

Universidad de las Ciencias Informáticas

Facultad 3



Desarrollo del módulo Económico para el  
Sistema de Administración y Economía de la Facultad 3

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

**Autoras:**

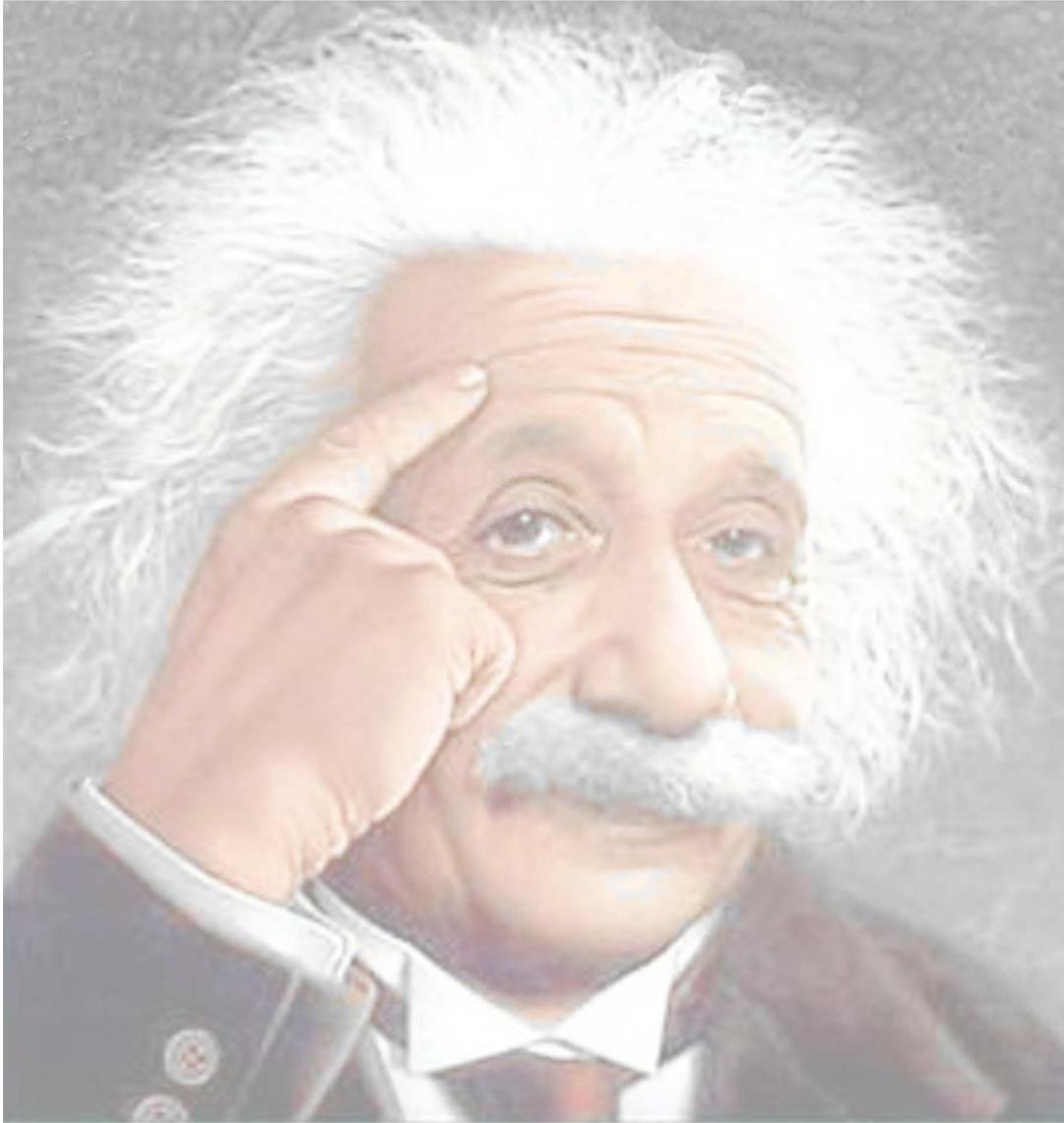
Dailét Manuela Soto Fumero

Elizabeth Guerra Pompa

**Tutora:**

MSc. Ana Marys Garcia Rodríguez

La Habana, Junio de 2015



*“Si buscas resultados distintos, no hagas siempre lo mismo”*

*A. Einstein*

## *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 \_\_\_\_\_.

Autoras:

\_\_\_\_\_  
Elizabeth Guerra Pompa

\_\_\_\_\_  
Dailét Manuela Soto Fumero

Tutora:

\_\_\_\_\_  
MSc. Ana Marys Garcia Rodríguez

**RESUMEN**

La Universidad de las Ciencias Informáticas es uno de los proyectos surgidos al calor de la Batalla de Ideas, que tiene como misión la formación de profesionales cualificados y altamente comprometidos con la Revolución, bajo un paradigma de formación que integra los procesos de formación, investigación, producción y extensión. En su estructura organizacional cuenta con siete facultades que a su vez disponen de un Vicedecanato de Economía y Administración. En la Facultad 3, dicho vicedecanato tiene entre sus funciones el control de los procesos asociados al área económica, los cuales se realizan de forma manual y generan gran cantidad de documentación, provocando que en ocasiones se cometan errores en las actividades de gestión y que se dificulte mantener el control y disponibilidad de la información ante las pérdidas, desactualización y deterioro de la documentación. Como solución a esta problemática, se desarrolló el módulo Económico del Sistema de Administración y Economía de la Facultad 3, bajo la metodología de desarrollo Proceso Unificado Ágil con la versión establecida por la UCI y siguiendo las políticas de soberanía tecnológica establecidas por la universidad. El funcionamiento de la herramienta informática desarrollada, fue verificado mediante la aplicación de pruebas de caja negra. Además, se realizó la validación de la investigación mediante las pruebas de aceptación y la aplicación de la técnica de ladov, donde se evidenció un alto nivel de satisfacción por parte del cliente.

### AGRADECIMIENTOS

*Son muchas las personas a las cuales quisiera agradecer, porque cada una de ellas me ha brindado enseñanzas que atesoraré por siempre. Empezaré por agradecer a la persona que más aprecio y cuyas palabras siempre dejan eco en mí, a mi mamá Maura Fumero Gómez, ella ha sido siempre mi impulso, el primer motivo por el cual decidí cursar estudios universitarios y la primera persona que me venía a la mente cada vez que me desesperaba a la mitad del camino o disminuían mis esperanzas, y la que nunca se cansó de darme ánimos.*

*Me gustaría agradecer a mi familia: mis tías María, Graciela, Merilyn, a mis tíos Jesús, Ismael y Vicente, todos y cada uno de ellos me han aportado grandes enseñanzas y profundas reflexiones que me han servido de gran ayuda en momentos en los que no he sabido qué hacer, y me brindaron apoyo cada vez que lo he necesitado. A mis primas Beatriz y Yamilka por sus consejos. A mi novio, Yairon Vargas Águila por toda su paciencia durante el desarrollo de la tesis, todos los ánimos que me daba cada vez que me sentía agotada, por creer siempre en mí y por toda su ayuda. A Isael Pichardo, que durante los 5 años de mi carrera me apoyó con todo aquello que estuviera a su alcance y pudiera facilitar mis estudios.*

*Mis agradecimientos también van para una persona que en su momento guio mi camino y dejó huella en mí: al profesor Rufino, quien impartía clases en mi secundaria básica “Romualdo de la Cuesta” y la persona que me enseñó a apreciar las matemáticas y a entenderlas.*

*Agradecer a mi tutora Ana Marys García Rodríguez que ha sido durante todo este proceso como una madre que me brindó su apoyo, su tiempo, sus conocimientos, su paciencia, su casa y su comida; gracias por guiarme y enseñarme todos los días un poquito más. Al profesor Idel Jorge Sánchez que me atendió siempre que tuve alguna interrogante con la mejor voluntad, sin importar cuán descabellada fuese la pregunta. A Roberto Jesús Rielo Villadoniga por su ayuda. Al profesor Adrián Naranjo García que nos brindó su tiempo y conocimientos; a las trabajadoras del Vicedecanato de Administración y Economía que me atendieron siempre muy amablemente y respondieron todas mis preguntas con la mayor profesionalidad y a los trabajadores del departamento de Nómina de la UCI.*

*Agradecer también a mis amistades que durante los cinco años de carrera han sido mi familia dentro de la universidad.*

*A todas estas personas van mis agradecimientos.*

**Dailét Manuela Soto Fumero**

## *Agradecimientos*

*A ti Mami y a Papi, les dedico éste día, éste momento, éstas palabras, porque gracias a ustedes me he convertido en la persona que soy. Fueron, son y siempre serán mi escuela, mi modelo a seguir, mi fuerza. Gracias por enseñarme a lograr mis metas en la vida, a superar los obstáculos sin importar qué, a sacudirme el polvo después de caer, levantarme y caminar con la cabeza erguida. Por su apoyo, esfuerzo y amor incondicional gracias.*

*A mi familia de la UCI, Javito, Yordan, Burry, Ever, Lorena, Dariel, Omar, Bárbaro por ayudarme a resistir todo este tiempo, soportar todas mis malacrianzas, preocuparse por mí, llenarme de confianza, estar siempre ahí cuando más me hicieron falta, compartir mi felicidad, secar mis lágrimas y ayudarme a cumplir con mis metas. Por todos los momentos que compartimos juntos, buenos y malos, y los que nos quedan por vivir.*

*A mis amigos en general, de la universidad, a los que vienen conmigo desde hace mucho; a los que están lejos y a los que ya no están, todos son una parte importante de mi vida y siempre los llevo presente.*

*A ti Ana, mi tutora, por enseñarme que no hay gloria sin sacrificios y que debemos tomar las riendas de nuestra vida para forjarnos un futuro. Por tu voto de confianza y ser más que tutora, ser una amiga. Al 3505, no fuimos el mejor grupo ni el más unido pero cada día era una nueva aventura con ustedes. A todos mis profesores, por contribuir a mi formación con sus conocimientos y consejos, especialmente a ti Rosa, eres una persona maravillosa y me alegro haber sido una de tus estudiantes.*

*A todas las personas que de una forma u otra han influido en mi vida, que me han hecho cambiar y crecer como persona. A los que están por venir y que también dejarán su huella.*

*Gracias a los que de alguna manera, no importa cómo, hicieron posible este momento.*

**Elizabeth Guerra Pompa**

**DEDICATORIA**

*Mis logros van dedicados a una persona que ha servido de luz y guía a mi vida, que ha batallado conmigo y me ha amado desde el primer día que supo de mi existencia, esa persona es mi madre, mi padre, mi apoyo, mi amiga y confidente, mi mamá Maura Fumero Gómez.*

*Mi mamá es mi heroína y mi mayor inspiración. Tener como madre a una mujer con su trayectoria y con sus logros, me hacen sentirme orgullosa de ser mujer y de poder decir al mundo que esa mujer grandiosa es mi mamá. Ella me enseñó que ser madre significa dar todo por tus hijos sin importar lo que haya que sacrificar, apoyarlos incluso cuando cometen errores y esperarlos con los brazos abiertos sin ningún rencor cuando se dan cuenta que se han equivocado, me enseñó que el amor más grande, más puro y más genuino que existe es el de una madre. Me enseñó que ser mujer no es una debilidad es un reto. Me enseñó que debo convertir los problemas en retos a vencer, a darme mi lugar como mujer y persona, a exigir el respeto que sea capaz de ganarme, a buscar siempre el conocimiento, a trazarme metas grandes y alcanzarlas un paso a la vez.*

*Mi mamá siempre me alentó a ser yo misma, a quererme por quien soy y a tomar decisiones que mi hicieran sentir bien conmigo misma.*

*Si no fuera por mi mamá, realmente no sé si habría podido llegar a donde estoy. Verla superarse constantemente me sirvió de impulso para hacer lo mismo. Desde niña he imitado las cosas que mi mamá hacía, con pequeñas diferencias pero con la misma esencia, desde ponerme a garabatear papeles como si estuviera escribiendo lo mismo que ella escribía, o ponerme sus tacones y pintarme con sus maquillajes, o practicar karate al lado de ella, hasta el día de hoy que estoy culminando mis estudios universitarios al igual que lo hizo ella en su momento.*

*Es por esto que los logros que he alcanzado están dedicados a mi mamá. Mamita querida, gracias por estar siempre a mi lado, por nunca rendirte conmigo y por ayudarme a crecer hasta ser quien soy. Todos los días pido que me permitan seguir disfrutando de tu presencia, de tus consejos, de tus regaños, de tu calor, de tus abrazos, de tus besos, de tu risa, de tus bailes y de tu voz por muchos años más y que mis hijos también puedan aprender de ti. Te quiero mami, gracias por ser quien eres y por siempre creer en mí. Tu hijita querida te ama.*

**Dailét Manuela Soto Fumero**

*A mis padres, este título es más de ustedes que mío por mi formación y además por haber cargado conmigo todos estos años, así como a mí hermano.*

*A mi familia UCI, mis nakamas, ustedes dieron vida a estos años*

**Elizabeth Guerra Pompa**

**ÍNDICE**

|  |    |
|--|----|
| INTRODUCCIÓN .....   | 1  |
| CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA .....                        | 5  |
| 1.1 Introducción .....   | 5  |
| 1.2 Conceptos asociados al dominio del problema .....          | 5  |
| 1.3 Análisis de soluciones existentes .....                    | 9  |
| 1.3.1 Sorolla .....  | 9  |
| 1.3.2 Sistema Integral de Gestión Económica (SIGE) .....       | 10 |
| 1.3.3 SICAP .....  | 12 |
| 1.4 Proceso de desarrollo de software .....                    | 14 |
| 1.4.1 Metodología de desarrollo.....                           | 14 |
| 1.4.2 Ingeniería de requisitos.....                            | 16 |
| 1.4.3 Patrones arquitectónicos.....                            | 16 |
| 1.4.4 Patrones de diseño .....                                 | 17 |
| 1.4.5 Estándares de codificación .....                         | 18 |
| 1.4.6 Calidad de software .....                                | 19 |
| 1.5 Herramientas y tecnologías a utilizar.....                 | 26 |
| 1.5.1 Aplicación web .....                                     | 26 |
| 1.5.1 Servidor web .....                                       | 27 |
| 1.5.2 Lenguajes de programación.....                           | 27 |
| 1.5.3 Entorno de Desarrollo Integrado (IDE).....               | 29 |
| 1.5.4 SGBD .....   | 29 |
| 1.5.5 Marco de Trabajo ( <i>Framework</i> ).....               | 30 |
| 1.5.6 <i>Computer Aided Software Engineering</i> (CASE).....   | 32 |
| 1.5.7 Lenguaje Unificado de Modelado (UML).....                | 33 |
| 1.6 Conclusiones parciales.....                                | 33 |
| CAPÍTULO 2 CARACTERÍSTICAS DEL MÓDULO ECONÓMICO DEL SAEF3..... | 34 |
| 2.1 Introducción .....   | 34 |
| 2.2 Diagrama de procesos del negocio .....                     | 34 |

|  |    |
|--|----|
| 2.3 Descripción de la solución .....                         | 35 |
| 2.3.1 Requisitos funcionales .....                           | 37 |
| 2.3.2 Requisitos no funcionales .....                        | 40 |
| 2.3.4 Prototipos de interfaz de usuario .....                | 41 |
| 2.3.5 Validaciones de requisitos .....                       | 43 |
| 2.4 Diseño de la solución.....                               | 43 |
| 2.4.1 Patrones de diseño .....                               | 43 |
| 2.4.2 Modelo de datos.....                                   | 48 |
| 2.4.3 Verificación del diseño .....                          | 49 |
| 2.5 Conclusiones parciales.....                              | 53 |
| CAPÍTULO 3 IMPLEMENTACIÓN Y ANÁLISIS DE LOS RESULTADOS ..... | 54 |
| 3.1 Introducción .....                                       | 54 |
| 3.2 Aspectos relevantes de la implementación .....           | 54 |
| 3.2.1 Diagrama de componentes .....                          | 54 |
| 3.2.2 Estándares de codificación .....                       | 54 |
| 3.2.3 Tratamiento de errores.....                            | 56 |
| 3.3 Validación de la solución .....                          | 57 |
| 3.3.1 Pruebas de caja negra .....                            | 57 |
| 3.3.2 Tipos de pruebas .....                                 | 59 |
| 3.4 Validación de las variables de la investigación.....     | 59 |
| 3.4 Conclusiones parciales.....                              | 62 |
| CONCLUSIONES GENERALES.....                                  | 64 |
| RECOMENDACIONES .....  | 65 |
| REFERENCIAS.....   | 66 |
| ANEXOS .....   | 69 |
| Anexo #1. Diagramas de procesos de negocio .....             | 69 |
| Anexo #2. Prototipos de interfaz de usuario.....             | 73 |
| Anexo #3. Resultado de aplicación de la métrica TC .....     | 82 |

**ÍNDICE DE TABLAS**

|   |    |
|---|----|
| Tabla 1. Comparación entre los sistemas .....                           | 13 |
| Tabla 2. Procesos del VDEA que ejecutan los sistemas .....              | 14 |
| Tabla 3. Métricas para la validación del diseño .....                   | 22 |
| Tabla 4. Cuadro lógico de ladov. ....                                   | 25 |
| Tabla 5. Índice de satisfacción de ladov .....                          | 26 |
| Tabla 6. Personal involucrado al módulo Económico. ....                 | 36 |
| Tabla 7. Aplicación de las métricas de validación de requisitos .....   | 43 |
| Tabla 8. Resultado de la aplicación de las métricas del diseño.....     | 50 |
| Tabla 9. Resultado de la aplicación de las pruebas unitarias .....      | 57 |
| Tabla 10. Validación de las variables de la investigación .....         | 60 |
| Tabla 11. Cuadro lógico de ladov. ....                                  | 61 |
| Tabla 12. Resultado de aplicación de la técnica ladov .....             | 61 |
| Tabla 13. Clases del modelo a las que se les aplicó la métrica TC ..... | 82 |

**ÍNDICE DE FIGURAS**

Figura 1: Sistema de Gestión..... 6

Figura 2. Esquema de gestión ..... 7

Figura 3: Ciclo de vida de la Metodología AUP-versión UCI..... 15

Figura 4: Modelo Vista Controlador..... 17

Figura 5. Asignación de recursos materiales. .... 34

Figura 6. Prototipo de interfaz de usuario “Asignar recursos materiales” 1. .... 41

Figura 7. Prototipo de interfaz de usuario “Asignar recursos materiales” 2. .... 41

Figura 8. Prototipo de interfaz de usuario “Adicionar Vale de salida” ..... 42

Figura 9. Prototipo de interfaz de usuario “Adicionar Bonificación de pasaje”..... 42

Figura 10. Diagrama de clases del diseño de Bonificación de pasajes a estudiantes ..... 44

Figura 11. Clase *baseec.html.twig* ..... 45

Figura 12. Clase *layout.html.twig* ..... 45

Figura 13. Clase *EReintegroPasaje/index.html.twig* ..... 46

Figura 14. Clase *EReintegroPasaje/new.html.twig* ..... 47

Figura 15. Ejemplo de aplicación del patrón de inicialización vaga ..... 48

Figura 16. Comportamiento del indicador responsabilidad ..... 50

Figura 17. Comportamiento del indicador complejidad ..... 51

Figura 18. Comportamiento del indicador reutilización ..... 51

Figura 19. Comportamiento del indicador acoplamiento ..... 51

Figura 20. Comportamiento del indicador complejidad de mantenimiento ..... 52

Figura 21. Comportamiento del indicador reutilización ..... 52

Figura 22. Comportamiento del indicador cantidad de pruebas..... 52

Figura 23. Diagrama de componentes del módulo Económico del SAEF3 ..... 54

Figura 24. *Screenshot* al código de *EReintegroPasajeController.php* ..... 55

Figura 25. *Screenshot* al código de la interfaz de modificar *EReintegroPasaje* ..... 55

Figura 26. *Screenshot* a la controladora de *EReintegroPasaje* ..... 56

Figura 27. Control del presupuesto. .... 69

Figura 28. Control de préstamos y ayudas económicas. .... 70

Figura 29. Control de bonificación de pasajes a estudiantes..... 71

Figura 30. Solicitud de recurso material..... 72

Figura 31. Prototipo de interfaz de usuario "Solicitar recursos materiales"..... 73

Figura 32. Prototipo de interfaz de usuario "Modificar solicitud de recursos materiales" ..... 73

Figura 33. Prototipo de interfaz de usuario "Consultar solicitud de recursos materiales" ..... 74

Figura 34. Prototipo de interfaz de usuario "Consultar vale de salida" ..... 75

## *Índice de figuras*

|   |    |
|---|----|
| Figura 35. Prototipo de interfaz de usuario "Modificar vale de salida" .....                        | 75 |
| Figura 36. Prototipo de interfaz de usuario "Consultar bonificación de pasajes a estudiantes" ..... | 76 |
| Figura 37. Prototipo de interfaz de usuario "Modificar bonificación de pasajes a estudiantes" ..... | 76 |
| Figura 38. Prototipo de interfaz de usuario "Adicionar préstamo o ayuda económica" .....            | 78 |
| Figura 39. Prototipo de interfaz de usuario "Consultar préstamo o ayuda económica" .....            | 80 |
| Figura 40. Prototipo de interfaz de usuario "Modificar préstamo o ayuda económica" .....            | 81 |

## **INTRODUCCIÓN**

La Economía es el estudio de cómo las sociedades utilizan recursos escasos para producir bienes valiosos y distribuirlos (Méndez Morales, 2009). Esta esfera es el eslabón fundamental para el desarrollo sostenible de las naciones y está enmarcada dentro del grupo de ciencias sociales que se dedica al estudio de los procedimientos productivos y de intercambio, y al análisis del consumo de bienes y servicios. Las organizaciones bajo una perspectiva económica, combinan distintos factores con el ánimo de alcanzar los objetivos definidos (Bueno Campos, y otros, 1979).

A pesar de que la Economía establece cómo deben utilizarse los recursos limitados para la satisfacción de las necesidades, debe combinarse con conocimientos administrativos que posibiliten su gestión eficaz. La administración se enfoca en un conjunto de actividades encaminadas a aprovechar los recursos de manera eficiente, con el propósito de alcanzar los objetivos o metas de la organización. Para desplegar una visión integral sobre los fines, estrategias y fundamentar el proceso de toma de decisiones, se basa en reglas sistemáticas y en la coordinación del trabajo y las personas mediante el análisis de la información disponible (Da Silva, 2002).

La esencia de la gestión administrativa es la toma de decisiones, la cual incluye la formulación, evaluación y selección de opciones para resolver problemas administrativos. La calidad de las decisiones está íntimamente ligada a la disponibilidad de información útil en el momento en que se toma la decisión (Davis, 2001). Para viabilizar esta accesibilidad se ha promovido el desarrollo de sistemas informáticos bajo los nuevos paradigmas de las Tecnologías de la Información y las Comunicaciones (TIC), las cuales han revolucionado su desarrollo y proceder en el mundo actual.

En diversas universidades a escala mundial, la gestión económica es el área encargada de los procedimientos establecidos para la administración de los procesos de la actividad económico-financiera, mediante el registro del presupuesto, ingresos y gastos. Dichos procesos comprenden además la elaboración del anteproyecto y liquidación anual del mismo, la tutela de la contabilidad y de la cuenta, la gestión de la tesorería, la elaboración de los estados contables, de la información económica y de las declaraciones fiscales de la universidad (UCO, 2013).

La Universidad de las Ciencias Informáticas (UCI), institución que surge en un profundo y novedoso proceso de transformaciones educacionales y sociales en Cuba, fue concebida como un centro de formación con una destacada actividad política y cultural de sus educandos, mediante la integración de la docencia, la investigación y la producción. Dentro de la estructura organizativa de las facultades que la integran, se encuentra el Vicedecanato de Economía y Administración (VDEA), el cual dirige y controla los procesos asociados al área económica de la facultad, referentes al control de:

bonificaciones de pasajes a estudiantes, préstamos estudiantiles y ayudas económicas, planificación y ejecución del presupuesto, solicitud de recursos materiales y asignación de recursos materiales.

Actualmente, en el VDEA de la Facultad 3 los procesos generan gran cantidad de documentación y se realizan manualmente; como consecuencia han existido errores durante la ejecución de los mismos al contar con información desactualizada. Se han identificado pérdidas y deterioro de la documentación, lo cual afecta el desarrollo de las actividades cotidianas del vicedecanato y la entrega oportuna de reportes solicitados tanto por la Decana como por la Vicerrectoría Económica. Por otra parte, todos los documentos referentes al área económica, se encuentran centralizados en el VDEA, por lo que se dificulta el acceso y control sobre la misma desde las áreas de la facultad. Además, el manejo del gran cúmulo de datos disponible solo en formato duro, obstaculiza el control y disponibilidad de la información y entorpece la actuación proactiva que debe caracterizar al VDEA.

