015