De la problemática anteriormente descrita se deriva el **problema de la investigación**: ¿cómo gestionar la información asociada a los procesos del área económica de la Facultad 3 de manera que contribuya a elevar la disponibilidad y control de la información?

Se identifica como **objeto de estudio**: la informatización de los procesos asociados a la gestión económica y enmarcado en el **campo de acción**: la informatización de los procesos asociados al área económica de la Facultad 3.

Para dar solución al problema planteado se define el **objetivo general**: desarrollar el módulo Económico para el Sistema de Administración y Economía de la Facultad 3, de manera que eleve la disponibilidad y control de la información en los procesos asociados al área económica de la Facultad 3. El objetivo general se desglosa en los siguientes **objetivos específicos**:

1. Definir el marco teórico de la investigación mediante el estudio y el análisis de los principales referentes teóricos para el desarrollo de la solución.
2. Identificar y describir los requisitos del sistema a implementar para el área económica de la Facultad 3.
3. Diseñar la propuesta de solución de los requisitos para los procesos asociados al área económica de la Facultad 3.
4. Implementar la solución para obtener los componentes de software de los procesos asociados al área económica de la Facultad 3.
5. Verificar la solución propuesta mediante la realización de pruebas de caja negra.
6. Validar las variables de la investigación.

Con el propósito de dar cumplimiento al objetivo propuesto, se plantean las siguientes **tareas de la investigación**:

1. Análisis de los referentes teóricos para identificar los elementos que constituyen aportes a la solución.
2. Estudio del estado de arte de sistemas informáticos afines a la problemática para valorar posibles contribuciones a la solución.
3. Modelado de los procesos asociados al área económica de la Facultad 3 para la comprensión del negocio a informatizar.
4. Identificación y descripción de las funcionalidades del módulo para su posterior desarrollo.
5. Diseño e implementación de la solución propuesta para la obtención de un producto que satisfaga las necesidades del Vicedecanato de Economía y Administración.
6. Diseño y aplicación de casos de prueba con la finalidad de verificar el correcto funcionamiento del módulo implementado.
7. Validación de las variables de la investigación para corroborar el cumplimiento del objetivo general.

Para dar cumplimiento a las tareas especificadas se aplicaron los métodos de investigación:

### Teóricos:

- **Analítico-Sintético:** proporcionó la realización de un estudio teórico de la investigación facilitando el análisis y la comprensión de la documentación consultada contribuyendo a realizar comparaciones entre los diferentes sistemas, extraer y precisar los elementos y características fundamentales del diseño de la propuesta de solución y establecer las conclusiones de la investigación.
- **Histórico-Lógico:** se utiliza para el trabajo recopilatorio sobre el desarrollo de sistemas para la gestión económica.

### Empíricos:

- **Observación:** se emplea para conocer el estado actual del entorno sin necesidad de apoyarse en terceros o intermediarios, obteniendo la información a partir de la percepción propia. Se realiza en distintos momentos de la investigación como método de esclarecimiento.
- **Entrevista:** se emplea la entrevista no estructurada o libre, que se realiza con preguntas abiertas sin un orden preestablecido, adquiriendo características de conversación. Consiste en realizar preguntas de acuerdo a las respuestas. Se utilizó para obtener conocimiento del negocio y su funcionamiento, así como la manera de proceder de los trabajadores del VDEA.

## **Cualitativos y cuantitativos:**

- Los **modelos de análisis cuantitativos** para el procesamiento de los datos y análisis de las encuestas.

## **Matemáticos:**

- La **estadística descriptiva** para el análisis de los resultados.

Se empleó además la **Técnica de ladov** para conocer el nivel de satisfacción del cliente con el módulo propuesto.

## **Posibles resultados:**

Módulo Económico para el Sistema de Administración y Economía de la Facultad 3 (SAEF3).

El trabajo está estructurado en tres capítulos, descritos a continuación:

### **Capítulo 1: Fundamentación teórica**

Se realiza un análisis de los referentes teóricos entorno al problema identificado. Se hace referencia a las tendencias actuales, así como las tecnologías, herramientas y metodología aplicables al desarrollo de la solución.

### **Capítulo 2: Características del módulo Económico del SAEF3**

Se realiza el modelado de los procesos del negocio y posteriormente se definen los requisitos funcionales y no funcionales para el establecimiento de las características y cualidades que debe cumplir el módulo. Se muestran además los resultados de la aplicación de las métricas para la validación de los requisitos. Se describe el diseño de la solución y se especifican las características y patrones de diseño utilizados. Para verificar que el diseño cumpla con los atributos de calidad, se aplican métricas del diseño que corroboren el correcto uso de los patrones.

### **Capítulo 3: Implementación y análisis de los resultados**

Se describen los estándares de codificación empleados durante la implementación, para propiciar legibilidad, claridad, mantenibilidad y una mayor comprensión del código por parte de los desarrolladores. Se efectúa el tratamiento de errores para brindar información oportuna ante eventualidades. Se especifica el nivel de prueba aplicado para la verificación del módulo desarrollado, así como los tipos y métodos de pruebas. Se muestran además los resultados obtenidos durante la ejecución de los casos de pruebas diseñados.

## CAPÍTULO 1 FUNDAMENTACIÓN TEÓRICA

### 1.1 Introducción

En el presente capítulo se define el marco teórico referencial de la investigación. Se realiza un análisis alrededor de los conceptos asociados al dominio del problema y se estudia el estado de arte de los procesos y sistemas de gestión económica existentes en la actualidad. Se describen además, las tecnologías y herramientas, así como el lenguaje de programación a utilizar en el desarrollo del módulo Económico del SAEF3.

### 1.2 Conceptos asociados al dominio del problema

A continuación se presentan los conceptos asociados al dominio del problema.

#### **Economía**

La Economía es una ciencia social que estudia cómo los individuos o las sociedades usan o manejan los recursos para satisfacer sus necesidades. Estos recursos pueden ser distribuidos entre la producción de bienes y servicios, y el consumo, ya sea presente o futuro, de diferentes personas o grupos de personas en la sociedad (Subgerencia Cultural Banco de la República, 2014).

A pesar de que la Economía establece cómo deben utilizarse los recursos limitados para la satisfacción de las necesidades, debe combinarse con conocimientos administrativos.

#### **Administración**

Varios autores plantean que la administración es el proceso de coordinar las actividades de trabajo mediante la planeación, organización, dirección y control de los recursos para el logro de los objetivos (Chiavenato , 2004) (Robbins , y otros, 2005) (Koontz, y otros, 2004) (Oliveira Da Silva , 2002). Todos coinciden en que el aprovechamiento de los recursos debe realizarse de manera eficiente.

En el contexto de la investigación se concibe la administración como la coordinación de un conjunto de actividades para el logro de los objetivos de la organización mediante una correcta planeación y control de los recursos, a fin de garantizar la eficiencia de los procesos, donde tienen un papel primordial los sistemas de gestión para el apoyo en la toma de decisiones.

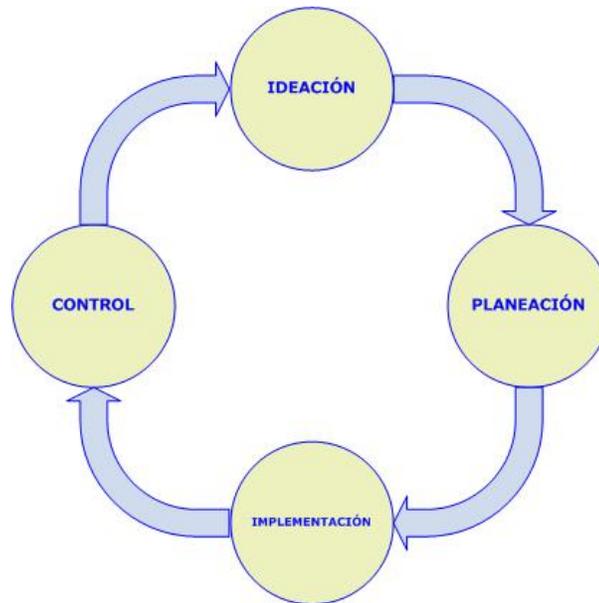
#### **Sistema de gestión**

Los sistemas de gestión son estructuras probadas para la gestión y mejora continua de las políticas, procedimientos y procesos de la organización. La tendencia actual de las medianas y grandes empresas es ir integrando dichos sistemas a medida que van añadiéndolos a su organización.

Un sistema de gestión ayuda a lograr los objetivos de una organización mediante una serie de estrategias que incluyen la optimización de procesos, el enfoque centrado en la gestión de la

## Capítulo 1 Fundamentación teórica

información, entre otros (The British Standards Institution, 2013). Además, es considerado como un conjunto de etapas unidas en un proceso continuo, que permite trabajar ordenadamente una idea hasta lograr mejoras y su continuidad. Vergara (Vergara, Gonzalo, 2009) establece en su definición, un desglose más objetivo al definir las cuatro etapas del sistema de gestión.



**Figura 1: Sistema de Gestión**

Fuente: (Vergara, Gonzalo, 2009)

Las cuatro etapas del sistema de gestión son: ideación, planeación, implementación y control.

### Etapa de ideación

El objetivo de esta etapa es trabajar en la idea que guiará los primeros pasos del proceso de creación que se logra con el sistema de gestión propuesto. Existen varias metodologías para lograr refinar la idea, sin embargo, se recomienda una muy práctica conocida como lluvia de ideas o *brainstorming* (Vergara, Gonzalo, 2009).

### Etapa de planeación

Dentro del proceso, la planificación constituye una etapa fundamental y el punto de partida de la acción directiva, la misma supone el establecimiento de sub-objetivos y los cursos de acción para alcanzarlos.

En esta etapa, se definen las estrategias que se utilizarán, la estructura organizacional que se requiere, el personal que se asigna, el tipo de tecnología que se necesita, el tipo de recursos que se utilizan y la clase de controles que se aplican en todo el proceso. Si bien es cierto que el proceso de planificación depende de las características particulares de cada organización, dentro de cualquier proceso formal

de planificación, existen tres perspectivas básicas comunes: la estrategia corporativa, de negocios y funcional.

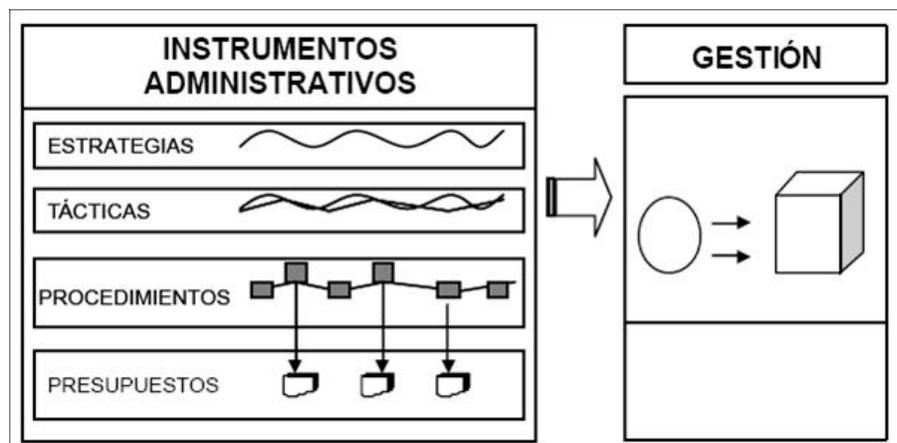
El proceso de planificación contiene un número determinado de etapas que hacen de ella una actividad dinámica, flexible y continua. En general, estas etapas consideran, para cada una de las perspectivas mencionadas, el examen del medio externo (identificación de oportunidades y amenazas), la evaluación interna (determinación de fortalezas y debilidades), y concluye con la definición de una postura competitiva sugerida (objetivos y metas).

A nivel corporativo, se obtienen como resultado las directrices estratégicas y los objetivos de desempeño de la organización. Además, se determina la asignación de recursos, la estructura de la organización (que se necesita para poner en práctica exitosamente la estrategia definida), los sistemas administrativos y las directrices para la selección y promoción del personal clave.

A nivel de negocios y funcional, los resultados se enmarcan en propuestas de programas estratégicos de acción y programación de presupuestos. Estas propuestas son, finalmente, evaluadas y consolidadas a nivel corporativo (Vergara, Gonzalo, 2009).

### Etapa de implementación (gestión)

En su significado más general, se entiende por gestión, la acción y efecto de administrar. Pero, en un contexto empresarial, esto se refiere a la dirección que toman las decisiones y las acciones para alcanzar los objetivos trazados. Es importante destacar que las decisiones y acciones que se toman para llevar adelante un propósito, se sustentan en los mecanismos o instrumentos administrativos (estrategias, tácticas, procedimientos, presupuestos, entre otros), que están sistémicamente relacionados y que se obtienen del proceso de planificación (Ver Figura 2.)



**Figura 2. Esquema de gestión**

Fuente (Vergara, Gonzalo, 2009).

### Etapa de control

El control es una función administrativa, esencialmente reguladora, que permite verificar, si el elemento seleccionado, está cumpliendo sus objetivos o alcanzando los resultados que se esperan.

Es importante destacar que la finalidad del control es la detección de errores, fallas o diferencias, en relación a un planteamiento inicial, para su corrección y/o prevención. Por tanto, el control debe estar relacionado con los objetivos inicialmente definidos, debe permitir la medición y cuantificación de los resultados, la detección de desviaciones y el establecimiento de medidas correctivas y preventivas (Vergara, Gonzalo, 2009).

Las cuatro etapas mencionadas previamente permiten visualizar los sistemas de gestión como un proceso virtuoso y continuo, pues en la medida que el ciclo se repite de manera recurrente y recursiva, se logra obtener una mejora aplicable a la gestión económica de cualquier entidad, donde no quedan exentas las universidades.

### **Gestión económica en las universidades**

En las instituciones de educación superior constituye un eslabón fundamental la gestión económica, que es el área encargada de los procedimientos establecidos para la administración de los procesos de la actividad económico-financiera, mediante el registro del presupuesto, ingresos y gastos (PAe, 2010) (UCO, 2014) (SIUV, 2015). Estos procesos se realizan de manera distinta en cada una de las universidades, atendiendo a las particularidades de las mismas, pero en su esencia comprenden además la elaboración del anteproyecto y liquidación anual del mismo, la tutela de la contabilidad y de la cuenta, la gestión de la tesorería, la elaboración de los estados contables, de la información económica y de las declaraciones fiscales de la universidad (UCO, 2013).

La UCI fue concebida como un centro de formación con una destacada actividad política y cultural de sus educandos, mediante la integración de la docencia, la investigación y la producción. Dentro de la estructura organizativa de las facultades que la integran, se encuentra el Vicedecanato de Economía y Administración (VDEA), el cual dirige y controla los procesos asociados al área económica de la facultad, referentes al control de: bonificaciones de pasajes a estudiantes, préstamos estudiantiles y ayudas económicas, planificación y ejecución del presupuesto, solicitud de recursos materiales y asignación de recursos materiales. Actualmente, en el VDEA de la Facultad 3, los procesos se realizan manualmente y generan gran cantidad de documentación, lo cual obstaculiza el control y disponibilidad de la información y entorpece la actuación proactiva que debe caracterizar al VDEA. Lo anteriormente planteado constituyó la motivación de la presente investigación para desarrollar un módulo integrado al SAEF3 que facilite la gestión de los procesos asociados a dicha área económica. Como paso inicial

se realizó un estudio de las soluciones informáticas que facilitan el control económico en las universidades, con el objetivo de valorar sus posibles aplicaciones al dominio en cuestión.

### 1.3 Análisis de soluciones existentes

En el presente epígrafe se analizan soluciones homólogas existentes principalmente a nivel internacional, pues en el ámbito nacional no existen sistemas informáticos que permitan gestionar los procesos asociados a la gestión económica de las facultades que se subordinan a las universidades.

#### 1.3.1 Sorolla

El sistema de apoyo a la gestión económico-presupuestaria de las entidades públicas administrativas españolas SOROLLA, fue creado en 1996. Presenta una arquitectura modular, a la vez que integrada, que permite a cada instalación el empleo de los módulos de su interés, sin otra limitación que la establecida por la propia estructura y lógica del sistema (PAe, 2010). Las herramientas de gestión contempladas en el proyecto son:

- Un sistema para la tramitación de expedientes de gasto que se realiza a través del módulo DocumentA y que alcanza la doble vertiente de la tramitación: administrativa y contable. Las últimas modificaciones funcionales aplicadas a este módulo permiten, además, aplicar la firma electrónica a los documentos generados en la tramitación, la generación de expedientes electrónicos y el envío/recepción por vía electrónica de dichos expedientes a /de la Intervención delegada correspondiente.
- Un sistema de gestión de justificantes de gasto que se instrumenta a través del módulo denominado Justificantes del Gasto que permite la gestión de las cajas pagadoras en lo referente a los pagos que se realizan mediante el sistema de "A justificar", tanto por Acuerdo de Caja Fija como por Pagos a Justificar.
- Un sistema para la elaboración de documentos contables a través del módulo Docuconta integrado con los módulos de DocumentA y Justificantes del Gasto, que contempla un modelo de comunicación automática con un Sistema de Información Contable (SIC).
- Un sistema de gestión del inmovilizado material instrumentado a través del módulo de Gestión de Inventario que permite el mantenimiento actualizado de dicho inventario, tanto desde el punto de vista de los detalles de su composición como en su faceta contable.
- Además de la gestión de comisiones de servicio incorporada al módulo de Justificantes del Gasto, dispone de un módulo independiente SOROnet, desarrollado bajo tecnología web que permite a los propios comisionados introducir en el sistema los datos de la comisión, solicitar adelanto si procede, obtener la orden de comisión de servicios y generar la correspondiente liquidación con carácter provisional.

Junto a estas herramientas de gestión, se contemplan otras para el tratamiento de la información:

- Un módulo de elaboración de informes de ejecución presupuestaria, en base a los datos tratados en los sistemas de gestión, aún cuando estos datos todavía no hayan dado lugar a la actualización correspondiente en el SIC.
- Un módulo para la elaboración de la documentación de carácter tributario relativa a los pagos realizados a personas físicas o jurídicas y otro correspondiente al resumen anual de retenciones e ingresos.
- Un módulo generador de informes personalizados a través de la vinculación de vistas de la base de datos SOROLLA a una base de datos de tipo ACCESS97/ACCESS2000/ACCESS2003.

### Descripción Técnica

Desarrollado bajo arquitectura cliente-servidor con productos de Microsoft. En la actualidad hay dos formas de instalar SOROLLA dentro de una red:

- Mediante una instalación cliente-servidor, donde en cada puesto cliente se instalan y ejecutan los programas que componen el aplicativo.
- Utilizando el servicio Terminal Server de Microsoft, en el que los programas se instalan en un servidor de aplicaciones o incluso en el servidor de datos y en cada puesto cliente se instala la parte cliente de este servicio.

Se recomienda la primera opción cuando el número de puestos clientes no es muy elevado (entre dos y ocho) y están dentro de una red LAN. Por el contrario, se recomienda la segunda opción, cuando el número de puestos clientes es superior a ocho o están en el contexto de redes WAN.

En lo relativo a las características técnicas de los programas y bases de datos empleados en SOROLLA, cabe indicar que la línea estratégica establecida pretende, en la medida de lo posible, ser independiente de costes de licencias de software y vincular el proyecto a un entorno tecnológico adecuado con una proyección de futuro asegurada.

### 1.3.2 Sistema Integral de Gestión Económica (SIGE)

Desarrollado en 1998 por el personal de servicio de informática de la Universidad de Córdoba – España, junto con los gestores contables de la misma, se adapta perfectamente a las necesidades existentes en esta universidad, a la vez que permite una adaptación a los requerimientos que van surgiendo a lo largo del tiempo, estando el equipo de desarrollo en constante actividad, poniendo a disposición de los usuarios contables una aplicación a la medida. Así mismo, permite un soporte personalizado por parte de personal de la propia universidad.

## Capítulo 1 Fundamentación teórica

Al ser una herramienta propia permite una interrelación con los diferentes programas de gestión de esta universidad, evitando redundancias entre las diferentes aplicaciones, permitiendo el trasvase de información y posibilitando la obtención de información para combinar los diferentes elementos que forman parte de la Gestión Universitaria (Recursos Humanos, Gestión de Alumnos, Gestión de I+D, entre otros) (UCO, 2014).

Funcionalidades que ofrece:

- **Pagos:** contempla la introducción de justificantes del gasto (facturas), permitiendo una reserva de crédito en el presupuesto implicado desde el momento de la grabación de la misma en el caso de las facturas en firme. Se pueden grabar también los justificantes de los pagos a justificar. Se gestionan las operaciones en divisas y la generación de los documentos contables de pago de justificantes.
- **Ingresos:** se recogen tanto las liquidaciones como la recaudación de los ingresos permitiendo un control de los ingresos pendientes de aplicar. Se generan los documentos contables correspondientes y permite el control de la evolución de las previsiones de ingresos en el ejercicio.
- **Gestión del presupuesto:** introducción y actualización del presupuesto inicial de la universidad, modificación del presupuesto por incorporaciones y anulaciones, gestión de expedientes presupuestarios, repartos presupuestarios, certificaciones de las diferentes ayudas recibidas, obtención de información presupuestaria (mayores, balances, diarios, entre otros).
- **Tesorería:** gestión de los anticipos de caja y transferencias del banco en casa a las unidades de gasto. Consolidación bancaria, transferencias bancarias, emisión de cheques y de pagarés, ordenación del pago y pago de documentos contables. Gestión de pagos por *confirming*.
- **Auditoría y control interno:** firmas autorizadas, fiscalización, control y verificación de todas las operaciones realizadas en el sistema.
- **Contratación:** gestiona los expedientes de contratación que se realizan y todas las fases implicadas en los mismos (solicitudes, ofertas, adjudicaciones, concursos, fianzas, contratos, actas, entre otros). Permite generar a partir de las certificaciones, los documentos de pago correspondientes, así como las autorizaciones, disposiciones y obligaciones, contabilizándose y apuntándose directamente en la contabilidad.
- **Inventario:** mantenimiento de todos los bienes de la universidad, traspasos y traslados de bienes, consolidación de inventario, incorporación directa de los justificantes de compra con las hojas de inventario, gestión de suministros.
- **Cierre del ejercicio económico:** generación de movimientos, cuentas e informes de cierre.

- **Informes y estadísticas:** SIGE incluye todos los listados necesarios para una correcta gestión de la contabilidad Universitaria, así como todas las declaraciones, formatos bancarios y libros solicitados por organismos y empresas ajenos a la universidad.

### 1.3.3 SICAP

Desarrollada en el año 200 por la empresa TAO y adquirida por la Universidad de Valencia en el año 2000 con el fin de mejorar su gestión económica y sus sistemas de información (SIUV, 2015). Pretende cumplir entre otros, los siguientes objetivos:

- Mejorar el control de la gestión.
- Aumentar la descentralización de la gestión a nivel de centros y departamentos.
- Establecer un mayor grado de automatización y modernización en la gestión.
- Optimizar los circuitos de información.

### Características especiales

- Base de datos única. La gestión del presupuesto y la contabilidad general son dos "visiones" de la misma información que es única, evitando incoherencias.
- Toda la información bajo el modelo relacional. Ello permite explotarla desde herramientas informáticas como hojas de cálculo o de ayuda a la decisión.
- Definición libre de "Asientos-Patrón" incluida su lógica de tramitación.
- Permite en cada instalación adaptar la normativa contable para que haga cómoda su gestión. Hay adaptaciones a todas las Administraciones.
- Permite tanto la gestión centralizada como la gestión desconcentrada o la gestión descentralizada.

### Características funcionales

- Aplicación modular: dota al sistema contable de las herramientas de control de que carece la contabilidad por partida doble y permite ajustarlas a los requisitos de la Instrucción y ahorrar al usuario la redundancia de los asientos múltiples mediante la tipificación de las operaciones que pueden realizarse desde las distintas contabilidades auxiliares.
- Multi-institución con proceso de consolidación contable entre instituciones.
- Integración de datos externos y carga de datos desde otras aplicaciones: nómina, recaudación, entre otros.
- Permite trabajar con un nuevo ejercicio sin haber cerrado el anterior.
- Seguimiento de la tramitación presupuestaria de gastos e ingresos.
- Documentación de libre definición asociada a cada posible operación contable.
- Salidas de listados a formatos Access o Excel, salidas gráficas, simulaciones.

- Elaboración automática de la memoria.
- Tratamiento de "conjuntos de operaciones".
- Pre-introducción de terceros.
- Multi-aplicación y registro de facturas.
- Multi-institución, varias instituciones en un sistema.

**Características Técnicas**

- Servidor de base de datos
- Sistema operativo: AIX
- Gestor de base de datos: ORACLE
- Servidor de terminal Server (10 máquinas)
- Sistema operativo: WINDOWS 2000 ADVANCED SERVER
- Aplicación de contabilidad SICAP + ACCES 97
- Computadoras de usuarios (alrededor de 250)
- Software mínimo: cliente de Terminal Server

A continuación, se muestran las tablas comparativas correspondientes a determinados indicadores asociados a la soberanía tecnológica por la cual abogan Cuba y la UCI, teniendo también en consideración los procesos que se realizan en el VDEA (Tablas 1 y 2).

| Sistemas | Exento de pago | Open Source | Multiplataforma |
|----------|----------------|-------------|-----------------|
| SICAP    | -              | -           | -               |
| SIGE     | -              | -           | ✓               |
| SOROLLA  | ✓              | -           | ✓               |

Tabla 1. Comparación entre los sistemas

Fuente: (Elaboración propia)

| Procesos  | Sistemas |      |         |
|---|----------|------|---------|
|   | SICAP    | SIGE | SOROLLA |
| Control de recursos asignados al área   | ✓        | ✓    | ✓       |
| Ejecución y control de la(s) bonificación(es) de pasajes a estudiantes de pasajes a estudiantes | -        | -    | -       |

|                                      |   |   |   |
|--------------------------------------|---|---|---|
| Control de préstamos estudiantiles   | - | - | - |
| Control de ejecución del presupuesto | ✓ | ✓ | ✓ |

Tabla 2. Procesos del VDEA que ejecutan los sistemas

Fuente: (Elaboración propia)

Como se muestra en la tabla 2, los sistemas analizados no satisfacen las necesidades que presenta el Vicedecanato de Economía y Administración de la UCI, principalmente porque todos ellos son aplicaciones desarrolladas bajo sistemas económicos y políticas tecnológicas distintas a las de Cuba, y con requerimientos funcionales que no contemplan la totalidad de los requisitos planteados por el Vicedecanato Economía y Administración de la Facultad 3.

#### 1.4 Proceso de desarrollo de software

Un proceso de desarrollo de software es la definición del conjunto de actividades que guían los esfuerzos de las personas implicadas en el proyecto, a modo de plantilla que explica los pasos necesarios para terminar el proyecto. Este conjunto de actividades en el proceso de desarrollo de software tiene la misión de transformar los requerimientos del usuario en un producto de software; de manera que los integrantes del equipo y todo aquel interesado en el producto final, tenga la misma visión y no ocurra cuando no se aplica un proceso de desarrollo (Jacobson, y otros, 2000).

Para guiar el proceso de desarrollo de software se requiere de la aplicación de una metodología de desarrollo de software.

##### 1.4.1 Metodología de desarrollo

Una metodología consiste en múltiples herramientas, modelos y métodos para asistir en el proceso de desarrollo de software donde se definen con precisión los artefactos, roles y actividades involucrados, junto con prácticas y técnicas recomendadas, guías de adaptación de la metodología al proyecto, guías para uso de herramientas de apoyo (Figuerola, y otros, 2008).

Para la selección de una metodología se debe tener en consideración las características del proyecto de desarrollo de software. Atendiendo a que el equipo de trabajo es pequeño, el tiempo de entrega del producto es relativamente corto, la comunicación con el cliente es activa y que la universidad se encuentra enfrascada en un programa de mejora de procesos de software, se determinó escoger la metodología AUP con la variación establecida por la UCI.

**Metodología AUP versión de la UCI**

Se seleccionó la metodología aprobada por la universidad para la actividad productiva de la misma de tal forma que se adapte a su ciclo de vida definido, la cual consiste en una variación de la metodología Proceso Unificado Ágil (AUP por sus siglas en inglés). Quedando estructurada en tres fases, las cuales son detalladas a continuación (Ver Figura 3) (CEIGE, 2015) (SIUV, 2015).

| <b>Fases AUP</b> | <b>Fases Variación AUP-UCI</b> | <b>Objetivos de las fases (Variación AUP-UCI)</b>   |
|------------------|--------------------------------|---|
| Inicio           | Inicio                         | Durante el inicio del proyecto se llevan a cabo las actividades relacionadas con la planeación del proyecto. En esta fase se realiza un estudio inicial de la organización cliente que permite obtener información fundamental acerca del alcance del proyecto, realizar estimaciones de tiempo, esfuerzo y costo y decidir si se ejecuta o no el proyecto.   |
| Elaboración      | Ejecución                      | En esta fase se ejecutan las actividades requeridas para desarrollar el software, incluyendo el ajuste de los planes del proyecto considerando los requisitos y la arquitectura. Durante el desarrollo se modela el negocio, obtienen los requisitos, se elaboran la arquitectura y el diseño, se implementa y se libera el producto. Durante esta fase el producto es transferido al ambiente de los usuarios finales o entregado al cliente. Además, en la transición se capacita a los usuarios finales sobre la utilización del software. |
| Construcción     |                                |   |
| Transición       |                                |   |
|                  | Cierre                         | En esta fase se analizan tanto los resultados del proyecto como su ejecución y se realizan las actividades formales de cierre del proyecto.   |

**Figura 3: Ciclo de vida de la Metodología AUP-versión UCI**

Fuente: (CEIGE, 2015)

### 1.4.2 Ingeniería de requisitos

La IEEE define un requisito como una condición o capacidad que tiene que ser alcanzada o poseída por un sistema o componente de un sistema para satisfacer un contrato, estándar, u otro documento impuesto formalmente (IEEE, 1990). Los requisitos o necesidades del cliente, son sometidos a un proceso de ingeniería para su correcta concepción y gestión durante el proceso de desarrollo de software.

La Ingeniería de Requisitos (IR) proporciona los mecanismos para comprender las necesidades del cliente, evaluar la factibilidad de desarrollo, negociar una solución razonable y especificarla sin ambigüedades, validar la especificación y administrar los requisitos (Pressman, 2009). Es reconocida como el proceso de descubrir, analizar, documentar y verificar los servicios y restricciones del sistema (Sommerville, 2011). Pressman (Pressman, 2009) y Sommerville (Sommerville, 2011), profundizan en este tema estableciendo las etapas de la IR, entre las que se encuentran: obtención y análisis, especificación y validación de los requisitos (Pressman, 2009) (Sommerville, 2011), las cuales fueron abordadas durante el desarrollo de la investigación.

Para el desarrollo de una solución de software, no solo es importante tener en cuenta las características del sistema a desarrollar, sino también la arquitectura que debe regir la solución técnica.

### 1.4.3 Patrones arquitectónicos

Los patrones arquitectónicos ofrecen soluciones a problemas de arquitectura en ingeniería de software. Dan una descripción de los elementos y el tipo de relación que tienen, junto a un grupo de restricciones sobre cómo pueden ser usados. Un patrón arquitectónico expresa un esquema de organización estructural esencial para un sistema de software, que consta de subsistemas, sus responsabilidades e interrelaciones (Buschmann, y otros, 1996).

#### 1.4.3.1 Patrón arquitectónico Modelo-Vista-Controlador (MVC)

Dentro de los patrones arquitectónicos existentes, se encuentra el patrón de arquitectura de las aplicaciones de software MVC, el cual resulta uno de los más empleados para el desarrollo de aplicaciones web. Este patrón o modelo, separa los datos de una aplicación, la interfaz de usuario, y la lógica de negocio en tres componentes distintos. El patrón de llamada y retorno MVC, se ve frecuentemente en aplicaciones web, donde la vista es una página de Lenguaje de Marcado de Hipertexto (HTML) y el código que provee datos dinámicos a la página, el modelo incluye el Sistema de Gestión de Base de Datos (SGBD) y la lógica de negocio, y el controlador es el responsable de recibir y atender los eventos de entrada desde la vista (Buschmann, y otros, 1996) (Díaz González, y otros, 2012).

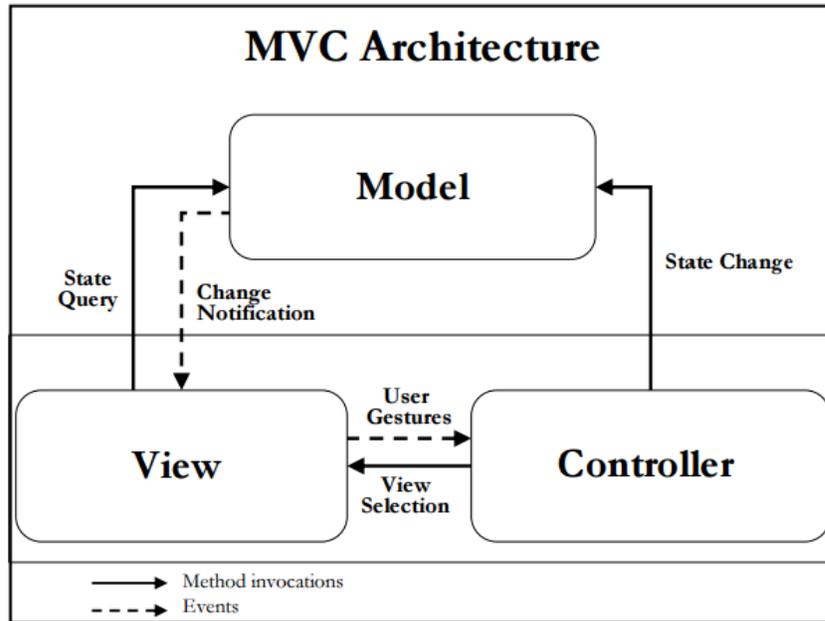


Figura 4: Modelo Vista Controlador

Fuente: (Liu, 2007)

#### 1.4.4 Patrones de diseño

Los patrones de diseño 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 están basados en la recopilación del conocimiento de los expertos en desarrollo de software.

#### Patrones GRASP

Para el desarrollo del módulo se aplican los patrones de diseño *GRASP*, los cuales describen los principios fundamentales de la asignación de responsabilidades a objetos. Entre los más conocidos y utilizados para la solución están (Pressman, 2009):

- **Experto:** cada clase dentro del módulo tiene la responsabilidad de utilizar únicamente la información que ella misma posee para realizar la labor para la que fue concebida. Tal es el caso de las clases del modelo que son las encargadas de toda la lógica del acceso a los datos.
- **Creador:** identifica quien debe ser el responsable de la creación de nuevos objetos o clases, donde la nueva clase deberá ser creada por la clase que tiene toda la información necesaria para realizar la acción, que usa directamente las instancias creadas del objeto, almacena o maneja varias instancias de clase y contiene o agrega la clase.
- **Alta cohesión:** cada clase que pertenece al modelo tiene como responsabilidad fundamental realizar las labores que solo le competen a ella y que no son desempeñadas por otros elementos

## Capítulo 1 Fundamentación teórica

del diseño. Dentro de esas labores se encuentra crear las consultas de acceso a los datos. Todas las clases están agrupadas por las funcionalidades que realizan.

- **Bajo acoplamiento:** las clases que implementan la lógica del negocio y de acceso a datos se encuentran en el modelo, las cuales no tienen asociaciones con las de la vista, lo que proporciona que la dependencia en este caso sea baja.
- **Controlador:** el módulo tiene clases controladoras que se encargan de atender todas las peticiones y pasar los datos de la misma a las clases del modelo para su procesamiento. Al mismo tiempo es la clase que se encarga de enviar las respuestas a la vista.

La aplicación de estos patrones se ve reflejada con la asignación de las responsabilidades por clases para realizar solo las funcionalidades correspondientes a la información que las mismas contienen evitando una sobrecarga. Además, se da la responsabilidad de crear instancias de otras clases solamente a aquellas que contengan a las mismas y se modelan clases controladoras, encargadas de controlar el flujo de las operaciones del sistema. El diseño de la aplicación permite la interacción entre las mismas propiciando un bajo acoplamiento y en consecuencia la reutilización.

### **Patrones Banda de los Cuatro (GoF por sus siglas en inglés)**

Adaptados para solucionar un problema común en particular mediante la descripción de las clases y la comunicación entre objetos; se agrupan en tres categorías: creacionales, estructurales y de comportamiento (Gamma, y otros, 1994).

#### Creacionales:

- **Inicialización vaga:** refleja la delegación de la creación de un objeto o el cálculo de un valor hasta que sea realmente necesitado.

#### Estructurales:

- **Adaptador:** permite que la interfaz de una clase existente pueda ser usada desde otra interfaz sin necesidad de cambiar su código. Como ejemplo de ello está el motor de plantillas Twig donde cada plantilla redefine los bloques a su beneficio sin necesidad de cambiar la base.

#### Comportamiento:

- **Iterador:** provee una manera de acceder a los elementos de un objeto sin necesidad de exponer su representación interna, facilitando las iteraciones y el filtrado sobre colecciones de objetos.

### **1.4.5 Estándares de codificación**

Un estándar de codificación comprende aspectos de la generación de código que los programadores deben implementar de forma prudente. El código fuente debe reflejar un estilo armonioso, como si un

único programador hubiese escrito todo el código de una sola vez. Es importante establecer un estándar de codificación al comenzar un proyecto para propiciar el trabajo de forma coordinada. El propósito principal para llevar a cabo revisiones del código a lo largo de todo el desarrollo es localizar defectos en el mismo, las revisiones también pueden afianzar los estándares de codificación de manera uniforme. Usar técnicas de codificación sólidas y aplicar buenas prácticas de programación con vistas a generar un código de alta calidad es de gran importancia para obtener un buen rendimiento y contribuir a la calidad del software (Microsoft, 2015).

### 1.4.6 Calidad de software

La calidad del software es el grado en que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario. Es el conjunto de características de una entidad que le confieren su aptitud para satisfacer las necesidades expresadas y las implícitas (IEEE, 2002), sirven como soporte a los procesos de verificación y validación de software para garantizar la calidad de un producto.

#### Verificación y validación de software

La definición de verificación y validación (V&V) abarca muchas actividades de aseguramiento de calidad, como son: revisiones técnicas, auditorías de calidad y configuración, monitoreo de desempeño, simulación, estudio de viabilidad, revisiones de documentación y bases de datos, análisis de algoritmos y pruebas de desarrollo, calidad, aceptación e instalación. Las pruebas juegan un rol extremadamente importante en las verificaciones y validaciones, aunque no dejan de ser necesarias otras actividades (Pressman, 2009).

Las V&V son vistas también como los procesos de chequeo y análisis que se realizan durante y después de la implementación, para asegurar que el programa desarrollado cumpla con sus especificaciones y con las funcionalidades esperadas por el cliente. Estas actividades se realizan en todas las etapas del desarrollo de software (Sommerville, 2011).

#### Verificación

La verificación es el proceso de determinar si los productos de una determinada fase del desarrollo de software cumplen o no los requisitos establecidos (IEEE, 2004). Ayuda a comprobar si se ha construido el producto correctamente. Estas verificaciones se realizan por parte del equipo de desarrollo (Boehm, 1984) (Pressman, 2009) (Sommerville, 2011).

#### Validación

La validación, por otra parte, es el proceso de evaluación del software al final del proceso de desarrollo para asegurar el cumplimiento de las necesidades del cliente (IEEE, 1983) (IEEE, 2004), ayuda a

comprobar si se ha construido el producto correcto (Boehm, 1984) (Pressman, 2009) (Sommerville, 2011).

Las V&V comienzan con las revisiones de requisitos y continúan con las del diseño y las inspecciones de código en las pruebas al producto (Sommerville, 2011). Para su ejecución se aplican técnicas de evaluación que pueden ser estáticas o dinámicas.

Las técnicas de evaluación estáticas buscan fallas sobre el sistema en reposo y estudian los distintos modelos que componen el producto de software. Se les conoce de modo genérico como revisiones, propician el intercambio de información entre participantes y reducen el esfuerzo de realizar re-trabajo (Juristo, y otros, 2004).

### **Métricas para la validación de requisitos**

La validación de los requisitos debe estar orientada a certificar que los requisitos definidos cumplen con los atributos de calidad (Davis, y otros, 2005) (García Rodríguez, 2010):

- No ambiguo

Un requisito no es ambiguo, sí y solo sí, tiene una única posible interpretación. Múltiples interpretaciones de un requisito pueden causar desacuerdos entre clientes y desarrolladores. Este atributo se mide a partir del porcentaje de requisitos interpretados de una única manera:

$$Q_1 = \frac{n_{ui}}{n_r} \quad (1)$$

Donde:

$Q_1$  = coeficiente de no ambigüedad

$n_{ui}$  = número de requisitos con una única interpretación

$n_r$  = número total de requisitos identificados.

Cuando el valor resultante se aproxima a 0, se interpreta como que los requisitos tienen múltiples interpretaciones; cuando el valor se aproxima a 1, se interpreta como que los requisitos tienen una única interpretación. Debido a que este atributo es muy crítico para el éxito de un proyecto se recomienda que  $Q_1 = 1$ .

- Comprensible

Un requisito es comprensible cuando todos lo leen (clientes, usuarios, desarrolladores) y comprenden su significado con un mínimo de explicación.

$$Q_2 = \frac{n_{ur}}{n_r} \quad (2)$$

Donde:

$Q_2$  = coeficiente del atributo de calidad comprensible

$n_{ur}$  = número de requisitos identificados que todos los revisores entienden

$n_r$  = número total de requisitos identificados

El valor resultante oscila en el rango entre 0 (ningún requisito comprendido) y 1 (todos los requisitos comprendidos). Debido a que este atributo es crítico para el éxito de un proyecto se recomienda que  $Q_2 = 1$ .

### - Correcto

Un requisito es correcto, sí y solo sí, representa una necesidad de función del sistema a construir.

Este atributo puede ser medido de la siguiente manera:

$$Q_3 = \frac{n_c}{n_r} \quad (3)$$

Donde:

$Q_3$  = coeficiente del atributo de calidad correcto

$n_c$  = número de requisitos correctos,

$n_r$  = número total de requisitos identificados.

### - Posible de trazar

Un requisito es posible de trazar, sí y solo sí, es escrito de manera que facilite la referencia de cada requisito individualmente a los requisitos con los cuales se relaciona, así como a la fuente de origen en el modelo del negocio.

$$Q_4 = \frac{n_{pt}}{n_r} \quad (4)$$

Donde:

$Q_4$  = coeficiente del atributo de calidad posible de trazar

$n_{pt}$  = número de requisitos identificados posibles de trazar

$n_r$  = número de requisitos identificados

## **Métricas para la validación del diseño**

Una métrica es una medida efectuada sobre los programas, documentación, su desarrollo y mantenimiento que permite medir de forma cuantitativa la calidad de los atributos internos del software (Pressman, 2009).

## Capítulo 1 Fundamentación teórica

Lorenz y Kidd proponen un conjunto de métricas para la validación del diseño del software siguiendo el paradigma orientado a objetos, para ello las clasifican en cuatro amplios grupos: tamaño, herencia, valores internos y valores externos. Las métricas orientadas al tamaño de las clases se centran en el recuento de atributos y operaciones para cada clase y los valores promedio para el sistema como un todo; las métricas basadas en la herencia se centran en la forma en que las operaciones se reutilizan en la jerarquía de clases; las métricas para valores internos examinan la cohesión y los aspectos orientados al código y las métricas orientadas a valores externos examinan el acoplamiento y reutilización entre las clases. Una sus métricas más empleadas es Tamaño de Clases (TC), pues permite medir el total de atributos y operaciones encapsulados en una clase para valorar la sobrecarga de responsabilidades asignadas (Lorenz, et al., 1994) (Pressman, 2009).

Chidamber y Kemerer también realizan una propuesta para medir la calidad del diseño, entre las cuales se encuentran: Métodos Ponderados por Clases (MPC), Árbol de Profundidad de Herencia (APH), Numero de Descendientes (ND), Acoplamiento entre Clases Objeto (ACO), Relaciones entre Clases (RC) y Carencia de Cohesión de los Métodos (CCM) (Chidamber, y otros, 1994) (Pressman, 2009).

Por la relevancia para la investigación se considera oportuno aplicar de Lorenz y Kidd la métrica TC, pues permite visualizar si se distribuyen correctamente las asignaciones de responsabilidades entre las clases, verificándose así la cohesión y armonía entre las mismas, y de Chidamber y Kemerer las métricas RC para evaluar el grado de acoplamiento entre las clases y CCM para constatar el correcto cumplimiento del patrón Experto.

En la siguiente tabla se muestra el resumen de las métricas a aplicar para validar el diseño de la solución.

| Métrica | Aplicación   |
|---------|--|
| RC      | RC = cantidad de posibles métodos a ejecutarse al hacer uso de un objeto de otra clase   |
| CCM     | CCM = 0 cuando no existan uno o varios métodos que accedan a un mismo atributo<br>CCM = X<br>X: cantidad de métodos que acceden a uno o varios atributos en común. |
| TC      | TC = T/CC<br>T = total de operaciones de cada una de las clases<br>CC = cantidad de clases   |

**Tabla 3. Métricas para la validación del diseño**

Fuente: (Lorenz, et al., 1994) (Chidamber, y otros, 1994) (Pressman, 2009)

“Las pruebas son parte inevitable en el desarrollo responsable de software”. Un caso de prueba consiste en un conjunto de datos de entrada, condiciones de ejecución y resultados esperados, que son desarrollados para cumplir una función esperada del sistema. (Pressman, 2009)

Las técnicas de evaluación dinámicas generan entradas al sistema en ejecución con el objetivo de detectar fallos. La aplicación de estas técnicas se conoce como pruebas de software y se aplican generalmente sobre el código (Juristo, et al., 2004).

Para una mejor organización de la ejecución de las pruebas de software durante el proceso de desarrollo es conveniente desarrollar una estrategia de pruebas de software.

### **Estrategia de pruebas de software**

La estrategia de pruebas de software describe el enfoque y los objetivos generales de las actividades de prueba. Incluye los niveles de prueba, el tipo de prueba a ser ejecutada y los casos de prueba diseñados para lograr los objetivos (Pressman, 2009).

### Niveles de prueba

Los niveles de prueba son aplicados en diferentes escenarios o niveles de trabajo (Pressman, 2009):

- Prueba de desarrollador: es la prueba diseñada e implementada por el equipo de desarrollo. Tradicionalmente estas pruebas han sido consideradas solo para la prueba de unidad, aunque en la actualidad, en algunos casos, pueden ejecutar pruebas de integración. Se recomienda que estas pruebas cubran más que las pruebas de unidad.
- Prueba independiente: es la prueba que es diseñada e implementada por alguien independiente del grupo de desarrolladores. El objetivo de estas pruebas es proporcionar una perspectiva diferente y en un ambiente más rico que los desarrolladores. Una vista de la prueba independiente es la prueba independiente de los stakeholders, que son pruebas basadas en las necesidades y preocupaciones de los stakeholders.
- Prueba de unidad: es la prueba enfocada a los elementos testeables más pequeño del software. Es aplicable a componentes representados en el modelo de implementación para verificar que los flujos de control y de datos están cubiertos, y que ellos funcionen como se espera. Antes de iniciar cualquier otra prueba es preciso probar el flujo de datos de la interfaz del componente. Si los datos no entran correctamente, todas las demás pruebas no tienen sentido. El diseño de casos de prueba de una unidad comienza una vez que se ha desarrollado, revisado y verificado en su sintaxis el código a nivel fuente.
- Prueba de sistema: se hacen cuando el software está funcionando como un todo, están dirigidas a verificar el programa final después que todos los componentes han sido integrados.

## Capítulo 1 Fundamentación teórica

- Prueba de integración: es ejecutada para asegurar que los componentes en el modelo de implementación operen correctamente cuando son combinados para ejecutar un caso de uso. Se aprueba un paquete o un conjunto de paquetes del modelo de implementación. Estas pruebas descubren errores o falta de completitud en las especificaciones de las interfaces de los paquetes. Esta prueba debe ser responsabilidad de desarrolladores y de independientes, sin solaparse las pruebas. Es el proceso de combinar y probar múltiples componentes juntos. El objetivo es tomar los componentes probados en unidad y construir una estructura de programa que esté de acuerdo con lo que dicta el diseño.
- Prueba de aceptación: prueba de aceptación del usuario es la prueba final antes del despliegue del sistema. 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.

### Métodos de prueba

- Pruebas de caja negra: se centran en verificar el cumplimiento de los requisitos funcionales del software. Permiten encontrar:
  - o Funciones incorrectas o ausentes
  - o Errores de interfaz
  - o Errores en estructuras de datos o en accesos a las bases de datos externas
  - o Errores de rendimiento
  - o Errores de inicialización y terminación
- Pruebas de caja blanca: se centran en evaluar la ejecución por lo menos una vez de cada sentencia del programa. Estas pruebas deben garantizar como mínimo que:
  - o Se ejerciten por lo menos una vez todos los caminos independientes para cada módulo.
  - o Se ejerciten todas las decisiones lógicas en sus vertientes verdaderas y falsas.
  - o Se ejerciten las estructuras internas de datos para asegurar su validez.

El software debe cumplir con ciertos atributos de calidad que se encuentran en correspondencia con los Requisitos No Funcionales (RFN). Para verificar que el sistema cumpla con estos atributos de calidad se aplican diferentes tipos de pruebas

Entre los tipos de pruebas que pueden aplicarse se encuentran: (Hetzel, 1993)

- **Funcionalidad:** verifican la habilidad del software para realizar el trabajo deseado.
- **Usabilidad:** verifican la habilidad del software para satisfacer al usuario.
- **Portabilidad:** verifican la habilidad del software para correr en diferentes entornos informáticos.

## Capítulo 1 Fundamentación teórica

El principal objetivo del flujo de pruebas es evaluar la calidad del producto a través de la búsqueda y documentación de errores, la validación del cumplimiento de los requisitos y la validación del desempeño.

Si bien es importante comprobar que se desarrolla un producto correctamente, también es esencial corroborar que se desarrolló el producto correcto, para ello es aconsejable aplicar técnicas que permitan medir la satisfacción del cliente.

### Técnica ladov

La técnica ladov permite medir la satisfacción del cliente con un producto, 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 un 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 (Kuzmina, 1970).

1. Máxima satisfacción.
2. Más satisfecho que insatisfecho.
3. No definida.
4. Más insatisfecho que satisfecho.
5. Máxima insatisfacción.
6. Contradictoria.

| Pregunta cerrada 3                 | Pregunta cerrada 1 |       |    |       |       |    |    |       |    |
|------------------------------------|--------------------|-------|----|-------|-------|----|----|-------|----|
|                                    | No                 |       |    | No sé |       |    | Sí |       |    |
|                                    | Pregunta cerrada 2 |       |    |       |       |    |    |       |    |
|                                    | 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 4. Cuadro lógico de ladov.**

Fuente: (Kuzmina, 1970)

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 5. Índice de satisfacción de ladov

Fuente: (Kuzmina, 1970)

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 o 6

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

A partir del análisis realizado se decide aplicar el nivel de prueba integración; los tipos de prueba de funcionalidad, usabilidad y portabilidad; así como el método de prueba caja negra.

### 1.5 Herramientas y tecnologías a utilizar

Por las necesidades del VDEA de poder contar con una información asequible tanto por la vicedecana, como los jefes de áreas y estudiantes de la facultad desde cualquier estación de trabajo en la universidad, se hace necesario que la implementación sea una solución web.

#### 1.5.1 Aplicación web

Una aplicación web consiste de software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador en un documento electrónico que contiene información. Están formadas por diferentes módulos, independientes entre sí, que se pueden ejecutar en el servidor o en el cliente y pueden estar implementados utilizando lenguajes o tecnologías diferentes (Pardo Kuklinski, 2005).

### 1.5.1 Servidor web

Existen numerosos servidores web dentro de los que se encuentran: Microsoft IIS, Java System Web Server, APACHE, entre otros.

#### Apache server:

Apache es software libre y uno de los servidores web más populares. Más del 70% de los sitios web en internet están manejados por Apache, haciéndolo más extensamente usado que todos los otros servidores web juntos ( Apache Software Foundation, 2011).

Entre las características más relevantes de Apache se encuentran:

- Apache es un servidor web flexible, rápido y eficiente, continuamente actualizado y adaptado a al Protocolo de Transferencia de Hipertexto (HTTP).
- Multiplataforma.
- Modular: puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona, y con la *Application Programming Interface* (API) de programación de módulos, para el desarrollo de módulos específicos.
- Extensible: gracias a ser modular se han desarrollado diversas extensiones entre las que destaca *HyperText Preprocessor* (PHP), un lenguaje de programación del lado del servidor.

Entre las ventajas que presenta el servidor Apache se encuentran:

- La arquitectura modular de Apache es personalizable.
- Permite construir un servidor hecho a la medida.

Se considera Apache en su versión 2.0 como servidor web para el desarrollo del presente trabajo, pues además de las ventajas de ser de código abierto, gratis y de fácil adquisición, Apache es el servidor web más empleado en los centros de desarrollo de la UCI.

### 1.5.2 Lenguajes de programación

Entre los lenguajes de programación existentes de código abierto para desarrollar aplicaciones orientadas a la web se encuentran dos grupos fundamentales de acuerdo con la arquitectura Cliente/Servidor, la programación del lado del servidor y la programación del lado del cliente. La programación del lado del cliente incluye aquellos lenguajes que son únicamente interpretados por una aplicación cliente como el navegador web, entre estos lenguajes se encuentra HTML. Los lenguajes de programación del lado servidor son los reconocidos, ejecutados e interpretados por el propio servidor y que se envían al cliente en un formato comprensible para él. Para el desarrollo de la herramienta propuesta se emplean los siguientes lenguajes:

### **HTML5**

HTML5 es mantenido por la *World Wide Web Consortium* (W3C), es utilizado para la creación de páginas web estáticas. Esta versión incorpora algunas etiquetas nuevas pensadas para hacer que la estructura de la página web sea más lógica y funcional.

El enfoque general ha cambiado bastante respecto a versiones anteriores de HTML, añadiendo semántica y accesibilidad implícitas, especificando cada detalle y borrando cualquier ambigüedad. También se tiene en cuenta que muchas páginas web actuales son dinámicas, pareciéndose más a aplicaciones que a documentos. Algo básico es que HTML5 está definido en base al *Document Object Model* (DOM), la representación interna de una web con la que trabaja un navegador, dejando de lado la representación real y definiendo a la vez un estándar HTML y XHTML (Nieto, 2010).

### **Hojas de Estilos en Cascada (CSS 3, por sus siglas en inglés)**

CSS es un lenguaje creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas web complejas. Separar la definición de los contenidos y la definición de su aspecto presenta numerosas ventajas, entre las que se encuentra la posibilidad de crear documentos HTML/XHTML bien formateados y con significado completo (también llamados documentos semánticos). Además, mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes (Pérez, 2007).

Al crear una página web, se utiliza en primer lugar el lenguaje HTML/XHTML para marcar los contenidos, es decir, para designar la función de cada elemento dentro de la página: párrafo, titular, texto destacado, tabla, lista de elementos, entre otros. Una vez creados los contenidos, se utiliza el lenguaje CSS para definir el aspecto de cada elemento: color, tamaño y tipo de letra del texto, separación horizontal y vertical entre elementos, posición de cada elemento dentro de la página, además de otras funcionalidades (Pérez, 2007).

La versión 3 brinda un mayor control de la presentación y los estilos, permite ahorrar tiempo y trabajo mediante las técnicas que brinda, además, la versión 3 de Bootstrap implementada trae por defecto estilos predefinidos de CSS 3.

### **PHP 5.4.4-14**

PHP es un lenguaje de programación rápido, gratuito, interpretado, usado normalmente para la creación de páginas web dinámicas, con una gran librería de funciones y una amplia documentación; se encuentra incluido dentro de los lenguajes del lado del servidor.

## Capítulo 1 Fundamentación teórica

PHP es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML. Entre sus principales características cabe destacar su potencia, su alto rendimiento, su facilidad de aprendizaje, su escasez de consumo de recursos y que es extremadamente simple para el principiante, pero a su vez, ofrece muchas características avanzadas para los programadores profesionales (The PHP Group, 2001).

Entre las ventajas que tiene PHP se encuentran que es un lenguaje multiplataforma, con capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, capacidad de expandir su potencial al utilizar una enorme cantidad de módulos (llamados extensiones), posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda. Es libre y permite las técnicas de Programación Orientada a Objetos, posee una biblioteca nativa de funciones sumamente amplia, no requiere definición de tipos de variables y facilita el manejo de excepciones. (ProgramaciónWeb.Net, 2011).

### 1.5.3 Entorno de Desarrollo Integrado (IDE)

#### PhpStorm 8

Trae consigo tecnologías emergentes para ayudar a disfrutar el desarrollo web con un entendimiento profundo del código y soporte avanzado para ambientes remotos. PhpStorm ofrece un editor enriquecido e inteligente para PHP que realmente comprende el código y entiende profundamente su estructura, soportando PHP 5.3, 5.4, 5.5 & 5.6 para proyectos modernos y legados. El IDE provee finalización inteligente de código, señalamiento de sintaxis, configuración extendida de formato de código, chequeo de errores al instante, plegado de código, soporte para mezclas de lenguajes y más (JetBrains, 2000).

### 1.5.4 SGBD

Un SGBD o *Data Base Management System* (DBMS) es una colección de programas cuyo objetivo es servir de interfaz entre la base de datos, el usuario y las aplicaciones. Está compuesto por un lenguaje de definición de datos, un lenguaje de manipulación de datos y un lenguaje de consulta. Un SGBD permite definir los datos a distintos niveles de abstracción y manipular dichos datos, además de garantizar la seguridad e integridad de los mismos (Ramos Martín, y otros, 2006).

#### PostgreSQL

PostgreSQL es un SGBD relacional, orientado a objetos y libre, que tiene prestaciones y funcionalidades equivalentes a muchos gestores de bases de datos comerciales.

Dentro de sus principales ventajas se encuentran:

- Soporta distintos tipos de datos.
- Posee una gran escalabilidad, haciéndolo idóneo para su uso en sitios web que atienden un gran número de solicitudes.
- Puede ser instalado un número ilimitado de veces sin temor de sobrepasar la licencia.
- Posee estabilidad y confiabilidad legendaria.
- Es extensible a través del código fuente, disponible sin costos adicionales.
- Soporte nativo para los lenguajes más populares del medio: PHP, C++, Perl, Python, entre otros.
- Extensiones para alta disponibilidad, nuevos tipos de índices, datos especiales, minería de datos, entre otros.
- Es multiplataforma, disponible en Linux, Unix, Mac Os X y Windows, entre otros (Equipo de desarrollo de PostgreSQL, 1996).

Se seleccionó **PostgreSQL** en su versión 9.2, debido a que es multiplataforma, confiable, estable, con gran escalabilidad, control de concurrencia y funcionalidades que lo destacan como uno de los SGBD más potentes en la actualidad.

### **PgAdmin 3 v1.14.2-2**

PgAdmin 3 es una aplicación que maneja y diseña base de datos para ser usadas con PostgreSQL. La aplicación puede ser usada para administrar PostgreSQL 7.3 y superiores, corre en cualquier plataforma. PgAdmin3 está diseñado para responder ante las necesidades de todos los usuarios, desde escribir simples consultas SQL hasta desarrollar bases de datos complejas. La interfaz gráfica soporta todos los componentes de PostgreSQL y permite una fácil administración. La aplicación también incluye un editor señalamientos de sintaxis SQL, un editor de código server-side, un agente de SQL/batch/shelljobscheduling, soporte para motores de replicación Slony-I y más. Se pueden realizar conexiones al servidor utilizando puertos TCP/IP o Dominios Unix, y puede ser encriptado mediante SSL para protección. No se requieren drivers adicionales para comunicarse con el servidor de base de datos, además, está integrado en la instalación de PostgreSQL.

### **1.5.5 Marco de Trabajo (*Framework*)**

#### **Symfony 2.5.6**

Es un *framework* para aplicaciones web desarrollado en PHP5. Symfony permite la reutilización de código, con lo que agiliza el proceso de creación y mantenimiento de aplicaciones web. Su diseño permite optimizar las aplicaciones y mantener una organización de la misma. Además reduce el tiempo de desarrollo de estas, pues este *framework* proporciona varias herramientas y clases, las cuales pueden ser reutilizadas por los desarrolladores.

## Capítulo 1 Fundamentación teórica

Symfony es compatible con la mayoría de los gestores de base de datos como MySQL, PostgreSQL, Oracle y SQL Server de Microsoft. Es multiplataforma pues se puede ejecutar tanto en Unix (Unix, Linux, entre otros) como plataformas Windows. Permite el desarrollo por capas, pues está basado en el patrón arquitectónico MVC, toma lo mejor de esta arquitectura y la implementa de forma que el desarrollo de aplicaciones sea rápido y sencillo.

Symfony2 es la versión más reciente de Symfony, el popular *framework* implementa dentro de su configuración el patrón MVC para desarrollar aplicaciones PHP. Se anunció por primera vez a principios de 2009 y supone un cambio radical tanto en arquitectura interna como en filosofía de trabajo respecto a sus versiones anteriores. Ha sido ideado para exprimir al límite todas las nuevas características de PHP 5.3 y por eso es uno de los *framework* PHP con mejor rendimiento. Su arquitectura interna está completamente desacoplada, lo que permite reemplazar o eliminar fácilmente aquellas partes que no encajan en el proyecto (symfony.es, 2015).

Entre sus características están:

- **Alto rendimiento:** Symfony2 ha sido desarrollado teniendo en cuenta el rendimiento como mayor prioridad, por lo que es uno de los frameworks más rápidos. Hasta 3 veces más rápido que Symfony 1.4 o Zend Framework 1.10 y consume la mitad de la memoria.
- **Usabilidad avanzada:** es un framework fácil de utilizar gracias a que cuenta con una API de desarrollo muy intuitiva.
- **Extensible:** Symfony2 se construye a base de bundles (plugins en Symfony 1).
- **Flexible:** gracias a que Symfony2 cuenta con un micro-kernel basado en un contenedor de inyección de dependencia y un manejador de eventos es muy fácil configurarlo a voluntad.
- **Construido para desarrolladores:** Symfony2 proporciona las herramientas que en gran medida mejoran la productividad de los desarrolladores, como la famosa barra de depuración web, soporte nativo de entornos, páginas detalladas de errores entre otras cosas.
- **Construido en base a otros grandes frameworks:** Symfony2 tomó lo mejor de los conceptos de otros frameworks de desarrollo como Django, Spring y Ruby on Rails. También aprovecha componentes de Zend Framework y de Doctrine.

### Bootstrap 3

Bootstrap hace uso de ciertos elementos HTML y propiedades CSS que requieren del uso de HTML5 doctype. Con Bootstrap 2, se añadieron estilos móviles opcionales amistosos para aspectos esenciales del framework. Con Bootstrap 3, se ha reescrito el proyecto para ser adaptable a los móviles desde el inicio. En lugar de adicionar estilos para móviles opcionales, los mismos son añadidos desde el núcleo (MIT, 2015).

### **jQuery1.9**

jQuery es un nuevo tipo de librería JavaScript que simplifica el recorrido de documentos HTML, el manejo de eventos, animaciones, interacciones Ajax para un rápido desarrollo de aplicaciones web. Esta versión marca una importante evolución en jQuery, muchas funcionalidades obsoletas han sido retiradas del framework para volverlo más liviano y limpio; y para prepararnos a la versión 2.0. Será la última versión con soporte a IE6, IE7 e IE8. Además del retiro de las funcionalidades obsoletas y corrección de bugs, la v.1.9 de jQuery también nos trae nuevas características: `.css()`. (jQuery Foundation, 2015)

### **1.5.6 Computer Aided Software Engineering (CASE)**

Las herramientas CASE brindan soporte para desarrollar y mantener software. Además, se consideran de gran ayuda al desarrollador de software o administrador de proyecto, durante una o más fases del desarrollo de software (Dpto. Sistemas Informáticos y Computación, 2013).

Para el desarrollo del presente trabajo se selecciona la herramienta CASE *Visual Paradigm*.

### **Visual Paradigm 8.0**

Es una herramienta CASE 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 rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. La herramienta CASE también proporciona abundantes tutoriales de UML, demostraciones interactivas de UML y proyectos UML (Visual Paradigm, 2014).

Características:

- Soporte de UML versión 2.1
- Diagramas de Procesos de Negocio
- Ingeniería inversa
- Modelo a código, diagrama a código
- Diagramas de flujo de datos
- Soporte ORM
- Generación de bases de datos
- Transformación de diagramas de Entidad-Relación en tablas de base de datos
- Distribución automática de diagramas

### 1.5.7 Lenguaje Unificado de Modelado (UML)

UML es un lenguaje muy conocido y utilizado en el modelado de sistemas de software, permite visualizar, especificar, construir y documentar el sistema a desarrollar. Ofrece un estándar para detallar un plano del sistema (modelo), el cual incluye características conceptuales como procesos de negocio, funciones del sistema, aspectos concretos como expresiones de lenguaje de programación, esquemas de bases de datos, entre otros (Rumbaugh, y otros, 2000).

Se selecciona UML 2.0 como lenguaje de modelado pues presenta un conjunto de herramientas que permiten modelar (analizar y diseñar) sistemas orientados a objetos, con la modelación de los de artefactos durante las primeras fases del ciclo de vida del software. Los implementadores, en fases posteriores tendrán mayor dominio y comprensión sobre qué es lo que se debe implementar, permitiendo tanto al cliente como a los desarrolladores tener una representación real del alcance y factibilidad que puede llegar a tener el producto.

### 1.6 Conclusiones parciales

- En el presente capítulo se abordaron los elementos fundamentales relacionados con los sistemas de gestión y las aplicaciones web. Además se plantearon los conceptos básicos asociados al objeto de estudio lo que permitió tener una mejor visión del problema. Partiendo del estudio de soluciones existentes de sistemas de gestión de procesos a nivel internacional fundamentalmente, se puede afirmar que los mismos no brindan una solución completa al problema existente en el área económica de la Facultad 3. Dicho estudio además, sirve de apoyo a la investigación para lograr un mejor entendimiento de estos sistemas, y al análisis de la situación problemática antes planteada; además, aportó ideas para el diseño y desarrollo del sistema propuesto.
- Se determinó el empleo de tecnologías como: servidor web apache server, HTML5, Metodología de desarrollo AUP con versión para la UCI, PHP5.4.4-14, PhpStorm 8, PostgreSQL v9.2, pgAdmin3 v1.14.2-2, Symfony 2.5.6, Bootstrap 3, jQuery1.9, Visual Paradigm 8.0 y UML.
- Además, se decidió aplicar para los procesos de verificación y validación de la calidad del software el nivel de prueba de integración; los tipos de prueba de funcionalidad, fiabilidad, usabilidad y portabilidad; así como el método de prueba caja negra.

### CAPÍTULO 2 CARACTERÍSTICAS DEL MÓDULO ECONÓMICO DEL SAEF3

#### 2.1 Introducción

En el presente capítulo se definen los requisitos funcionales y no funcionales para el desarrollo del módulo y se aplican métricas de validación de requisitos para corroborar el cumplimiento de los atributos de calidad. Se muestran además los prototipos de interfaz de usuario, los artefactos resultantes del diseño y el cumplimiento de los patrones del diseño mediante la aplicación de métricas.

#### 2.2 Diagrama de procesos del negocio

Para representar de forma clara y precisa los procesos que se desarrollan en el área económica del VDEA, se desarrollaron los diagramas de procesos del negocio que permiten visualizar gráficamente la secuencia lógica de las actividades y facilitan la comprensión entre el cliente y los desarrolladores. A continuación se muestra el diagrama correspondiente al proceso Asignación de Recursos Materiales (ver Figura 4), el resto de los diagramas de procesos de negocio se refleja en el Anexo #1.

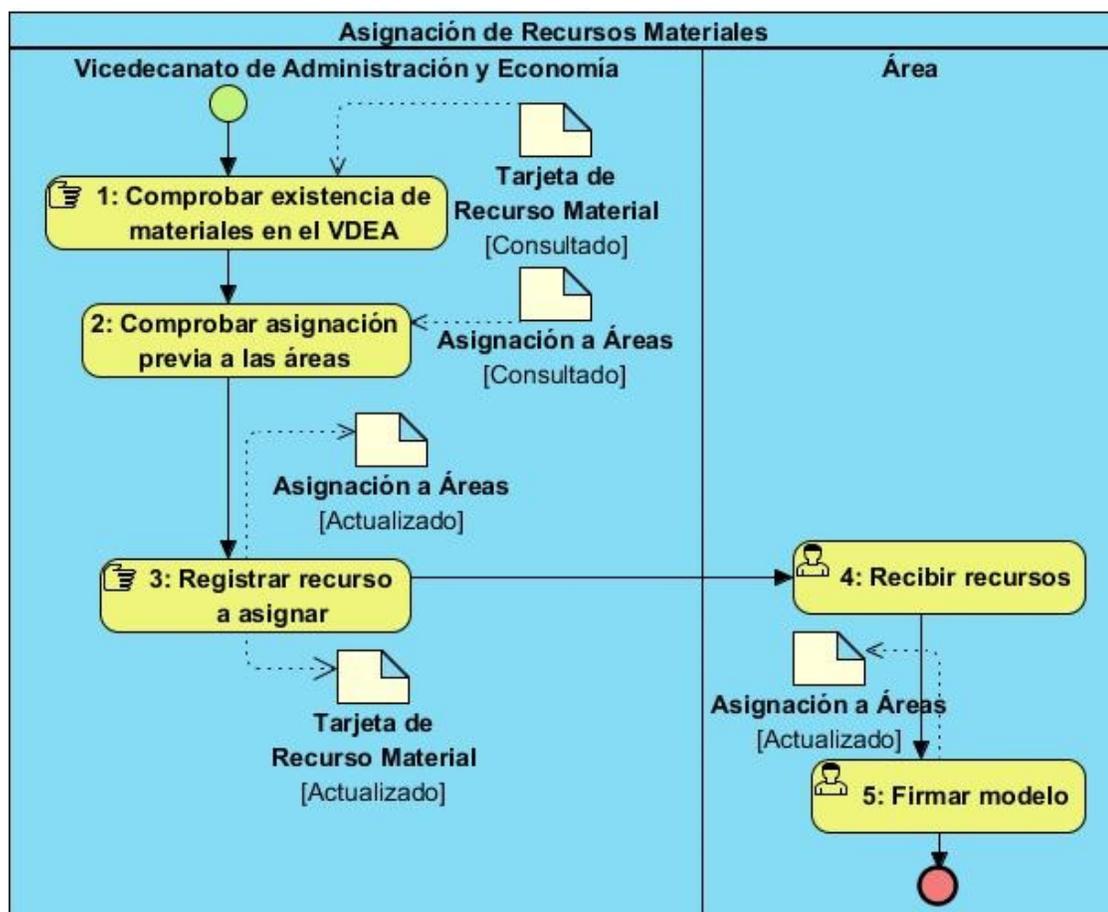


Figura 5. Asignación de recursos materiales.

Fuente: (Elaboración propia)

## Capítulo 2 Características del módulo Económico del SAEF3

### 2.3 Descripción de la solución

El módulo Económico del SAEF3 es la solución propuesta que ofrece funcionalidades de apoyo a las actividades que tienen lugar en el área económica del VDEA. Fue concebido para brindar un ambiente de desarrollo que le permita al vicedecanato efectuar y controlar los procesos vinculados al área mencionada anteriormente de un modo sencillo y fácil que libera al proceso en general de un gran cúmulo de probabilidades de cometer errores humanos y reduce las pérdidas de información.

El módulo Económico del SAEF3 es el encargado de gestionar los procesos referentes al control de: recursos asignados al área, bonificaciones de pasajes a estudiantes, préstamos y ayudas económicas, planificación y ejecución del presupuesto, con el objetivo de:

- Elevar la disponibilidad de la información.
- Mejorar el control de la información generada en los procesos.
- Proveer un mecanismo de generación de reportes.

Para ello, se cuenta con una serie de funcionalidades que proporcionan solución a los problemas que presenta en estos momentos el VDEA, ofrece un medio donde se facilita el acceso a la documentación por parte de los involucrados y propicia un mayor control de los datos que permiten una gestión económica responsable por parte del vicedecanato de la facultad.

El módulo proporciona la gestión de usuarios y el establecimiento de roles para garantizar acceso apropiado y seguridad a cada una de las funcionalidades, así como un registro de las trazas de los usuarios que requiera el módulo. Permite manejar los procesos que actualmente se realizan manualmente, a través de una aplicación web que presenta una interfaz visual fácil de manipular por parte de los trabajadores del VDEA, proporcionando la generación de reportes para una actuación proactiva en el control de la actividad económica de la facultad.

### Personal involucrado al módulo Económico del SAEF3

A continuación se lista el personal con acceso al módulo Económico (ver Tabla 6):

| Proceso                                 | Personal                                | Niveles de permiso  |
|---|---|---------------------|
| Bonificaciones de pasajes a estudiantes | Vicedecana de Economía y Administración | Escritura y lectura |
|   | Especialista VDEA                       | Lectura             |
|   | Secretaria VDEA                         | Escritura y lectura |
|   | Decana                                  | Lectura             |

## *Capítulo 2 Características del módulo Económico del SAEF3*

|                                   |   |   |
|-----------------------------------|---|---|
| Préstamos y ayudas económicas     | Vicedecana de Economía y Administración | Escritura y lectura                     |
|                                   | Especialista VDEA                       | Escritura y lectura                     |
|                                   | Secretaria VDEA                         | Escritura y lectura                     |
|                                   | Estudiante facultad                     | Escritura (parcial) y lectura (parcial) |
|                                   | Decana                                  | Lectura                                 |
| Planificación del presupuesto     | Vicedecana de Economía y Administración | Escritura y lectura                     |
|                                   | Especialista VDEA                       | Lectura                                 |
|                                   | Secretaria VDEA                         | Lectura                                 |
|                                   | Jefes de áreas                          | Lectura                                 |
| Ejecución del presupuesto         | Vicedecana de Economía y Administración | Escritura y lectura                     |
|                                   | Especialista VDEA                       | Lectura                                 |
|                                   | Secretaria VDEA                         | Lectura                                 |
|                                   | Jefes de áreas                          | Lectura                                 |
| Solicitud de recursos materiales  | Vicedecana de Economía y Administración | Escritura y lectura                     |
|                                   | Secretaria VDEA                         | Escritura y lectura                     |
|                                   | Especialista VDEA                       | Escritura y lectura                     |
|                                   | Jefes de áreas                          | Lectura                                 |
| Asignación de recursos materiales | Vicedecana de Economía y Administración | Escritura y lectura                     |
|                                   | Secretaria VDEA                         | Escritura y lectura                     |
|                                   | Especialista VDEA                       | Escritura y lectura                     |
|                                   | Jefes de áreas                          | Lectura (parcial)                       |

**Tabla 6. Personal involucrado al módulo Económico.**

Fuente: (Elaboración propia)

## *Capítulo 2 Características del módulo Económico del SÆF3*

### **2.3.1 Requisitos funcionales**

Para el desarrollo de la solución se identificaron 87 requisitos funcionales:

- RF1 Gestionar solicitud de recursos materiales
  - RF1.1 Solicitar recursos materiales
  - RF1.2 Modificar solicitud de recursos materiales
  - RF1.3 Consultar solicitud de recursos materiales
  - RF1.4 Listar solicitudes de recursos materiales
  - RF1.5 Eliminar solicitud de recursos materiales
- RF2 Gestionar asignación de recursos materiales
  - RF2.1 Asignar recursos materiales
  - RF2.2 Modificar asignación de recursos materiales
  - RF2.3 Consultar asignación de recursos materiales
  - RF2.4 Listar asignaciones de recursos materiales
  - RF2.5 Eliminar asignación de recursos materiales
- RF3 Gestionar Vale de Salida
  - RF3.1 Adicionar vale de salida
  - RF3.2 Modificar vale de salida
  - RF3.3 Consultar vale de salida
  - RF3.4 Listar vales de salida
  - RF3.5 Eliminar vale de salida
- RF4 Gestionar solicitud de préstamo o ayuda económica
  - RF4.1 Adicionar solicitud de préstamo o ayuda económica
  - RF4.2 Modificar solicitud de préstamo o ayuda económica
  - RF4.3 Consultar solicitud de préstamo o ayuda económica
  - RF4.4 Listar solicitudes de préstamo o ayuda económica
  - RF4.5 Eliminar solicitud de préstamo o ayuda económica
  - RF4.6 Renovar Solicitud de Préstamo o Ayuda Económica
- RF5 Gestionar recurso material
  - RF5.1 Adicionar recurso material
  - RF5.2 Modificar recurso material
  - RF5.3 Consultar recurso material
  - RF5.4 Listar recursos materiales
  - RF5.5 Eliminar recurso material
- RF6 Gestionar bonificación de pasaje a estudiantes
  - RF6.1 Adicionar bonificación de pasaje a estudiantes

## *Capítulo 2 Características del módulo Económico del SAEF3*

- RF6.2 Modificar bonificación de pasaje a estudiantes
- RF6.3 Consultar bonificación de pasaje a estudiantes
- RF6.4 Listar bonificaciones de pasajes a estudiantes
- RF6.5 Eliminar bonificación de pasaje a estudiantes
- RF7 Gestionar partida de presupuesto
  - RF7.1 Adicionar partida de presupuesto
  - RF7.2 Modificar partida de presupuesto
  - RF7.3 Consultar partida de presupuesto
  - RF7.4 Listar partidas de presupuesto
  - RF7.5 Eliminar partida de presupuesto
- RF8 Gestionar plan del presupuesto
  - RF8.1 Adicionar plan del presupuesto
  - RF8.2 Modificar plan del presupuesto
  - RF8.3 Consultar plan del presupuesto
  - RF8.4 Listar planes del presupuesto
  - RF8.5 Eliminar plan del presupuesto
- RF9 Gestionar ejecución mensual del presupuesto para un centro de costo
  - RF9.1 Adicionar ejecución mensual del presupuesto para un centro de costo
  - RF9.2 Modificar ejecución mensual del presupuesto para un centro de costo
  - RF9.3 Consultar ejecución mensual del presupuesto para un centro de costo
  - RF9.4 Eliminar ejecución mensual del presupuesto para un centro de costo
- RF10 Gestionar centro de costo
  - RF10.1 Adicionar centro de costo
  - RF10.2 Modificar centro de costo
  - RF10.3 Consultar centro de costo
  - RF10.4 Listar centros de costo
  - RF10.5 Eliminar centro de costo
- RF11 Gestionar almacén
  - RF11.1 Adicionar almacén
  - RF11.2 Modificar almacén
  - RF11.3 Consultar almacén
  - RF11.4 Listar almacenes
  - RF11.5 Eliminar almacén
- RF12 Gestionar área
  - RF12.1 Adicionar área

## *Capítulo 2 Características del módulo Económico del SAEF3*

- RF12.2 Modificar área
- RF12.3 Consultar área
- RF12.4 Listar áreas
- RF12.5 Eliminar área
- RF13 Gestionar unidad de medida
  - RF13.1 Adicionar unidad de medida
  - RF13.2 Modificar unidad de medida
  - RF13.3 Consultar unidad de medida
  - RF13.4 Listar unidades de medida
  - RF13.5 Eliminar unidad de medida
- RF14 Gestionar asignatura
  - RF14.1 Adicionar asignatura
  - RF14.2 Modificar asignatura
  - RF14.3 Consultar asignatura
  - RF14.4 Listar asignaturas
  - RF14.5 Eliminar asignatura
- RF15 Generar reportes
  - RF15.1 Obtener cantidad de recursos materiales solicitados en un rango de fechas determinado
  - RF15.2 Obtener cantidad de solicitudes canceladas
  - RF15.3 Listar solicitudes aprobadas parcialmente
  - RF15.4 Obtener cantidad de un recurso material asignado por rango de fechas y por áreas
  - RF15.5 Obtener monto de recursos materiales abastecidos por rango de fechas y por almacén
  - RF15.6 Obtener monto de préstamo o ayuda económica de un estudiante por rango de fechas y año académico
  - RF15.7 Obtener monto total de préstamos o ayudas económicas por rango de fechas y año académico
  - RF15.8 Obtener histórico de bonificaciones de pasaje por estudiante en un rango de fechas
  - RF15.9 Obtener histórico de bonificaciones de pasaje en un rango de fechas
  - RF15.10 Obtener bonificaciones de pasaje por provincia y municipio
  - RF15.11 Obtener partidas sobregiradas por centro de costo
  - RF15.12 Obtener partidas ejecutadas parcialmente por centro de costo

## *Capítulo 2 Características del módulo Económico del SAEF3*

- RF15.13 Obtener histórico de partidas por rango de años (comparar consumo de partidas por año)

### **2.3.2 Requisitos no funcionales**

#### **Funcionalidad**

##### Seguridad

- El sistema SAEF3 maneja la seguridad de acceso y administración de usuarios mediante el otorgamiento de privilegios y roles.
- Todo uso de las funcionalidades del módulo requiere la autenticación de los usuarios.
- El módulo concederá acceso a cada usuario autenticado solo a las funciones que le estén permitidas, de acuerdo a la configuración del sistema.

#### **Eficiencia**

##### Utilización de recursos

- El módulo interactuará con impresoras para imprimir los diferentes documentos que genere la aplicación como respuesta a las funcionalidades del sistema.

#### **Portabilidad**

##### Adaptabilidad

- El módulo permitirá la manipulación de dispositivos externos como las impresoras.

##### Instalabilidad

- El sistema podrá ser instalado en el ambiente especificado en los requisitos tecnológicos para servidores.

#### **Hardware**

##### Servidor Aplicación

- Procesador: 3.00 GHZ
- RAM: 1GB
- Disco duro: 160 GB
- Lector de CD: 1
- Tarjeta de Red: 1

##### Servidor de Base de Datos

- RAM: 1GB
- Disco duro: 160 GB

## Capítulo 2 Características del módulo Económico del SAEF3

### PC Cliente

- Procesador: 1.40 GHZ
- RAM: 512 Mb
- Tarjeta de Red: 1

### 2.3.4 Prototipos de interfaz de usuario

#### RF Asignar recursos materiales

Área: ---Seleccione área--- Receptor:

Fecha de Recepción: dd/mm/aa

Seleccione los materiales a asignar:

|                                     | Material   | Cantidad |   |
|-------------------------------------|------------|----------|---|
| <input checked="" type="checkbox"/> | Material 1 | XXX      | ^ |
| <input checked="" type="checkbox"/> | Material 2 | XXY      |   |
| <input type="checkbox"/>            | Material 3 | XYY      |   |
| <input checked="" type="checkbox"/> | Material 4 | YYY      |   |
| <input type="checkbox"/>            | Material 5 | YYX      | v |

Aceptar Cancelar

Figura 6. Prototipo de interfaz de usuario "Asignar recursos materiales" 1.

Fuente: (Elaboración propia)

Área: Área X Seleccionada

| No. Referencia | Recurso Material | Cantidad                       | Receptor                     | Fecha    |
|----------------|------------------|--------------------------------|------------------------------|----------|
| YYHHH          | Material 1       | <input type="text" value="0"/> | Nombre y Apellido1 Apellido2 | dd/mm/aa |
| YYYHH          | Material 2       | <input type="text" value="0"/> | Nombre y Apellido1 Apellido2 | dd/mm/aa |
| YYYYH          | Material 4       | <input type="text" value="0"/> | Nombre y Apellido1 Apellido2 | dd/mm/aa |
|                |                  |                                |                              |          |

+ Aceptar Cancelar

Figura 7. Prototipo de interfaz de usuario "Asignar recursos materiales" 2.

Fuente: (Elaboración propia)

**RF Adicionar Vale de Salida**

Destino:  No. de Vale:  Almacén:  Referencia:

Presenta costo en CUC  Presenta costo en MN Fecha:

| Código                           | Lote                           | Descripción                           | UM                            | Cantidad                         | Costo CUC                         | Importe CUC | Costo MN                          | Importe MN | PV CUC                            | PV MN                             |
|----------------------------------|--------------------------------|---------------------------------------|-------------------------------|----------------------------------|-----------------------------------|-------------|-----------------------------------|------------|-----------------------------------|-----------------------------------|
| <input type="text" value="000"/> | <input type="text" value="0"/> | <input type="text" value="Material"/> | <input type="text" value=""/> | <input type="text" value="000"/> | <input type="text" value="0.00"/> |             | <input type="text" value="0.00"/> |            | <input type="text" value="0.00"/> | <input type="text" value="0.00"/> |
| <input type="text" value="000"/> | <input type="text" value="0"/> | <input type="text" value="Material"/> | <input type="text" value=""/> | <input type="text" value="000"/> | <input type="text" value="0.00"/> |             | <input type="text" value="0.00"/> |            | <input type="text" value="0.00"/> | <input type="text" value="0.00"/> |
| <input type="text" value="000"/> | <input type="text" value="0"/> | <input type="text" value="Material"/> | <input type="text" value=""/> | <input type="text" value="000"/> | <input type="text" value="0.00"/> |             | <input type="text" value="0.00"/> |            | <input type="text" value="0.00"/> | <input type="text" value="0.00"/> |
| <input type="text" value="000"/> | <input type="text" value="0"/> | <input type="text" value="Material"/> | <input type="text" value=""/> | <input type="text" value="000"/> | <input type="text" value="0.00"/> |             | <input type="text" value="0.00"/> |            | <input type="text" value="0.00"/> | <input type="text" value="0.00"/> |

Total: \$0.00 \$0.00

PV: Precio Venta

Figura 8. Prototipo de interfaz de usuario “Adicionar Vale de salida”

Fuente: (Elaboración propia)

**RF Adicionar Bonificación de pasaje a estudiantes**

Estudiante:  CI:

Origen Destino

Fecha de ida:  Provincia:

Municipio:

Fecha de vuelta:  Provincia:

Municipio:

Persona que autoriza:

Figura 9. Prototipo de interfaz de usuario “Adicionar Bonificación de pasaje”.

Fuente: (Elaboración propia)

## Capítulo 2 Características del módulo Económico del SAEF3

### 2.3.5 Validaciones de requisitos

Para realizar la validación de los requisitos se tuvo en cuenta no solo el criterio de los desarrolladores, sino también el criterio del cliente y de sus homólogos, para un total de 9 personas. Los resultados obtenidos se muestran a continuación.

| Atributos de calidad | Resultado                                      | Interpretación   |
|----------------------|--|--|
| No ambiguo           | $Q_1 = \frac{n_{ui}}{n_r} = \frac{94}{94} = 1$ | Todos los requisitos tienen una única interpretación (no ambiguos)   |
| Comprensible         | $Q_2 = \frac{n_{ur}}{n_r} = \frac{94}{94} = 1$ | Todos los requisitos son comprendidos (comprensibles)  |
| Correcto             | $Q_3 = \frac{n_c}{n_r} = \frac{94}{94} = 1$    | Todos los requisitos representan una necesidad de función del sistema a construir (correctos)  |
| Posible de trazar    | $Q_4 = \frac{n_{pt}}{n_r} = \frac{94}{94} = 1$ | Todos los requisitos están escritos de manera que facilite la referencia de cada requisito individualmente a los requisitos con los cuales se relaciona (posibles de trazar) |

Tabla 7. Aplicación de las métricas de validación de requisitos

Fuente: (Elaboración propia)

A partir de las métricas de validación de requisitos se puede constatar que existe un 100% de concordancia entre el personal que participó en el desarrollo de los requisitos, respecto al cumplimiento de los atributos de calidad de los mismos, obteniéndose que en su totalidad son no ambiguos, correctos, comprensibles y posibles de trazar.

### 2.4 Diseño de la solución

En el presente epígrafe se especifican los patrones del diseño aplicados a la solución, el diagrama de clases del diseño y el modelo de datos obtenido.

#### 2.4.1 Patrones de diseño

A continuación se muestran evidencias de la aplicación de los patrones de diseño.

## Capítulo 2 Características del módulo Económico del SAEF3

### Patrones GRASP

En la figura 10 se evidencia la utilización de los patrones de asignación de responsabilidades, como es el caso del experto en la clase *CE\_ReintegroPasaje* y *EReintegroPasajeRepository*, las mismas contienen toda la información y la lógica del negocio que comprenden a ese proceso. El patrón controlador y creador se refleja en la clase *CC\_ReintegroPasaje*, ya que es la encargada de atender todas las peticiones de las vistas y pasar los datos a las clases del modelo, así como la responsabilidad de identificar la creación de los nuevos objetos por las clases que contienen la información necesaria para realizarla. En todas las clases de la lógica del negocio se muestra una alta cohesión y un bajo acoplamiento ya que cada una de ellas posee el trabajo de realizar las labores que solo le competen a ella sin asociaciones con la vista, favoreciendo que la dependencia sea baja

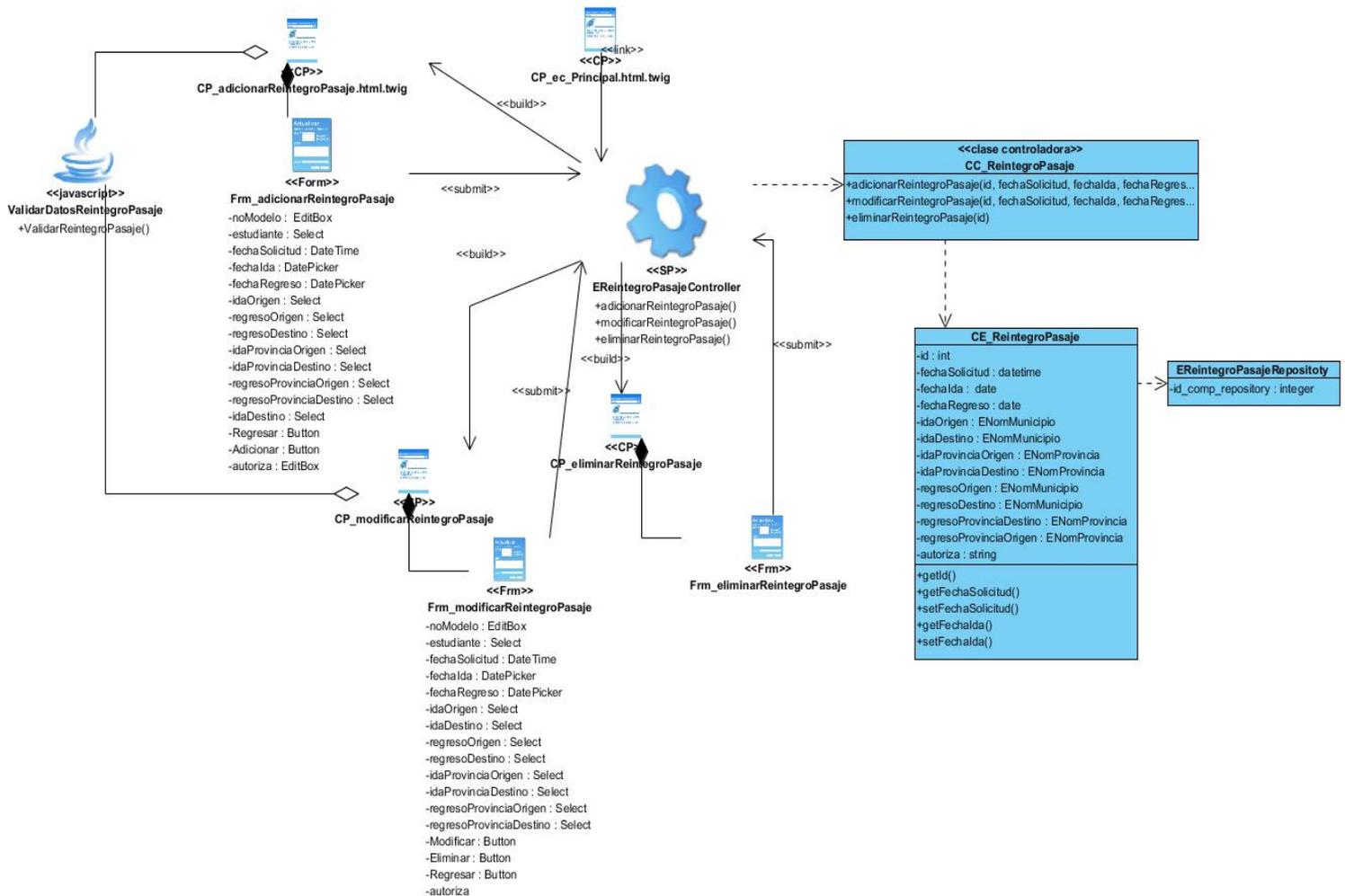


Figura 10. Diagrama de clases del diseño de Bonificación de pasajes a estudiantes

Fuente: (Elaboración propia)

### Patrones GoF

#### Estructurales: Adaptador

La vista del módulo implementa el uso de plantillas Twig, las cuales contienen variables o expresiones que se reemplazan por valores cuando se evalúa la plantilla, y las etiquetas controlan la lógica de la plantilla. A continuación se ilustran ejemplos de su utilización:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8"/>
  <title>{% block title %}SAEF3|Control Interno{% endblock %}</title>

  {% block stylesheets ...%}
</head>
<body>
  {% block body %}{% endblock %}

  {% block javascripts ...%}
</body>
</html>
```

Figura 11. Clase *baseec.html.twig*

Fuente: (Elaboración propia)

La herencia de plantillas permite crear un “esqueleto” de plantilla base (en el módulo esta plantilla esqueleto es *baseec.html.twig*) que contiene todos los elementos comunes del módulo y define bloques que las plantillas descendientes pueden sustituir; tal es el caso de los bloques definidos dentro de las etiqueta {% **block** ... %} ... {% **endblock** %}.

```
{% extends ':ec:baseec' %}
{% block title %}SAEF3|Módulo Económico{% endblock %}

{% block body %}

  <div id="wrapper">

    <!-- Sidebar -->
    <nav class="navbar navbar-inverse navbar-fixed-top" role="navigation"...>

    <div id="page-wrapper"...>
    <!-- /#page-wrapper -->

  </div><!-- /#wrapper -->
{% endblock %}

{% block javascripts ...%}
```

Figura 12. Clase *layout.html.twig*

Fuente: (Elaboración propia)

## Capítulo 2 Características del módulo Económico del SAEF3

La clase *layout.html.twig*, como se muestra en la figura 12, extiende de la clase *baseec.html.twig*, redefiniendo los bloques *title*, *body* y *javascripts*.

```
{% extends ':ec:layout' %}

{% block titulo_page %}Bonificaciones de Pasaje{% endblock %}
{% block migajas %}
<b>LISTADO DE SOLICITUDES DE BONIFICACIONES DE PASAJES</b>
{% endblock %}

{% block contenido %}
<div class="table-responsive"...>
<a class="btn btn-success pull-right" href="{{ path('reintegropasaje_new') }}"...>
<a class="btn btn-success pull-right" href="{{ path('ec_portada') }}"...>
{% endblock %}
{% block javascripts %}
{{ parent() }}
<script type="text/javascript"...>
{% endblock %}
```

Figura 13. Clase *EReintegroPasaje/index.html.twig*

Fuente: (Elaboración propia)

La clase *EReintegroPasaje/index.html.twig* (Figura 13) extiende de *layout.html.twig*, sin redefinir el *title*, se define un bloque *titulo\_page* y un bloque *migajas* y se redefine el bloque *contenido* así como el bloque *javascripts*. En el resto de las clases que extienden de *EReintegroPasaje/index.html.twig* no se redefine el bloque *titulo\_page* para que el usuario sepa en todo momento que está trabajando con las “Bonificaciones de pasaje”, como se muestra en la figura 14.

```
{% extends 'ec:EReintegracionPasaje/index' %}

{% block migajas %} <b>ADICIONAR SOLICITUD DE BONIFICACIÓN DE PASAJE</b> {% endblock %}
{% block contenido %}

    {% if error is defined ...%}

    {{ form_start(form) }}
    <div class="row">
        <div class="form-group col-lg-2">...</div>
        <div class="form-group col-lg-3">...</div>
    </div>

    <div class="row" name="lineaTitulo">
        <div class="form-group col-lg-3">...</div>
        <div class="form-group col-lg-4 col-lg-offset-3">...</div>
    </div>

    <div class="row" name="lineaFechaProv">...</div>

    <div class="row" name="municipiosIda">...</div>
    <div class="row">...</div>
    {{ form_end(form) }}
    <a class="btn btn-success pull-right" href="{{ path('reintegracionpasaje') }}">...</a>
{% endblock %}

{% block javascripts %}
    {{ parent() }}
    <script type="text/javascript">...</script>
{% endblock %}
```

Figura 14. Clase *EReintegracionPasaje/new.html.twig*

Fuente: (Elaboración propia)

### Creacional: Inicialización vaga

En la figura 15 se muestra la aplicación del patrón inicialización vaga al delegar la creación de los objetos *almacen*, *solicitud* y *partida*, así como el cálculo de sus valores hasta que sean realmente necesitados.

```
public function __construct()
{
    $this->almacen = new \Doctrine\Common\Collections\ArrayCollection();
    $this->solicitud = new \Doctrine\Common\Collections\ArrayCollection();
    $this->partida = new \Doctrine\Common\Collections\ArrayCollection();
}

public function addPartida(\AppBundle\Entity\ENomPartida $partida)
{
    $this->partida[] = $partida;

    return $this;
}

/** Remove partida ... */
public function removePartida(\AppBundle\Entity\ENomPartida $partida)
{
    $this->partida->removeElement($partida);
}

/** Get partida ... */
public function getPartida()
{
    return $this->partida;
}

/** Add almacen ... */
public function addAlmacen(\AppBundle\Entity\ENomAlmacen $almacen)
{
    $this->almacen[] = $almacen;

    return $this;
}
```

Figura 15. Ejemplo de aplicación del patrón de inicialización vaga

Fuente: (Elaboración propia)

### 2.4.2 Modelo de datos

El modelado y diseño de la base de datos tiene como objetivo generar un conjunto de entidades y las relaciones entre ellas, que permitan el almacenamiento de la información con un mínimo de redundancia, manteniendo su integridad y facilitando la recuperación para su consulta. Para ello se utilizó el Modelo Entidad-Relación (MER) compuesto por 38 tablas, de ellas 13 nomencladores y el resto destinadas al almacenamiento de la información correspondiente al dominio del problema. A continuación se muestra dicho diagrama.

## Capítulo 2 Características del módulo Económico del SAEF3

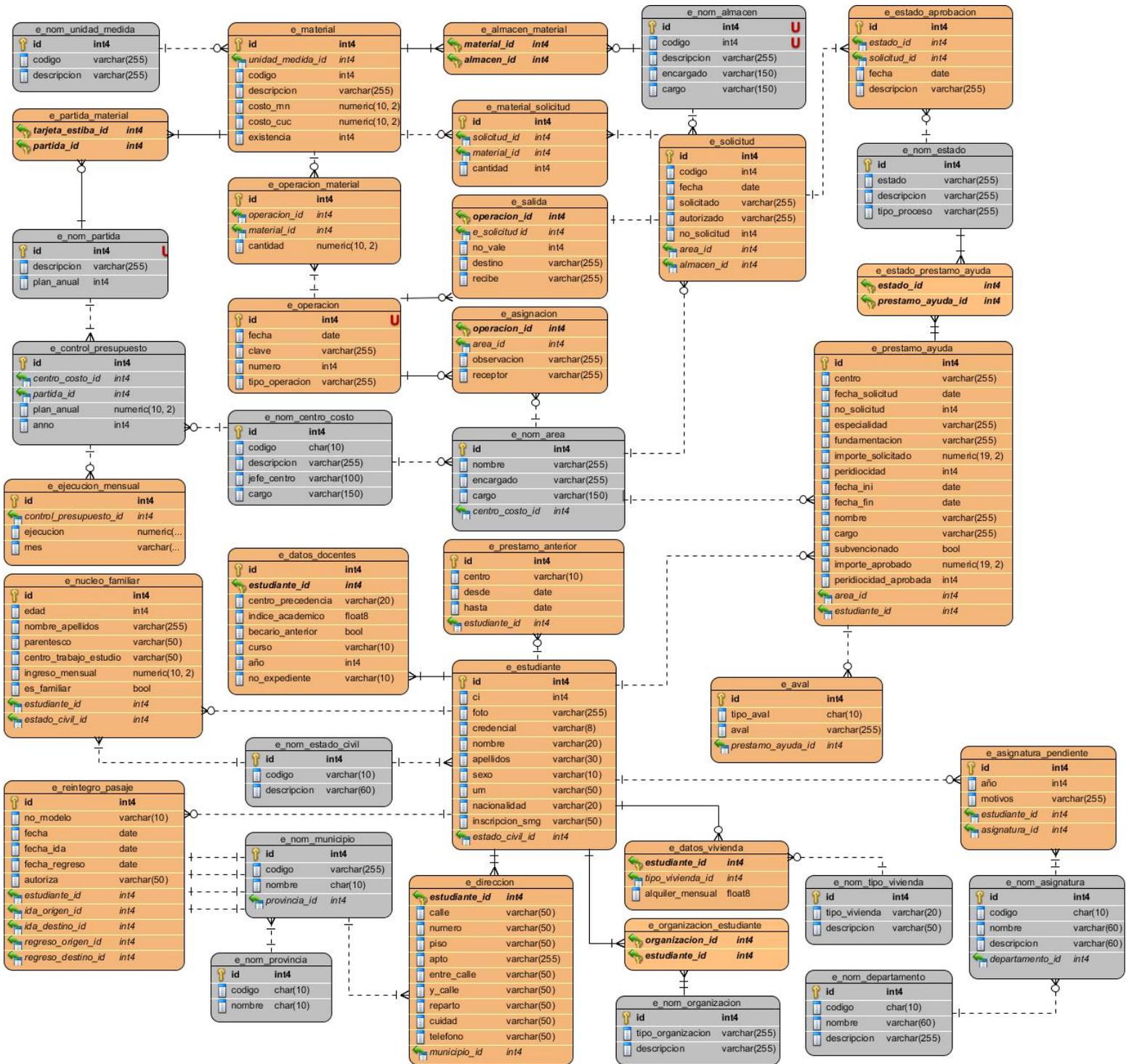


Figura 13. Diagrama MER. Modelo de datos

Fuente: (Elaboración propia)

### 2.4.3 Verificación del diseño

Para verificar la calidad del diseño de la solución se emplearon las métricas de RC, TC y CCM aplicadas a los indicadores: responsabilidad, complejidad de implementación y reutilización, clasificándolos en altos, medios o bajos (ver Anexo #3).

## Capítulo 2 Características del módulo Económico del SAEF3

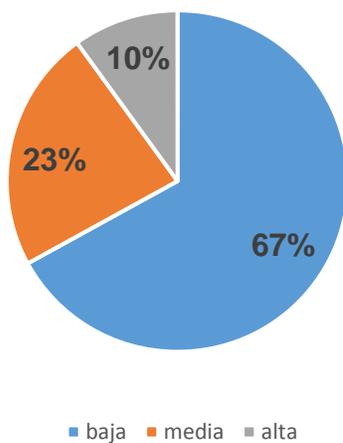
| Métrica | Resultados del análisis   |
|---------|---|
| RC      | La aplicación de la métrica arroja resultados favorables en cuanto a acoplamiento y complejidad de implementación con valores por encima del 70%, en el caso de la reutilización se alcanzó un 70% y la cantidad de prueba se obtuvo un 75% lo que conlleva a la conclusión de que la calidad del diseño escogido es aceptable.   |
| CCM     | Luego de revisar detenidamente cada diagrama de diseño se pudo constatar que en la mayoría de las clases, a lo sumo existe un método que accede a un mismo atributo. Esto arroja un valor CCM =2. Exceptuando la clase <i>EPrestamoAyuda</i> donde existen 7, para un CCM = 7. Se llega a la conclusión de que las clases del diseño poseen un alto grado de cohesión debido a que de un total de 30, solo una posee valor medio y el resto posee valor bastante pequeño. |
| TC      | Luego de la aplicación de la métrica los resultados obtenidos fueron satisfactorios pues más del 60% de las clases presentó baja responsabilidad y complejidad de implementación y solo el 10% presentó bajo grado de reutilización de un total de 30 clases.   |

**Tabla 8. Resultado de la aplicación de las métricas del diseño**

Fuente: (Elaboración propia)

En las gráficas que se muestran a continuación se visualizan los resultados obtenidos.

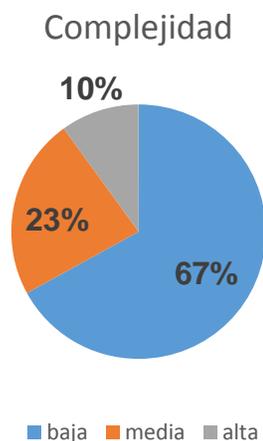
### Responsabilidad



**Figura 16. Comportamiento del indicador responsabilidad**

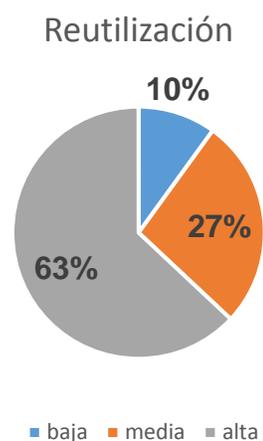
Fuente: (Elaboración propia)

## Capítulo 2 Características del módulo Económico del SAEF3



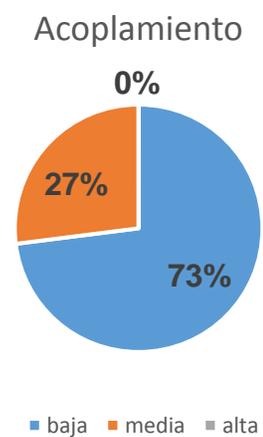
**Figura 17. Comportamiento del indicador complejidad**

Fuente: (Elaboración propia)



**Figura 18. Comportamiento del indicador reutilización**

Fuente: (Elaboración propia)



**Figura 19. Comportamiento del indicador acoplamiento**

Fuente: (Elaboración propia)

Capítulo 2 Características del módulo Económico del SAEF3

Complejidad de mantenimiento

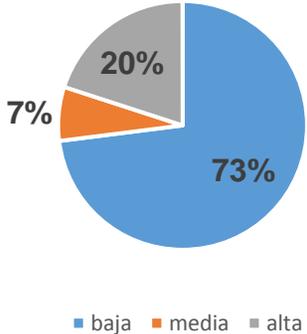


Figura 20. Comportamiento del indicador complejidad de mantenimiento

Fuente: (Elaboración propia)

Reutilización

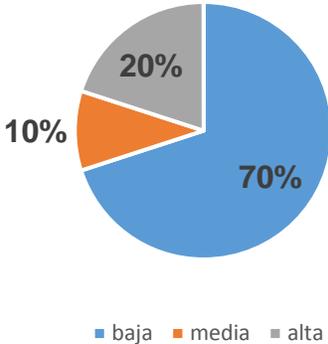


Figura 21. Comportamiento del indicador reutilización

Fuente: (Elaboración propia)

Cantidad de Pruebas

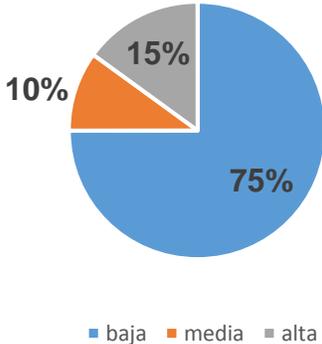


Figura 22. Comportamiento del indicador cantidad de pruebas

Fuente: (Elaboración propia)

## *Capítulo 2 Características del módulo Económico del SAEF3*

### **2.5 Conclusiones parciales**

- En el presente capítulo se describieron las características del sistema a partir de un modelado del negocio previo, para ello se identificaron requisitos funcionales y no funcionales, se realizó la especificación de los funcionales, los cuales fueron validados a partir de la aplicación de las métricas: no ambiguo, comprensible, correcto y traceable. Se realizó además el modelado del diseño, el cual fue verificado mediante la aplicación de los patrones TC, RC, CCM.
- Como resultado de lo anteriormente planteado se corroboró la obtención de un diseño que presenta un 100% de concordancia entre el personal que participó en el desarrollo de los requisitos, respecto al cumplimiento de los atributos de calidad de los mismos. Por otra parte, se evidenció que en el diseño existe baja responsabilidad y complejidad, lo cual favorece la reutilización, la alta cohesión y el bajo acoplamiento, pudiéndose comprobar una correcta asignación de responsabilidades.

## CAPÍTULO 3 IMPLEMENTACIÓN Y ANÁLISIS DE LOS RESULTADOS

### 3.1 Introducción

En el presente capítulo se muestra el diagrama de componentes que estructura la implementación, se describen los estándares de codificación y el tratamiento de errores empleados, finalizando con el análisis de los resultados obtenidos en la validación de la solución y las variables de la investigación.

### 3.2 Aspectos relevantes de la implementación

A continuación se muestra el diagrama de componentes desarrollado, los estándares de codificación empleados y el tratamiento de errores.

#### 3.2.1 Diagrama de componentes

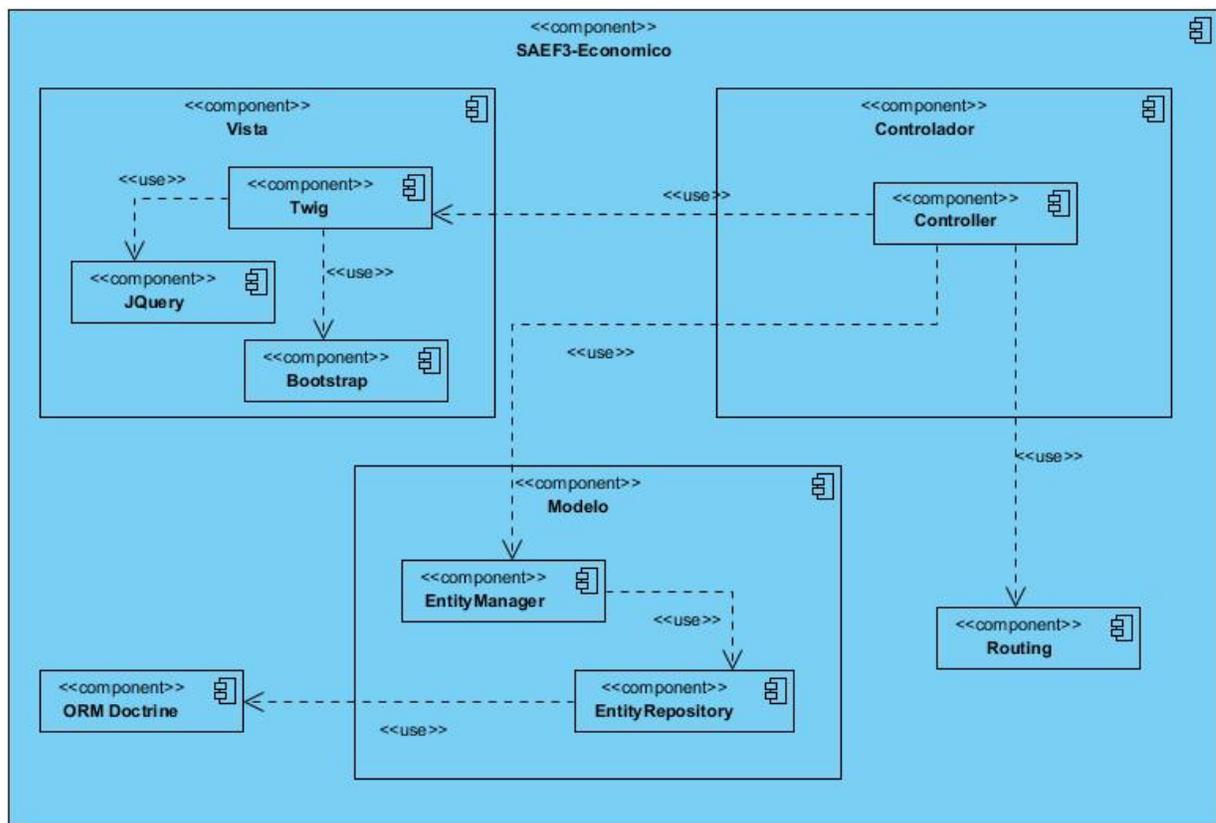


Figura 23. Diagrama de componentes del módulo Económico del SAEF3

Fuente: (Elaboración propia)

#### 3.2.2 Estándares de codificación

Symfony 2.6, el framework **empleado** para el desarrollo del módulo Económico, presenta estándares de codificación definidos en los documentos PSR-0 (htt1), PSR-1 (htt2) y PSR-2 (htt3), definiéndolos de la siguiente manera:

## Capítulo 3 Implementación y análisis de los resultados

### Estructura

- Añadir un solo espacio después de cada delimitador coma; una coma después de cada elemento del arreglo en un arreglo multilínea, incluso después del último; una línea en blanco antes de las declaraciones *return*, a menos que el valor devuelto solo sea dentro de un grupo de declaraciones (tal como una declaración *if*);

```
public function editAction($id)
{
    $em = $this->getDoctrine()->getManager();
    $errores = array();

    $entity = $em->getRepository('AppBundle:EReintegracionPasaje')->find($id);

    if (!$entity) {
        $error = "La bonificación de pasaje no existe";
        $errores[] = $error;
    }

    if (empty($errores)) {
        $editForm = $this->createForm($entity);
        $deleteForm = $this->createForm($id);

        return $this->render('ec/EReintegracionPasaje/edit', array(
            'entity' => $entity,
            'edit_form' => $editForm->createView(),
            'delete_form' => $deleteForm->createView(),
            'error' => $errores,
        ));
    }
}
```

Figura 24. Screenshot al código de EReintegracionPasajeController.php

Fuente: (Elaboración propia)

- Usar llaves para indicar la estructura del cuerpo de control, independientemente del número de declaraciones que contenga: `{% block body %} ... {% endblock %}`

```
{% extends 'ec:EReintegracionPasaje/index' %}

{% block migajas %} <b>SOLICITUD DE BONIFICACIÓN DE PASAJE {{ entity.noModelo }}</b> {% endblock %}
){% block contenido %}

1  {% if error is defined ...%}

2  <a href="{{ path('ec_pdf_prueba', {'id': entity.id}) }}" class="pull-right"><i class="icon-file-alt"></i> Exportar a
3  pdf</a><br/>

4  <table class="table table-bordered table-hover tablesorter"...>
5  <a href="{{ path('reintegracionpasaje edit', {'id':entity.id}) }}" class="btn btn-success pull-right"><i
6  class="icon-edit"></i> <b>Modificar</b></a>

7  <a href="{{ path('reintegracionpasaje eliminar', {'id':entity.id}) }}" class="btn btn-success pull-right">
8  <i class="icon-trash"></i> <b>Eliminar</b></a>

9  <a class="btn btn-success pull-right" href="{{ path('reintegracionpasaje') }}"
10 >
11 <i class="icon-reply"></i> <b>Regresar</b>
12 </a>

){% endblock %}
```

Figura 25. Screenshot al código de la interfaz de modificar EReintegracionPasaje

Fuente: (Elaboración propia)

## Capítulo 3 Implementación y análisis de los resultados

- Declarar las propiedades de clase antes que los métodos; utilizar paréntesis cuando se instancia una clase sin importar el número de argumentos que tenga el constructor.

### Convenciones de nomenclatura

- Utiliza mayúsculas intercaladas —sin guiones bajos— en nombres de variable, función, método o argumentos;

```
private function createEditForm(EReintegroPasaje $entity)
{
    $form = $this->createForm(new EReintegroPasajeType(), $entity, array(
        'action' => $this->generateUrl('reintegropasaje_update', array('id' => $entity->getId())),
        'method' => 'PUT',
    ));

    $form->add('submit', 'submit', array('label' => 'Modificar',
        'attr' => array('class' => "btn btn-success pull-right",
    )));

    return $form;
}
```

Figura 26. Screenshot a la controladora de EReintegroPasaje

Fuente: (Elaboración propia)

Usa guiones bajos para nombres de opción y nombres de parámetro; utiliza espacios de nombres para todas las clases; prefija las clases abstractas con *Abstract*. Es importante tener en cuenta que algunas de las primeras clases de Symfony2 no siguen esta convención y no se han rebautizado por razones de compatibilidad hacia atrás. No obstante, todas las nuevas clases abstractas tienen que seguir esta convención de nomenclatura; sufija las interfaces con *Interface*; sufija las características con *Trait*; sufija las excepciones con *Exception*; utiliza caracteres alfanuméricos y guiones bajos para los nombres de archivo.

### Documentación

Añade bloques PHPDoc a todas las clases, métodos y funciones; omite la etiqueta *@return* si el método no devuelve nada; las anotaciones *@package* y *@subpackage* no se utilizan.

### Licencia

Symfony se distribuye bajo la licencia MIT, y el bloque de la licencia tiene que estar presente en la parte superior de todos los archivos PHP, antes del espacio de nombres.

#### 3.2.3 Tratamiento de errores

El tratamiento de errores producidos por el sistema o generados por el usuario es un tema importante en el desarrollo del software. Las validaciones del negocio son las que controlan el flujo de los datos que llega al controlador para evitar la incongruencia de la información almacenada en la base de datos.

## Capítulo 3 Implementación y análisis de los resultados

Las validaciones realizadas en las vistas aseguran la entrada correcta de los datos a la aplicación. Su principal misión es evitar que datos incorrectos sean introducidos en los diferentes campos de los formularios y que datos inconsistentes o incorrectos lleguen al servidor. Para la notificación de errores en las interfaces del módulo se utilizan las validaciones que provee la librería jQuery. Por otro lado el marco de trabajo Symfony contiene un conjunto de procedimientos que garantizan validaciones básicas a disponibilidad del usuario.

### 3.3 Validación de la solución

A continuación se muestra los resultados obtenidos de la aplicación de las pruebas aplicadas al módulo.

#### 3.3.1 Pruebas de caja negra

Las pruebas de caja negra se basan en las especificaciones del módulo a ser probadas para elaborar los casos de prueba. Su comportamiento se evalúa mediante sus entradas y las salidas obtenidas a partir de ellas. Sin embargo, el estudio de todas las posibles entradas y salidas sería impracticable, es por ello que se selecciona un conjunto sobre las que se realizan las pruebas (Juristo, y otros, 2004). Para la aplicación de esta prueba se seleccionó la técnica de Particiones de Equivalencia.

Durante la aplicación de las pruebas se detectaron siete no conformidades en la primera iteración, que fueron solucionadas, en una segunda iteración se detectaron cinco no conformidades, las cuales fueron corregidas y posteriormente en una tercera iteración no se identificaron no conformidades.



Tabla 9. Resultado de la aplicación de las pruebas unitarias

A continuación se muestra un ejemplo de caso de prueba efectuado al requisito Adicionar Bonificación de pasaje a estudiante.

## Capítulo 3 Implementación y análisis de los resultados

| Escenario                                      | Descripción                                 | Nombre estudiante      | No Modelo  | Fecha      | Fecha ida  | Fecha regreso | Ida origen | Ida destino | Regreso origen             | Regreso destino | Autoriza                   | Respuesta del sistema   |
|--|---|------------------------|------------|------------|------------|---------------|------------|-------------|----------------------------|-----------------|----------------------------|---|
| <b>EC 1.1 Adicionar bonificación de pasaje</b> | Se inserta una nueva bonificación de pasaje | NA                     | NA         | NA         | NA         | NA            | NA         | NA          | NA                         | NA              | NA                         | El sistema muestra un mensaje indicando que debe llenar los campos                                |
|  |   | vacío                  | vacío      | vacío      | vacío      | vacío         | vacío      | vacío       | vacío                      | vacío           | vacío                      |   |
|  |   | V                      | V          | NA         | V          | V             | V          | NA          | NA                         | V               | V                          | El sistema muestra un mensaje indicando que debe llenar los campos                                |
|  |   | Elizabeth Guerra Pompa | 0003041    | vacío      | 05/05/2015 | 10/05/2015    | La Habana  | vacío       | vacío                      | La Habana       | Ana Marys Garcia Rodríguez |   |
|  |   | V                      | V          |            | V          | V             | V          | V           | V                          | V               | V                          | El sistema inserta la nueva bonificación y se muestra para impresión                              |
|  |   | Elizabeth Guerra Pompa | 0003071    | 12/05/2015 | 05/05/2015 | 10/05/2015    | La Habana  | Granma      | Granma                     | La Habana       | Ana Marys Garcia Rodríguez |   |
|  |   | V                      | V          |            | V          | V             | V          | V           | V                          | V               | V                          | El sistema muestra un mensaje indicando que ese estudiante ya recibió una bonificación en ese mes |
| Elizabeth Guerra Pompa                         | 0003091                                     |                        | 20/05/2015 | 28/05/2015 | La Habana  | Granma        | Granma     | La Habana   | Ana Marys Garcia Rodríguez |                 |                            |   |

## Capítulo 3 Implementación y análisis de los resultados

### 3.3.2 Tipos de pruebas

Se aplicaron tres tipos de pruebas:

- Funcionalidad: esta prueba fue aplicada en todos los niveles de prueba mediante el uso del método de caja negra.
- Portabilidad: el módulo fue probado en diferentes plataformas (Nova 4, Windows 7 y 8.1), mostrando un desempeño aceptable.
- Usabilidad: la clienta y sus homólogos le aplicaron pruebas al módulo, a partir de estas pruebas que se realizaron mostraron su satisfacción con respecto al producto, la cual se refleja en los resultados de la aplicación de la técnica ladov.

### 3.4 Validación de las variables de la investigación

Para realizar la validación de las variables de la investigación se analizaron las dimensiones de las mismas, como se muestra en la tabla 10.

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.es, 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. (Haimann, 2000)

| 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 encuentran disponibles en el sistema, tanto durante la ejecución de los procesos como una vez culminado los mismos.   |

### Capítulo 3 Implementación y análisis de los resultados

|                           |                         |   |  |
|---------------------------|-------------------------|---|--|
|                           |                         | 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 en el sistema, en cualquier momento.  |
| Control de la información | Supervisión             | Las cifras con las que se trabaja en el VDEA contienen referencias a otros documentos declarados por las áreas o generados por otros procesos, esta información puede sufrir alteraciones debido a errores humanos. | Las cifras se actualizan a medida que se van introduciendo datos en el sistema, validándose que no existan errores de cálculo.   |
|                           |                         | Es engorroso supervisar eficientemente a las áreas subordinadas considerando que son varias y las supervisiones se realizan manualmente.  | Con el módulo desarrollado se facilita la 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 error en la información.   | Los reportes son generados por el sistema mediante llamadas a la base de datos que permiten obtener la información necesaria.  |
|                           | Gestión de desviaciones | Para el control del presupuesto, actualmente se utiliza la herramienta Excel y calculadoras para verificar las partidas inejecutadas y sobregiradas, y en base a ello tratar de corregir esas desviaciones.         | Mediante la generación de reportes que realizan los cálculos pertinentes se facilita la detección de partidas inejecutadas y sobregiradas, permitiendo una temprana corrección de la desviaciones. |

**Tabla 10. Validación de las variables de la investigación**

Fuente: (Elaboración propia)

## Capítulo 3 Implementación y análisis de los resultados

Para valorar el grado de satisfacción del cliente con la solución desarrollada respecto al control y disponibilidad de la información, se aplicó la técnica ladov que permite el estudio del grado de satisfacción del personal involucrado en un proceso objeto de análisis.

Durante la valoración fungieron como clientes, los vicedecanos de economía y administración de todas las facultades de la UCI, para un total de siete personas. Como resultado se obtuvo:

| ¿Le gustaría hacer uso del módulo Económico del SAEF3 propuesto, para desarrollar los procesos asociados al área económica del VDEA? | ¿Considera usted oportuno continuar ejecutando los procesos asociados al área económica del VDEA manualmente, a pesar del gran volumen de información que se genera? |       |    |       |       |    |    |       |    |
|--|--|-------|----|-------|-------|----|----|-------|----|
|  | No   |       |    | No sé |       |    | Sí |       |    |
|  | ¿El módulo Económico 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 11. Cuadro lógico de ladov.

Fuente: (Elaboración propia)

$$ISG = \frac{5*1+2*0,5}{7} \quad (5)$$

De manera que el ISG = 0,86

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

| Nivel de satisfacción           | Cantidad | %     |
|---------------------------------|----------|-------|
| Máxima satisfacción             | 6        | 85,71 |
| Más satisfecho que insatisfecho | 1        | 7,14  |
| No definida                     | 0        | 0     |

Tabla 12. Resultado de aplicación de la técnica ladov

Fuente: (Elaboración propia)

## *Capítulo 3 Implementación y análisis de los resultados*

Al procesar las respuestas a las encuestas en el cuadro lógico de ladov, se obtiene un grado de satisfacción grupal de 0,86, lo cual se traduce en una clara satisfacción con el uso del módulo Económico 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 Económico del SAEF3 para desarrollar los procesos asociados al área económica del VDEA. Mientras que un 7,14% no sabe si sería oportuno continuar ejecutando los procesos asociados al área económica del VDEA manualmente, a pesar del gran volumen de información que se genera.

Las preguntas abiertas que se formularon fueron:

- ¿Qué valoraciones le sugiere el módulo 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 sistema permite un control más eficiente de los procesos asociados al área económica de las facultades.
- Facilita la obtención de reportes para una gestión proactiva desde el VDEA.
- Favorece la privacidad hacia los estudiantes en un tema delicado como lo es el proceso de solicitud de préstamos o ayudas económicas.
- Permite establecer un mejor control de la ejecución del presupuesto y la visualización del mismo por parte de cada jefe de área.
- El proceso de asignación de los recursos materiales favorece un mejor control sobre el destino final, tema recurrente en las auditorías del Gasto Público.

También se consideró como mejora del módulo, incluir el control de dietas tanto de la facultad como de sus centros de desarrollo.

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.

### **3.4 Conclusiones parciales**

- Los estándares de codificación empleados en la implementación permitieron ganar en legibilidad, claridad y mayor entendimiento del código por parte de los desarrolladores. El tratamiento de errores posibilitó evitar que datos incorrectos fuesen introducidos en los diferentes campos de los formularios mediante el uso de determinados métodos de validación.

### *Capítulo 3 Implementación y análisis de los resultados*

- Se aplicaron pruebas de caja negra para verificar el correcto funcionamiento del software que brindaron los siguientes resultados: en la primera iteración se detectaron seis no conformidades, las cuales fueron solucionadas y posteriormente, en una segunda iteración, no se detectaron no conformidades. También, se realizó la validación de las variables de la investigación mediante el análisis del comportamiento de las dimensiones de las mismas antes y después del desarrollo del producto y la aplicación de la técnica ladov que permitió medir la satisfacción del cliente respecto al control y disponibilidad de la información con el módulo Económico del SAEF3, donde se evidenció un índice de satisfacción grupal de 0,86 lo cual se traduce en una clara satisfacción por parte del cliente; además existió un 100% de concordancia de los participantes, en que la solución contribuye a elevar el control y disponibilidad de la información de los procesos económicos, y que les gustaría utilizar la solución obtenida para la gestión de dichos procesos en sus áreas.

### **CONCLUSIONES GENERALES**

- El estudio de sistemas que manejan la gestión económica en las universidades evidenció que los mismos no brindan una solución completa al problema existente en el área económica de la Facultad 3, lo cual constituyó motivación para el desarrollo de la presente investigación, que utiliza como framework a Symfony 2.5.6, empleando la metodología de desarrollo aprobada en el plan de mejoras de la UCI, garantizando un ambiente visual ameno y organizado con el empleo de Bootstrap 3 y jQuery 1.9, entre otras herramientas.
- Se describieron las características del módulo a partir de un modelado del negocio previo, para ello se identificaron requisitos funcionales y no funcionales, se realizó la especificación de los funcionales, los cuales fueron validados y el resultado de dicha validación constató que los mismos en su totalidad no son ambiguos, son correctos, comprensibles y posibles de trazar.
- La verificación del diseño realizado, mediante la aplicación de las métricas TC, RC y CCM, evidenció una baja responsabilidad y complejidad de implementación, la cual favoreció la reutilización corroborándose así una correcta asignación de responsabilidades.
- La aplicación de los estándares de codificación permitió una implementación responsable y organizada facilitando la obtención del módulo Económico de SAEF3.
- Para verificar el correcto funcionamiento de la solución obtenida se aplicaron pruebas de caja negra, durante las cuales se detectaron seis no conformidades en una primera etapa, las cuales fueron solucionadas, y posteriormente no se encontraron no conformidades.
- La validación de las variables de la investigación y la aplicación de la técnica ladov permitió medir la satisfacción del cliente respecto al control y disponibilidad de la información con el módulo Económico de SAEF3, donde los resultados obtenidos se traducen en un elevado agrado por parte del cliente con la solución.

**RECOMENDACIONES**

- Hacer extensivo el uso del módulo Económico del SAEF3 al resto de los Vicedecanatos de Economía y Administración de la UCI, para visualizar los resultados que arroja y futuras mejoras.
- Incorporar al módulo desarrollado el control de las dietas ejecutadas en las áreas.

REFERENCIAS

[En línea] <https://github.com/php-fig/fig-standards/blob/master/accepted/PSR-0.md>.

[En línea] <https://github.com/php-fig/fig-standards/blob/master/accepted/PSR-1-basic-coding-standard.md>.

[En línea] <https://github.com/php-fig/fig-standards/blob/master/accepted/PSR-2-coding-style-guide.md>.

**Apache Software Foundation. 2011.** Apache. [En línea] 2011. [Citado el: 16 de Febrero de 2014.] <http://es.opensuse.org/Apache>.

**Belt.es. 2013.** *Belt Ibérica, soluciones de seguridad global.* 2013.

**Boehm, Barry W. 1984.** *Verifying and validating software requirements and design specifications.* s.l. : IEEE software, 1984.

**Bueno Campos, Eduardo, Cruz Roche, Ignacio y Durán Herrera, Juan José. 1979.** *Economía de la empresa. Análisis de las decisiones empresariales.* 2. Madrid : Pirámide, 1979. pág. 718. Vol. I. 843680127X, 9788436801279.

**Buschmann, F., y otros. 1996.** *Pattern Oriented Software Architecture: A System of Patterns.* s.l. : John Wiley & Sons, 1996.

**CEIGE. 2015.** EXCRIBA. *Gestor de Documentos Administrativos.* [En línea] 20 de Enero de 2015. [Citado el: 11 de 02 de 2015.] <http://excriba.prod.uci.cu/page/context/shared/document-details?nodeRef=workspace://SpacesStore/a6002b5f-d3cb-4217-b90d-ddd55621c271>.

**Chiavenato , Idalberto. 2004.** *Introducción a la Teoría General de la Administración.* s.l. : McGraw-Hill Interamericana, 2004.

**Chidamber, Shyam R. y Kemerer, Chris F. 1994.** *A Metrics Suite for Object Oriented Design.* s.l. : Software Engineering, IEEE Transactions, 1994. 0098-5589 .

**Da Silva, Reinaldo O. 2002.** *Teoría de la Administración.* Sao Paulo : Thomson, 2002. pág. 523. 9789706862242.

**Davis, Alan y y otros. 2005.** *Identifying and Measuring Quality in a Software Requirements Specification.* 2005.

**Davis, Duane. 2001.** *Investigación en administración para la toma de decisiones.* [ed.] Marcela Rocha Martínez. [trad.] Duxbury. 5ta. México : International Thomson , 2001. pág. 605. 970-686-022-3.

**Díaz González, Yanette y Fernández Romero, Yenisleidy. 2012.** Patrón Modelo-Vista-Controlador. La Habana : CUJAE, 2012. Vol. 11, 1. 1729-3804.

**Dpto. Sistemas Informáticos y Computación. 2013.** *Introducción a Herramientas CASE y System Architect.* Valencia : Universidad Politécnica de Valencia, 2013.

**Equipo de desarrollo de PostgreSQL. 1996.** *Manual del usuario de PostgreSQL.* s.l. : Thomas Lockhart, 1996.

- Figueroa, Roberth G., Solís, Camilo J. y Cabrera, Armando A. 2008.** *Metodologías Tradicionales VS. Metodologías Ágiles*. s.l. : Universidad Técnica Particular de Loja, Escuela de Ciencias en Computación, 2008.
- Gamma, Dr. Erich, y otros. 1994.** *Design Patterns: Elements of Reusable Object-Oriented Software*. s.l. : Addison-Wesley Professional, 1994. pág. 416. 978-0201633610.
- García Rodríguez, Ana Marys. 2010.** *Validaciones de Requisitos con Expertos Funcionales*. 2010.
- Haimann, Theo. 2000.** *Zona Económica*. 2000.
- Hetzel, Bill. 1993.** *The Complete Guide to Software Testing*. 1993. 978-0471565673.
- IEEE. 1990.** IEEE Standard Glossary of Software Engineering Terminology. s.l. : Computer Society of the IEEE, 1990. 610.12-1990.
- . **2004.** Std 1012-2004 - IEEE Standard for Software Verification and Validation. [ed.] Software Engineering Standards Committee of the IEEE Computer Society. s.l. : IEEE Xplore Digital Library, 2004. 978-0-7381-4642-3.
- . **2002.** Std 610-1990 - IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries. 2002. pág. 217. 978-0-7381-4768-0.
- . **1983.** Std 729-1983. IEEE Guide to Software Requirements Specifications. 1983.
- Jacobson, I., Booch, G. y Rumbaugh, J. 2000.** *El Proceso Unificado de Desarrollo de software*. s.l. : Addison-Wesley, 2000.
- JetBrains. 2000.** The Most Intelligent PHP IDE. [En línea] JetBrains s.r.o, 2000. [Citado el: 05 de febrero de 2015.] <https://www.jetbrains.com/phpstorm/>.
- jQuery Foundation. 2015.** Version 1.9 | jQuery API Documentation. [En línea] 2015. [Citado el: 05 de febrero de 2015.] <https://api.jquery.com/category/version/1.9/>.
- Juristo, Natalia, Moreno, Ana M. y Vegas, Sira. 2004.** *Técnicas de evaluación de Software*. 2004.
- Koontz, Harold y Weihrich, Heinz. 2004.** *Administración Una Perspectiva Global*. s.l. : McGraw-Hill Interamericana, 2004.
- Kuzmina, N. V. 1970.** *Metódicas Investigativas de la actividad pedagógica*. Leningrado : s.n., 1970.
- Liu, Wendy. 2007.** *MVC (Model-View-Controller)*. 2007.
- Lorenz, Mark y Kidd, Jeff. 1994.** *Object-Oriented Software Metrics*. 1994. 9780131792920.
- Méndez Morales, José Silvestre. 2009.** *Fundamentos de economía*. Nueva York : Mc Graw Hill, 2009. 9789701069615.
- Microsoft. 2015.** Microsoft Developer Network. *Revisiones de código y estándares de codificación*. [En línea] 2015. [https://msdn.microsoft.com/es-es/library/aa291591\(v=vs.71\).aspx](https://msdn.microsoft.com/es-es/library/aa291591(v=vs.71).aspx).
- MIT. 2015.** CSS · Bootstrap. [En línea] 2015. [Citado el: 05 de febrero de 2015.] <http://getbootstrap.com/css/>.
- Nieto, Tito Fernando Ale. 2010.** *Introducción al HTML5*. 2010.
- Oliveira Da Silva, Reinaldo. 2002.** *Teorías de la Administración*. s.l. : International Thomson Editores, 2002.

- PAe. 2010.** Portal de Administración Electrónica. [En línea] Ministerio de Hacienda y Administraciones Públicas, 2010. [Citado el: 13 de 02 de 2015.] <http://administracionelectronica.gob.es/ctt/sorolla>.
- Pardo Kuklinski, Hugo. 2005.** *Un modelo de aplicación web institucional universitaria. El caso de los webcom: webs de facultades de comunicación de Iberoamérica.* 2005. Tesis doctoral. 8468962260.
- Pérez, Javier Eguíluz. 2007.** LibrosWeb.: *Introducción a CSS.* [En línea] 2007. [http://www.librosweb.es/css/capitulo\\_1/breve\\_historia\\_de\\_css.html](http://www.librosweb.es/css/capitulo_1/breve_historia_de_css.html).
- Pressman, Roger. 2009.** *Software Engineering: A Practitioner's Approach.* New York : McGraw-Hill, 2009. 978-0-07-337597-7.
- ProgramaciónWeb.Net. 2011.** ¿Que se puede decir del PHP? [En línea] 2011. [Citado el: 8 de Febrero de 2014.] <http://www.programacionweb.net/articulos/articulo/que-se-puede-decir-php/>.
- Ramos Martín , María Jesús, Ramos Martín, Alicia y Rodríguez Montero, Fernando. 2006.** *Sistemas gestores de bases de datos.* 1. 2006. 8448148797.
- Robbins , Stephen y Coulter, Mary. 2005.** *Administración.* s.l. : Pearson Educación, 2005.
- Rumbaugh, James, Jacobson, Ivar y Booch, Grady. 2000.** *El Lenguaje Unificado de Modelado.* s.l. : Addison Wesley, 2000. ISBN: 84-7829-037-0.
- SIUV. 2015.** Universidad de Valencia. [En línea] Servicio de Informática de la Universidad de Valencia, 2015. [Citado el: 13 de 02 de 2015.] <http://www.uv.es/siuv/cas/zapli/geseco.htm>.
- Sommerville, Ian. 2011.** *Software Engineering.* 8va. Boston : Addison-Wesley, 2011. 978-0-13-703515-1.
- Subgerencia Cultural Banco de la República. 2014.** Biblioteca Virtual. *Biblioteca Luis Ángel Arango.* [En línea] 07 de 11 de 2014. <http://www.banrepcultural.org/blaavirtual/ayudadetareas/economia/econo0.htm>.
- symfony.es. 2015.** symfony.es. [En línea] 2015. [Citado el: 5 de Febrero de 2015.] <http://www.symfony.es/2009/03/06/asi-seran-las-novedades-de-symfony-20>.
- The British Standards Institution. 2013.** BSI. [En línea] 2013. [Citado el: 10 de Febrero de 2014.] <http://www.bsigroup.com.mx/es-mx/Auditoria-y-Certificacion/Sistemas-de-Gestion/De-un-vistazo/Que-son-los-sistemas-de-gestion>.
- The PHP Group. 2001.** Manual de PHP. [En línea] 2001. [Citado el: 5 de Febrero de 2015.] <http://www.php.net/manual/es/intro-what-is.php>.
- UCO. 2013.** *Servicio de Gestión Financiera y Presupuestaria.* 2013.
- . **2014.** *Servicio de Informática de la Universidad de Córdoba.* 2014.
- Vergara, Gonzalo. 2009.** Mejora tu Gestión. [En línea] 31 de marzo de 2009. [Citado el: 12 de Febrero de 2014.] <http://mejoratugestion.com/mejora-tu-gestion/que-es-un-sistema-de-gestion/>.
- Visual Paradigm. 2014.** Visual Paradigm for UML. *UML & SysML Toolset.* [En línea] 2014. [Citado el: 10 de Febrero de 2014.] <http://www.visual-paradigm.com/features/uml-and-sysml-modeling/>.

ANEXOS

Anexo #1. Diagramas de procesos de negocio

Proceso de negocio: Control de ejecución del presupuesto

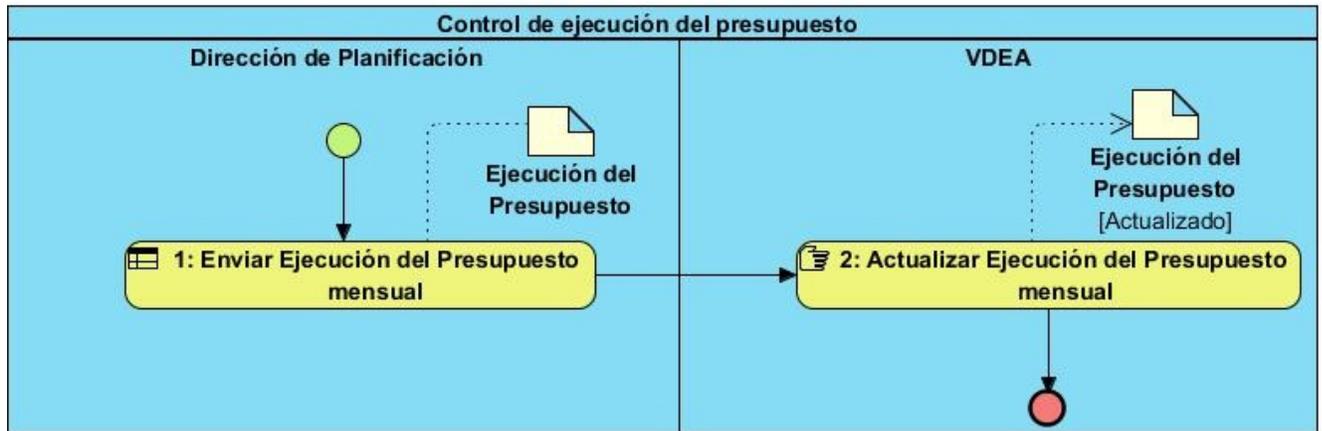


Figura 27. Control del presupuesto.

Fuente: (Elaboración propia)

Proceso de negocio: Control de préstamos y ayudas económicas

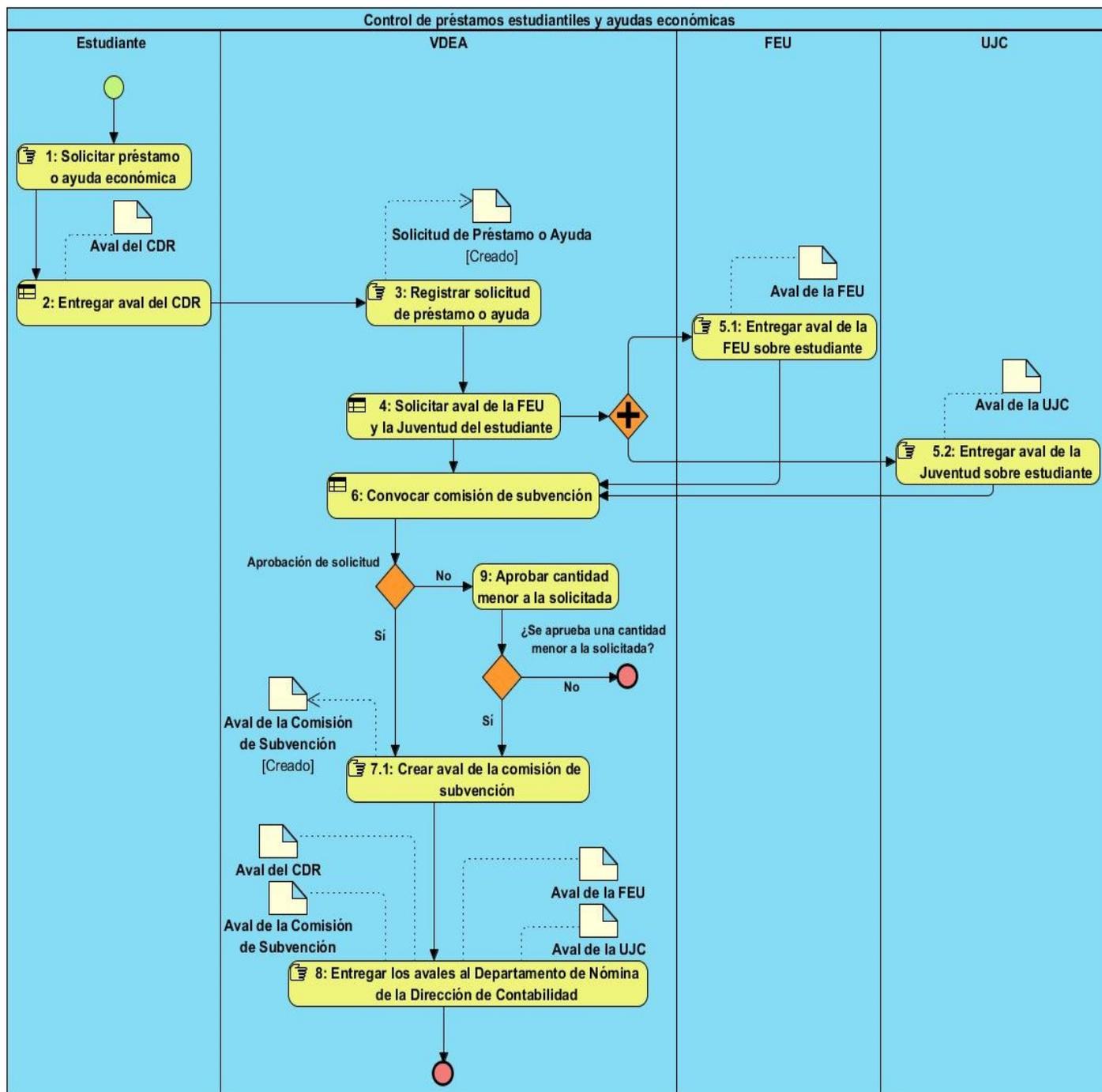


Figura 28. Control de préstamos y ayudas económicas.

Fuente: (Elaboración propia)

Proceso de negocio: Control de bonificación de pasajes a estudiantes

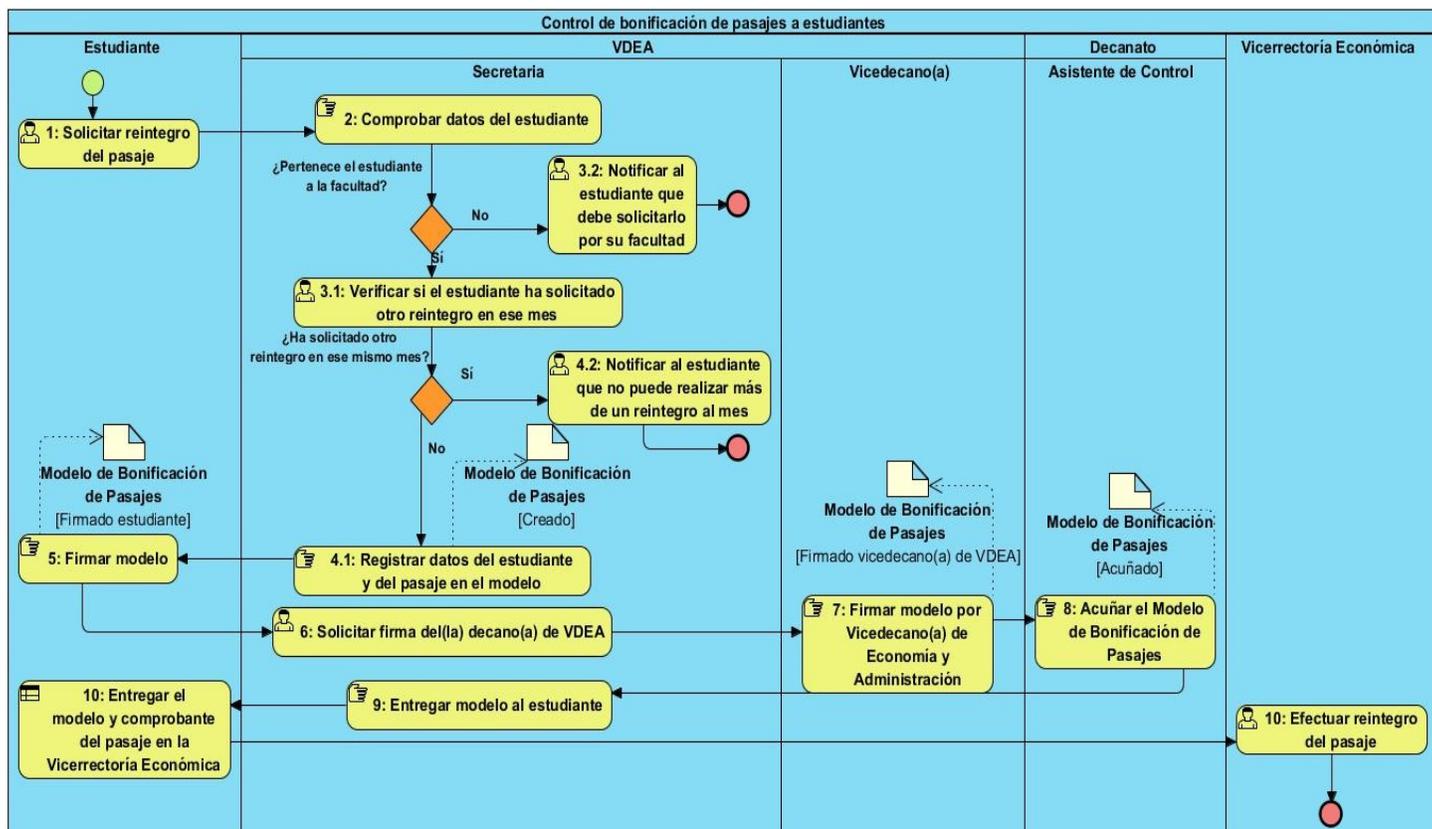


Figura 29. Control de bonificación de pasajes a estudiantes.

Fuente: (Elaboración propia)

Proceso de negocio: Solicitud de recursos materiales

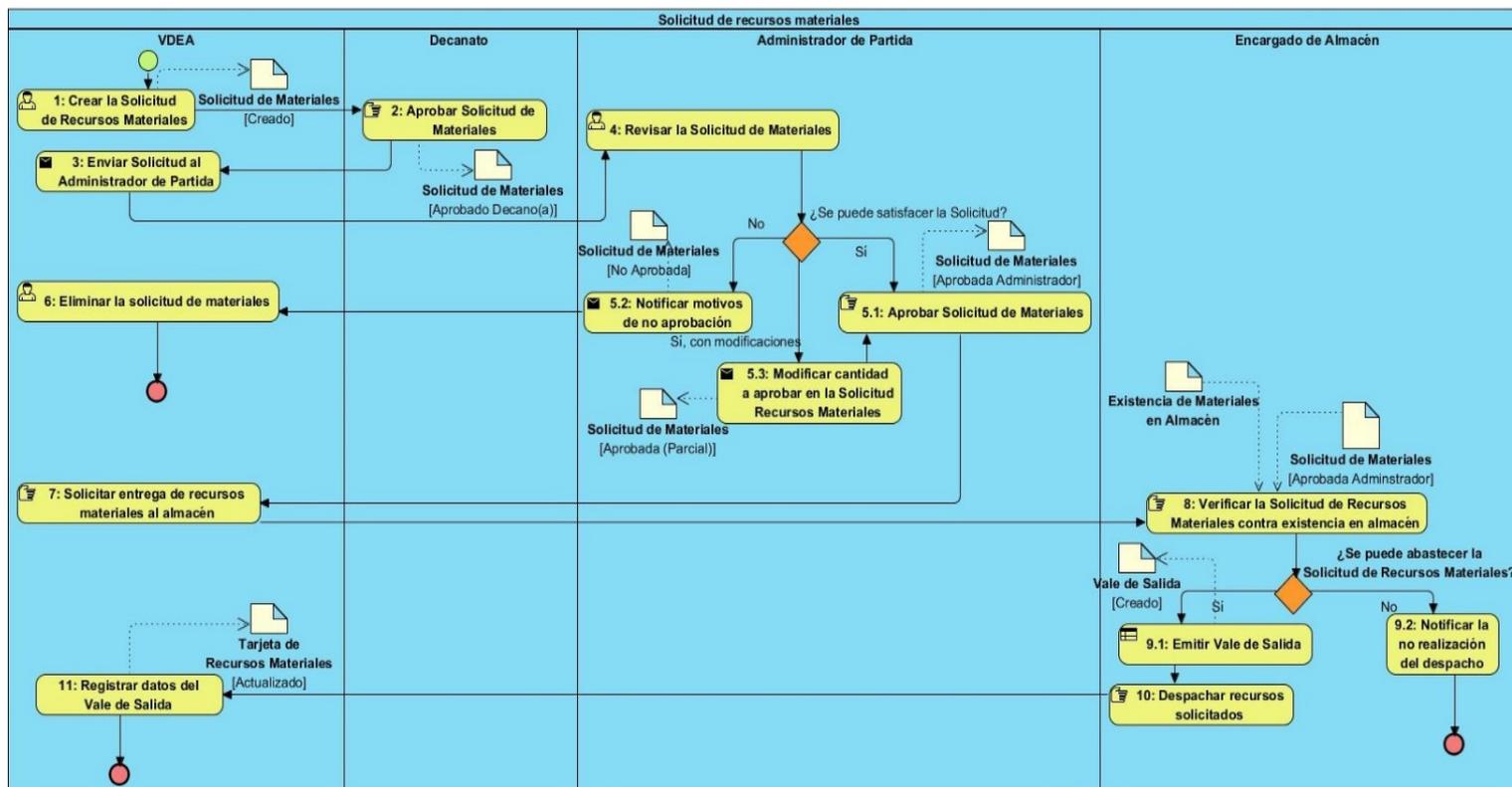


Figura 30. Solicitud de recurso material

Fuente: (Elaboración propia)

## Anexo #2. Prototipos de interfaz de usuario

### Prototipo de interfaz de usuario: Solicitud de recursos materiales

Almacén al que se solicita:  Código:

Destino:  
Proceso:  Código:

Seleccione los materiales a solicitar:

|                                     | Código               | Material                   | UM                   | Cantidad                            |
|-------------------------------------|----------------------|----------------------------|----------------------|-------------------------------------|
| <input type="checkbox"/>            | 21760                | PAPEL FOTOCOPIA 8 1/2 X 11 | RES                  | <input type="text" value="0.0000"/> |
| <input type="checkbox"/>            | 140123               | BOLIGRAFO B-01             | U                    | <input type="text" value="0.0000"/> |
| <input type="checkbox"/>            | 200059               | LAPIZ CON GOMA             | U                    | <input type="text" value="0.0000"/> |
| <input checked="" type="checkbox"/> | <input type="text"/> | <input type="text"/>       | <input type="text"/> | <input type="text" value="0.0000"/> |

Solicitado por:  Autorizado por:   
Cargo:  Cargo:

Figura 31. Prototipo de interfaz de usuario "Solicitar recursos materiales"

Fuente: (Elaboración propia)

Almacén al que se solicita:  Código:

Destino:  
Proceso:  Código:

Seleccione los materiales a solicitar:

|                                     | Código               | Material                   | UM  | Cantidad                             |
|-------------------------------------|----------------------|----------------------------|-----|--------------------------------------|
| <input checked="" type="checkbox"/> | 21760                | PAPEL FOTOCOPIA 8 1/2 X 11 | RES | <input type="text" value="10.0000"/> |
| <input type="checkbox"/>            | 140123               | BOLIGRAFO B-01             | U   | <input type="text" value="15.0000"/> |
| <input checked="" type="checkbox"/> | 200059               | LAPIZ CON GOMA             | U   | <input type="text" value="15.0000"/> |
| <input checked="" type="checkbox"/> | <input type="text"/> | LAPIZ BICOLOR              | U   | <input type="text" value="15.0000"/> |

Solicitado por:  Autorizado por:  Recibido por:   
Cargo:  Cargo:  Cargo:

Figura 32. Prototipo de interfaz de usuario "Modificar solicitud de recursos materiales"

Fuente: (Elaboración propia)

**Almacén al que se solicita:** 01      **Código:** 223.0.12709      **Fecha:** 10/01/2015

**Destino:**

**Proceso:** 101      **Código:** 2060      **Centro de costo:** Facultad3      **Área:** Decanato

**Descripción:**

| Código | Material                   | UM  | Cantidad |   |
|--------|----------------------------|-----|----------|---|
| 21760  | PAPEL FOTOCOPIA 8 1/2 X 11 | RES | 15.0000  | ^ |
| 200059 | LAPIZ CON GOMA             | U   | 10.0000  |   |
|        | LAPIZ BICOLOR              | U   | 10.0000  | v |

**Solicitado por:**      **Autorizado por:**      **Recibido por:**

**Nombre:** Persona Responsable que Solicita      **Nombre:** Persona Responsable que Autoriza      **Nombre:** Persona Responsable que Recibe

**Cargo:** Cargo Solicitante      **Cargo:** Cargo Autorizado      **Cargo:** Cargo Receptor

**25**

Aceptar      Cancelar

Figura 33. Prototipo de interfaz de usuario "Consultar solicitud de recursos materiales"

Fuente: (Elaboración propia)

Prototipo de interfaz de usuario: Asignación de recursos materiales

Área: Área 1

| No. Referencia | Recurso Material | Cantidad | Receptor             | Fecha    |
|----------------|------------------|----------|----------------------|----------|
| YYHHH          | Material 1       | 26       | Nombre 1 Apellidos 1 | dd/mm/aa |
| YYYHH          | Material 2       | 10       | Nombre 1 Apellidos 1 | dd/mm/aa |
| YYYYH          | Material 3       | 12       | Nombre 1 Apellidos 1 | dd/mm/aa |
| YYYY           | Material 4       | 20       | Nombre 1 Apellidos 1 | dd/mm/aa |

Cerrar

Figura 34. Prototipo de interfaz de usuario "Consultar vale de salida"

Fuente: (Elaboración propia)

Destino:  No. de Vale:  Almacén:  Referencia:

Presenta costo en CUC  Presenta costo en MN Fecha:

| Código | Lote | Descripción     | UM  | Cantidad | Costo CUC | Importe CUC | Costo MN | Importe MN | PV CUC | PV MN |
|--------|------|-----------------|-----|----------|-----------|-------------|----------|------------|--------|-------|
| 21770  | 1    | PAPEL FOTOCOPIA | RES | 10       | 0.00000C  | 0.00        | 3.26702C | 32.67      | 0.00   | 0.00  |
| 20005E | 1    | BOLIGRAFO B-01  | UNO | 15       | 0.28000C  | 4.20        | 0.00000C | 0.00       | 0.00   | 0.00  |
| 57494  | 1    | LAPIZ CON GOMA  | UNO | 15       | 0.039202  | 0.59        | 0.01361C | 0.20       | 0.00   | 0.00  |
| 25006  | 1    | LAPIZ BICOLOR   | UNO | 15       | 0.00000C  | 0.00        | 0.14095C | 2.11       | 0.00   | 0.00  |
| Total: |      |                 |     |          |           | \$4.79      |          | \$45.44    |        |       |

PV: Precio Venta

Aceptar Cancelar

Figura 35. Prototipo de interfaz de usuario "Modificar vale de salida"

Fuente: (Elaboración propia)

Prototipos de interfaz de usuario: Bonificación de pasajes a estudiantes

Fecha: dd/mm/aaaa

Estudiante: Nombre Apellido1 Apellido2      CI: XXZZCCVVBBN      Facultad: 3

Origen      Destino

Fecha de ida: dd/mm/aa      Provincia: Provincia1      Provincia2

Municipio: Municipio1      Municipio2

Fecha de vuelta: dd/mm/aa      Provincia: Provincia2      Provincia1

Municipio: Municipio2      Municipio1

Persona que autoriza: Nombre Apellido1 Apellido2

Aceptar      Cancelar

Figura 36. Prototipo de interfaz de usuario "Consultar bonificación de pasajes a estudiantes"

Fuente: (Elaboración propia)

Estudiante: Nombre Apellido1 Apellido2      CI: 12345678901

Origen      Destino

Fecha de ida: dd/mm/aa      Provincia: Provincia1      Provincia2

Municipio: Municipio1      Municipio2

Fecha de vuelta: dd/mm/aa      Provincia: Provincia2      Provincia1

Municipio: Municipio2      Municipio1

Persona que autoriza: Nombre Apellido 1 Apellido 2

Aceptar      Cancelar

Figura 37. Prototipo de interfaz de usuario "Modificar bonificación de pasajes a estudiantes"

Fuente: (Elaboración propia)

Prototipo de interfaz: Préstamos y ayudas económicas a estudiante

Solicita:

---

**I- Datos Personales del Solicitante**

Nombre(s):  Primer Apellido:  Segundo Apellido:

Sexo:  Estado Civil:  Nacionalidad:  Ct:

Natural de:  
 Provincia:  Municipio:  Inscripción SMG:

Dirección Particular:  
 Calle:  No:  Piso:  Apto:   
 Entre calles:  y  Reparto:   
 Ciudad o Pueblo:  Municipio:  Provincia:   
 País:  Teléfono:  Residencia:  UM.No:

Organizaciones:  FMC  UJC  CDR  PCC

---

**II- Datos Docentes**

Centro de estudio de procedencia:  Índice académico:

Provincia:  Municipio:  ¿Becario anterior?  Sí  No

Asignaturas pendientes: Curso:  -  Año:

Asignatura:  Del año:  Motivos:

---

**III- Datos del Núcleo Familiar**

| Nombre y Apellidos                              | Edad                 | Parentesco           | Estado Civil                            | CToE                 | IM                   |
|---|----------------------|----------------------|---|----------------------|----------------------|
| <input type="text" value="Nombre y Apellidos"/> | <input type="text"/> | <input type="text"/> | <input type="text" value="Seleccione"/> | <input type="text"/> | <input type="text"/> |
| <input type="text" value="Nombre y Apellidos"/> | <input type="text"/> | <input type="text"/> | <input type="text" value="Seleccione"/> | <input type="text"/> | <input type="text"/> |

CToE: Centro de Trabajo o Estudio  
 IM: Ingreso Mensual

---

**IV- Personas que dependen económicamente del solicitante fuera de su núcleo**

¿Dependen personas económicamente de usted? :  Sí  No

| Nombre y Apellidos                              | Edad                 | Parentesco           | Estado Civil                            | CToE                 | IM                   |
|---|----------------------|----------------------|---|----------------------|----------------------|
| <input type="text" value="Nombre y Apellidos"/> | <input type="text"/> | <input type="text"/> | <input type="text" value="Seleccione"/> | <input type="text"/> | <input type="text"/> |
| <input type="text" value="Nombre y Apellidos"/> | <input type="text"/> | <input type="text"/> | <input type="text" value="Seleccione"/> | <input type="text"/> | <input type="text"/> |

CToE: Centro de Trabajo o Estudio  
 IM: Ingreso Mensual

V- Fundamentación

Explique los motivos de la solicitud

---

VI- Para solicitudes de préstamo estudiantil reintegrable

¿Disponía de préstamo anteriormente?  Sí  No

Centro donde lo recibía antes:  Desde:  Hasta:

¿Subvencionado por la Resolución 25867 del MINTRAB?  Sí  No

¿Incorporado al Movimiento de Alumnos Ayudantes?  Sí  No

Préstamo mensual solicitado:

Importe:  Período de tiempo que solicita:

---

VII- Para uso de la facultad (Puede aprobar-proponer según el tipo de solicitud)

Analizada la fundamentación y los datos consignados se decide

Por un importe de:  Periodicidad:  Desde:  Hasta:

Nombre:  Cargo:

---

VIII- Observaciones

Observaciones para la facultad

Figura 38. Prototipo de interfaz de usuario "Adicionar préstamo o ayuda económica"

Fuente: (Elaboración propia)

**Alta de solicitud de:** Préstamo**I- Datos Personales del Solicitante****Nombre(s):** Estudiante      **Primer Apellido:** Apellido 1      **Segundo Apellido:** Apellido 2**Sexo:** M      **Estado Civil:** Soltero      **Nacionalidad:** Cubano      **CI:** 12345678901**Natural****Provincia:** La Habana      **Municipio:** La Lisa**Dirección Particular:**

Carretera a San Antonio Km 2 ½ . Torrens. Boyeros. La Habana. Cuba

**Teléfono:** 8372402      **Residencia:** De un familiar**Organizaciones:**  FMC  UJC  CDR  PCC**II- Datos Docentes****Centro de estudio de procedencia:** Instituto Preuniversitario      **Índice académico:** 99.04**Provincia:** La Habana      **Municipio:** La Lisa      ¿Becario anterior?  Sí  No**Asignaturas pendientes:****Asignatura:** No tiene**III- Datos del Núcleo Familiar**

| Nombre y Apellidos        | Edad | Parentesco | Estado Civil  | CToE          | IM     |
|---------------------------|------|------------|---------------|---------------|--------|
| Familiar 1 del Estudiante | 58   | Madre      | Divorciado(a) | Técnico Medio | 235.00 |
| Familiar 2 del Estudiante | 67   | Abuela     | Viudo(a)      | Jubilada      | 180.00 |

CToE: Centro de Trabajo o Estudio

IM: Ingreso Mensual

**IV- Personas que dependen económicamente del solicitante fuera de su núcleo**¿Dependen personas económicamente de usted? :  Sí  No**V- Fundamentación**

Estoy solicitando el préstamo por los motivos siguientes...

**VI- Para solicitudes de préstamo estudiantil reintegrable**¿Disponía de préstamo anteriormente?  Sí  No¿Subvencionado por la Resolución 25867 del MINTRAB?  Sí  No¿Incorporado al Movimiento de Alumnos Ayudantes?  Sí  No**Préstamo mensual solicitado:****Importe:** 100.00      **Período de tiempo que solicita:** trimestre

**VII- Para uso de la facultad (Puede aprobar-proponer según el tipo de solicitud)**

**Analizada la fundamentación y los datos consignados se decide** proponer

**Por un importe de:** 80.00    **Periodicidad:** mensual    **Desde:** 01/06/2015    **Hasta:** 01/07/2016

**Nombre:** Persona Autorizada Aprueba    **Cargo:** Jefe

---

**VIII- Observaciones**

La Comisión de Subvención determinó aprobar el importe determinado a pagarse mensualmente debido a los motivos siguientes ...

Aceptar    Cancelar

**Figura 39. Prototipo de interfaz de usuario "Consultar préstamo o ayuda económica"**

Fuente: (Elaboración propia)

Solicita: Préstamo

---

**I- Datos Personales del Solicitante**

Nombre(s): Estudiante    Primer Apellido: Apellido 1    Segundo Apellido: Apellido 2

Sexo: M    Estado Civil: Soltero(a)    Nacionalidad: Cubano    CI: 12345678901

Natural de:

Provincia: La Habana    Municipio: La Lisa    Inscripción SMG:

Dirección Particular:

Calle: Carretera a San Antonio Km 2 1/2    No:    Piso:    Apto:    Reparto:    Entre calles:    y    Ciudad o Pueblo: Torrens    Municipio: Boyeros    Provincia: La Habana

País: Cuba    Teléfono: 8372402    Residencia: De un familiar    UM.No:    Organizaciones:  FMC  UJC  CDR  PCC

---

**II- Datos Docentes**

Centro de estudio de procedencia: Instituto Preuniversitario    Índice académico: 99.04

Provincia: La Habana    Municipio: La Lisa    ¿Becario anterior?  Sí  No

Asignaturas pendientes:    Curso:    Año:    Asignatura: Título de la asignatura    Del año:    Motivos: Escriba los motivos aquí

---

**IV- Personas que dependen económicamente del solicitante fuera de su núcleo**

¿Dependen personas económicamente de usted? :  Sí  No

V- Fundamentación

Estoy solicitando el préstamo por los motivos siguientes...

VI- Para solicitudes de préstamo estudiantil reintegrable

¿Disponía de préstamo anteriormente?  Sí  No

¿Subvencionado por la Resolución 25867 del MINTRAB?  Sí  No

¿Incorporado al Movimiento de Alumnos Ayudantes?  Sí  No

Préstamo mensual solicitado:

Importe:  Período de tiempo que solicita:  (mes, trimestre, curso)

VII- Para uso de la facultad (Puede aprobar-proponer según el tipo de solicitud)

Analizada la fundamentación y los datos consignados se decide

Por un importe de:  Periodicidad:  Desde:  Hasta:

Nombre:  Cargo:

VIII- Observaciones

La Comisión de Subvención determinó aprobar el importe determinado a pagarse mensualmente debido a los motivos siguientes ...

**Figura 40. Prototipo de interfaz de usuario "Modificar préstamo o ayuda económica"**

Fuente: (Elaboración propia)

## Anexo #3. Resultado de aplicación de la métrica TC

| No. | Clases               | Operaciones | Responsabilidad | Complejidad | Reutilización |
|-----|----------------------|-------------|-----------------|-------------|---------------|
| 1.  | eAsignacion          | 12          | baja            | baja        | media         |
| 2.  | eAsignaturaPendiente | 9           | baja            | baja        | alta          |
| 3.  | eAval                | 7           | baja            | baja        | alta          |
| 4.  | eAyudantiaEstado     | 9           | baja            | baja        | alta          |
| 5.  | eDatosDocentes       | 18          | media           | media       | media         |
| 6.  | eDatosVivienda       | 6           | baja            | baja        | alta          |
| 7.  | eDireccion           | 22          | media           | media       | media         |
| 8.  | eEstudiante          | 28          | alta            | alta        | baja          |
| 9.  | eAlmacen             | 10          | baja            | baja        | alta          |
| 10. | eArea                | 9           | baja            | baja        | alta          |
| 11. | eAsignatura          | 7           | baja            | baja        | alta          |
| 12. | eCentroCosto         | 10          | baja            | baja        | alta          |
| 13. | eDepartamento        | 6           | baja            | baja        | alta          |
| 14. | eEstado              | 14          | media           | media       | media         |
| 15. | eEstadoCivil         | 5           | baja            | baja        | alta          |
| 16. | eMunicipio           | 8           | baja            | baja        | alta          |
| 17. | eOrganizacion        | 9           | baja            | baja        | alta          |
| 18. | ePartida             | 13          | media           | media       | media         |
| 19. | eProvincia           | 6           | baja            | baja        | alta          |
| 20. | eTipoVivienda        | 7           | baja            | baja        | alta          |
| 21. | ePrestamoAyuda       | 37          | alta            | alta        | baja          |
| 22. | ePrestamoAnterior    | 9           | baja            | baja        | alta          |
| 23. | eOperacion           | 11          | baja            | baja        | alta          |
| 24. | eNucleoFamiliar      | 17          | media           | media       | media         |
| 25. | eUnidadMedida        | 6           | baja            | baja        | alta          |
| 26. | eReintegroPasaje     | 21          | media           | media       | media         |
| 27. | eSalida              | 10          | baja            | baja        | alta          |
| 28. | eSolicitudMaterial   | 26          | alta            | alta        | baja          |
| 29. | eTarjetaEstibajaa    | 24          | media           | media       | media         |

Tabla 13. Clases del modelo a las que se les aplicó la métrica TC

Fuente: (Elaboración propia)